Journal of Applied Physics TUTORIAL pubs.aip.org/aip/jap

Lattice dynamics modeling of thermal transport in solids using machine-learned atomic cluster expansion potentials: A tutorial

Cite as: J. Appl. Phys. **137**, 081101 (2025); doi: 10.1063/5.0251119
Submitted: 29 November 2024 · Accepted: 30 January 2025 ·
Published Online: 24 February 2025

Liben Guo,[1] Yuanbin Liu,[2] Lei Yang,[1] and Bingyang Cao[1,a)]

AFFILIATIONS

[1]Key Laboratory for Thermal Science and Power Engineering of Ministry of Education, Department of Engineering Mechanics, Tsinghua University, Beijing 100084, China
[2]Inorganic Chemistry Laboratory, Department of Chemistry, University of Oxford, Oxford OX1 3QR, United Kingdom

[a)]**Author to whom correspondence should be addressed:** caoby@tsinghua.edu.cn

ABSTRACT

Lattice dynamics (LD) plays a crucial role in investigating thermal transport in terms of not only underlying physics but also novel properties and phenomena. Recently, machine learning interatomic potentials (MLIPs) have emerged as powerful tools in computational physics and chemistry, showing great potential in providing reliable predictions of thermal transport properties with high efficiency. This tutorial provides a comprehensive guideline for MLIPs' development and how they are used for the computational modeling of thermal transport. Using atomic cluster expansion (ACE) as the paradigmatic potential, we introduce the essential fundamentals of MLIPs, including data construction, model training, and hyperparameter optimization. With the developed ACE potentials, we further showcase their applications in the LD modeling of thermal transport for crystalline silicon and amorphous carbon. The corresponding code implementations for MLIP applications in calculating thermal conductivity are also provided for beginners to follow.

I. INTRODUCTION

Thermal transport is one of the most critical properties of solid materials, playing an essential role in a range of technological applications, such as electrical device design and thermoelectric energy conversion.[1–3] Thermal conductivity of materials is significantly related to chemistry compositions, atomic structures, sizes, and temperatures in complex manners, bringing challenges for thermal transport simulation, optimization, and manipulation in practical applications.[4–7] For in-depth understanding of thermal transport, there have been extensively developed computational methods, in which the phonon Boltzmann transport equation (PBTE)[8,9] combined lattice dynamics (LD) becomes widely adopted. This approach offers indispensable insight with high reliability, as it can derive accurate phonon properties from force constants (FCs) calculated using the quantum-mechanically accurate Density Functional Theory (DFT).[10,11] However, the high computational cost of DFT calculations makes first-principles methods quite expensive and limits their applications in complex situations, such as high-order anharmonicity and glasses.[12,13] Typically, DFT evaluations are restricted to the system with a few thousand atoms and only hundreds of atoms when conducting "*ab initio*" simulations.[14,15]

In recent years, machine learning interatomic potentials (MLIPs) have been attracting researchers' attention due to their computational efficiency and accuracy as a compelling alternative to DFT. In MLIP models, a potential energy surface (PES) is constructed for the accurate prediction of energies and forces by learning from the DFT reference results. They offer comparable accuracy but several orders of magnitude higher computational efficiency than DFT,[16–18] enabling reliable large-scale (up to millions of atoms) atomic simulations[19,20] and prohibitive computations for fourth- or higher-order phonon scatterings.[21–24] The applications of MLIPs in thermal properties prediction, with advantages in efficiency and accuracy, have been widely reported.[25–30] To date, numerous MLIPs have been developed with the corresponding code implementations for computational physics, chemistry, and

TABLE I. List of the representative MLIP models.

Model	Type	Supporting codes	Year	Reference
ACE	Kernel Regression	ASE, LAMMPS	2019	Drautz[34]
GAP	Kernel Regression	ASE, LAMMPS	2010	Bartok[35]
NEP	NN (Neural Net)	ASE, LAMMPS, GPUMD[36]	2021	Fan[37]
DeepPotential	NN	ASE, LAMMPS	2018	Han[38]
MACE	GraphNN	ASE, LAMMPS	2023	Batatia[39]
Allegro	GraphNN	ASE, LAMMPS	2023	Musaelian[20]
NequIP	GraphNN	ASE, LAMMPS	2022	Batzner[40]

materials. Representative MLIP models are enumerated in Table I, along with their regression types and supported tools such as the Atomic Simulation Environment (ASE)[31,32] in Python and the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS).[33] For thermal transport simulation, developing proper MLIPs and practical workflows remains nontrivial.

Herein, the purpose of this tutorial is to provide guidance for beginners on the development and application of MLIPs for thermal transport calculations. We aim to complement relevant works by others: focus on a step-by-step guide to developing MLIPs from scratch and applying them in thermal calculations using well-established interfaces. Our work strongly emphasizes MLIP applications in LD methods, complementing a related work by Dong *et al.*,[41] which focused on Molecular Dynamics (MD) simulation. For the theoretical details of LD, we refer interested readers to the previous papers.[42–45] We present the basic concepts, data construction, model training, and hyperparameters' selection segments. Additionally, we feature crystalline silicon and amorphous carbon to illustrate the practical implementation of MLIPs and the derived thermal transport calculations. For this tutorial, the atomic cluster expansion (ACE) potential has been selected as the paradigmatic model due to its precision with efficiency on high-performance parallel computation,[46,47] meeting a range of computational requirements. It is worth noting that although only the ACE potential is considered, the presented instructions are also applicable to other MLIPs.

The remaining parts of this tutorial are organized as follows. In Sec. II, we introduce the overall procedure of MLIP development, which includes a brief introduction to the ACE potential framework, data construction, potential training and estimation, and the optimization and selection of model hyperparameters. In Sec. III, we bring two applications of thermal calculations using MLIPs for crystalline silicon and amorphous carbon, respectively. Finally, this tutorial is summarized in Sec. IV.

II. MLIP DEVELOPMENT

A. Fundamental concepts and atomic cluster expansion framework

The primary purpose of MLIPs is to model the PES of a target system, from which the total energy and interatomic forces are derived based on a set of N atomic positions. This modeling process, represented as a regression task in mathematics, starts with the description of total energy, as the forces can be obtained by differentiating the energy with respect to the atomic positions. To simplify computational complexity and enhance efficiency, the total energy is decomposed into the sum of individual atomic contributions,

$$E_{\mathrm{tol}} = \sum_i E_i, \tag{1}$$

where the site energy E_i is determined by the atomic environment within a specified cutoff radius r_{cut} of the center atom under the assumption of localization. In the ACE framework, it is further represented by a general function $\mathcal{F}$ of p different atomic properties determined by the same atomic environment as

$$E_i = F\left(\varphi_i^{(1)},\ \varphi_i^{(2)},\ \ldots,\ \varphi_i^{(p)}\right). \tag{2}$$

These atomic properties $\varphi_i^{(p)}$ are then expanded on a group of basis functions as

$$\varphi_i^{(p)} = \sum_{\boldsymbol{\nu}} \tilde{c}_{\boldsymbol{\nu}}^{(p)} \boldsymbol{A}_{i\boldsymbol{\nu}}, \tag{3}$$

where $\boldsymbol{A}_{i\boldsymbol{\nu}}$ are the many-body basis functions and $\tilde{c}_{\boldsymbol{\nu}}^{(p)}$ are the expansion coefficients. Since the potential energy can be considered a sum of contributions of pair (two-body) potential, three-body potential, and so on, the basis functions of body order $\nu + 1$ can be defined as a product like

$$\boldsymbol{A}_{i\boldsymbol{\nu}} = \prod_{t=1}^{\nu} A_{i\boldsymbol{\nu}_t}. \tag{4}$$

Considering the atomic local environment of atomic species, the basis functions are further written as

$$A_{i\boldsymbol{\nu}} = \sum_j \delta_{\mu\mu_j} \phi_{\boldsymbol{\nu}}(\boldsymbol{r}_{ji}), \tag{5}$$

where μ_j is the species of a nearby atom j in the local atomic environment, $\delta_{\mu\mu_j}$ is the Kronecker symbol to consider only the bond type of species μ_j and μ, and $\boldsymbol{r}_{ji}$ denotes the relative position vector pointed from atom i to j. $\phi_{\boldsymbol{\nu}}$ is a single-bond basis function, which can then be given in terms of a radial function R_{nl} multiplied with a spherical harmonic function Y_{lm},

$$\phi_{\mu\mu_j nlm} = R_{nl}^{\mu\mu_j}(r_{ji}) Y_{lm}(\hat{\boldsymbol{r}}_{ji}). \tag{6}$$

Here, the compressed index $\boldsymbol{\nu}$ is expanded into a list of indices characterizing the chemical species μ and (nlm) indicating the

types of radial functions and spherical harmonic functions. The radial functions depend on the pairwise distance r_{ji} between atoms i and j, and the spherical harmonic functions depend on the respective direction $\hat{\boldsymbol{r}}_{ji}$. This expansion form fulfills the symmetry requirements of permutational and translational invariance.

It is worth noting that in Eq. (4), the basis functions $\boldsymbol{A}_{iv}$ are not invariant under rotation due to the selection of spherical harmonic functions; thus, the expansion coefficients $\tilde{c}_{\boldsymbol{v}}^{(p)}$ may vary under rotation. For implementation, the invariance is obtained by multiplying the basis with the generalized Clebsch–Gordan coefficients for appropriate symmetries. This transforms the expansion into another form,

$$\varphi_i^{(p)} = \sum_{\boldsymbol{v}} c_{\boldsymbol{v}}^{(p)} \boldsymbol{B}_{i\boldsymbol{v}}, \tag{7}$$

where $c_{\boldsymbol{v}}^{(p)}$ are the learnable parameters in the ACE model that can be optimized during fitting. More information about the ACE framework can be found in the cited literature.[34,46,47]

B. Data collection

Unlike empirically fitted potentials, MLIPs do not impose any prior assumptions on the functional form of the PES. Instead, the shape of the potential is directly learned through a regression process from a large set of reference configurations with their corresponding energies and forces, typically obtained from DFT calculations. Collectively, these elements constitute the dataset for training and validating the ML potential. Given this nonparametric nature of ML potentials, which lack specific physical function forms and involve tens of thousands of parameters to be determined, the selection of reference data is especially critical. Data collection must align with the application purposes of MLIPs: a general potential requires a diverse range of configurations including crystalline, amorphous, and surface structures,[48–50] while tasks of LD calculations for specific crystals required more targeted data.[25,30] To date, a variety of data construction approaches have been developed. Here, we will present a spectrum of strategies from manually crafted approaches to automated protocols.

1. Hand-built approaches

Directly selecting configurations for the dataset based on physical intuition is a straightforward yet practical and widely adopted method.[51,52] The common artificial methods include the perturbation method and *Ab Initio* Molecular Dynamics (AIMD).[53]

As illustrated in Fig. 1(a), the perturbation method begins with structures positioned at energy minima and then introduces random displacements to atoms in their supercells to generate reference configurations. This procedure can be conveniently executed with ASE's rattle function, following a specific statistical distribution, most commonly a Gaussian distribution with a predefined standard deviation (e.g., 0.1 Å). However, structures generated by this stochastic approach might sometimes exhibit large interatomic forces due to the possibility of atoms being placed very close together, resulting in significant repulsive forces. To address this issue, constraints on a minimum atomic distance can be added via a Monte Carlo (MC) process, as introduced in Ref. 54, where configurations with short atomic distances below a threshold are only accepted with low probability. Moreover, to ensure that the potential accurately forecasts forces, it might sometimes be necessary to subject the lattice to stress through tension or compression.[49] After acquiring these reference structures, they are subsequently employed to produce a reference dataset via DFT calibration.

Despite its high efficiency, the perturbation method requires initial structures, which cannot be obtained when considering amorphous and liquid materials. For this purpose, the more computationally demanding AIMD method [Fig. 1(b)] provides a distinct advantage in constructing datasets that lack a definable structural prototype. It involves performing an MD simulation for a specified duration, after which snapshots are sampled at regular intervals or using other sampling methods to form a dataset. In practice, these two hand-built approaches can be combined for data collection.[52]

2. Active learning protocols

Manually constructing datasets demands researchers to have extensive experience, which may inevitably introduce inductive biases. This can lead to high redundancy in structural features and an imbalance in data distribution, resulting in unnecessary increase in data collection efforts and reduced transferability of the potential.[17,55–58] Active learning (AL) strategies are employed to address this issue through an iterative procedure to gradually improve the ML model. Starting with a limited collection of labeled configurations, the procedure iteratively performs the ML potential training and appends the newly labeled data until certain stopping criteria are met. Here, selection rules for the appended data are based on predefined metrics centered around uncertainty, which quantifies the expected error of prediction on a given configuration. Two of the most commonly used definitions of uncertainty can be estimated through an analytical method[58,59] or Query-by-Committee (QbC),[60–63] as depicted in Figs. 1(c) and 1(d).

The analytical method is basically restricted to potentials with linear models or Gaussian process type models, where the model parameters and its prediction errors can be assumed as an isotropic Gaussian prior,

$$y = E + \varepsilon = cB + \varepsilon, \tag{8}$$

where $\boldsymbol{c} \sim \mathcal{N}(0, \alpha^{-1}\boldsymbol{I}), \varepsilon \sim \mathcal{N}(0, \lambda^{-1})$. From the given training dataset $\{\tilde{\boldsymbol{B}}, \tilde{y}\}$ in the current iteration, it can be derived that the posterior of model parameters also obeys a Gaussian distribution:

$$p(\boldsymbol{c}|\tilde{\boldsymbol{B}}, \tilde{y}) \propto N(\boldsymbol{c}|\boldsymbol{\mu}_{\boldsymbol{c}}, \boldsymbol{\Sigma}_{\boldsymbol{c}}), \tag{9}$$

where $\boldsymbol{\mu}_{\boldsymbol{c}} = \lambda \boldsymbol{\Sigma}_{\boldsymbol{c}} \tilde{\boldsymbol{B}} \tilde{y}$, $\boldsymbol{\Sigma}_{\boldsymbol{c}} = (\lambda \tilde{\boldsymbol{B}}^T \tilde{\boldsymbol{B}} + \alpha \boldsymbol{I})^{-1}$. The details of the derivation can be figured out in Refs. 58 and 64. This gives the determination of learnable parameters. Then, for a new configuration $\boldsymbol{B}^*$, its respective Bayesian evaluation is

$$y^* \sim N(\boldsymbol{\mu}_{\boldsymbol{c}} \boldsymbol{B}^*, \lambda^{-1} + \boldsymbol{B}^T \boldsymbol{\Sigma}_{\boldsymbol{c}} \boldsymbol{B}). \tag{10}$$

Thus, the analytical uncertainty can be inherently defined as the standard deviation of the prediction $\sigma_{\text{ana}} = \sqrt{\lambda^{-1} + \boldsymbol{B}^T \boldsymbol{\Sigma}_{\boldsymbol{c}} \boldsymbol{B}}$. Once

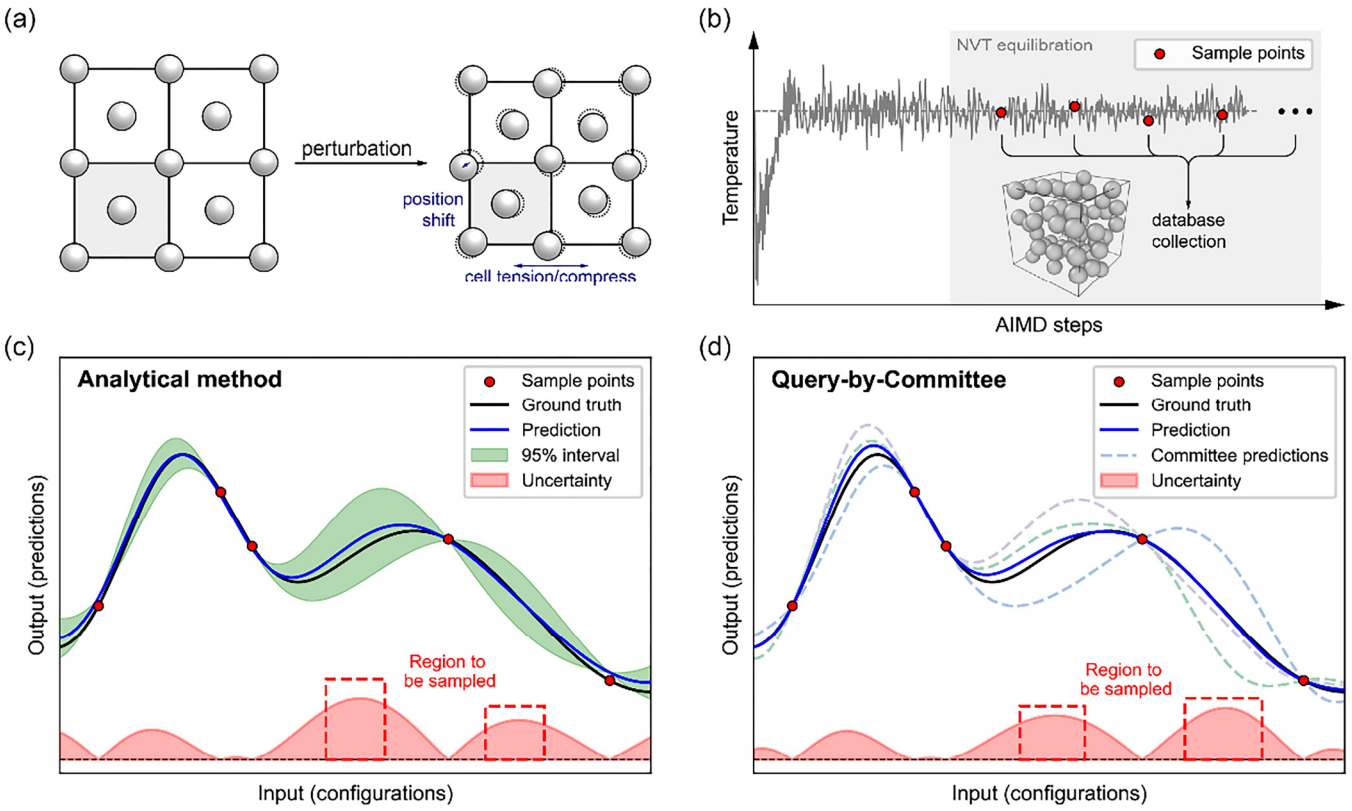

FIG. 1. Approaches for data collection. (a) Perturbation method. (b) AIMD method. (c) Analytical definition of uncertainty in AL. (d) Query-by-Committee definition of uncertainty in AL.

the uncertainty of all configurations is estimated, those configurations with the highest uncertainty can be sampled and augmented to the dataset, enhancing the potential's transferability.

Another approach to determine uncertainty can be implemented using the QbC technique, where the so-called "committee" consists of an ensemble of potentials with the same hyperparameters. In each iteration, these potentials are trained on different subsamples of the dataset, with a small fraction of certain data left out in each case. Due to the difference in the training dataset and stochastic initialization, the learned parameters of potentials in this committee vary, resulting in different predictions $\{E_1, \ldots E_{N_{\text{com}}}\}$ for the same configuration [dash line in Fig. 1(d)]. The uncertainty can then be estimated as the standard deviation of these predictions as

$$\sigma_{QbC} = [\text{Var}(E_1, \ldots, E_{N_{\text{com}}})]^{\frac{1}{2}}. \tag{11}$$

New configurations with high uncertainty are then added to the training dataset iteratively, similar to the analytical approach. Once the estimated uncertainties on all configurations are below a given threshold or other criteria are met, all data are collected as the final dataset to train a single potential.

Moreover, other AL schemes, including entropy maximization on feature space distance, dropout method, etc., are detailed in their respective literature structures.[64,65]

C. Training and estimation

With the data prepared, it will be divided into two parts (e.g., 80–20 split), where the majority serves as a training dataset for ML potential's development and the remainder is used for estimating the performance of the trained potential as the testing dataset. This division can be performed using the random division or stratified sampling techniques, ensuring both the diversity of the training data and the comprehensiveness of the potential's performance assessment across the entire PES landscape.

Specifically, the purpose of the training process is to optimize the model's learnable parameters so that the predictions of energy and force labels align with DFT reference in the training dataset, thereby accurately fitting the PES of the target system. This can be achieved by minimizing the loss function with the

following form:

$$L = \kappa \sum_{n=1}^{N_{\text{struct}}} \left(\frac{E_n^{\text{pred}} - E_n^{\text{ref}}}{N_{at,\,n}} \right)^2 + (1-\kappa) \sum_{n=1}^{N_{\text{struct}}} \frac{1}{N_{at,\,n}} \sum_{i=1}^{N_{at,\,n}} \boldsymbol{F}_{n,\,i}^{\text{pred}} - \boldsymbol{F}_{n,\,i}^{\text{ref}\,2}, \tag{12}$$

where N_{struct} and $N_{at,\,n}$ denote the number of structures and number of atoms in the nth structure, respectively, and κ represents the relative weight between the accuracy of energy prediction and forces prediction. Note that atomic forces are derived from the differential of the system's energy with respect to the atomic coordinates, which implies that fitting solely on energy could be adequate with sufficient accuracy. However, the complexity inherent in many-body interactions makes this challenging to achieve. It is necessary to balance the trade-off between the accuracy of energy and force. This relationship is illustrated in Fig. 2.

Furthermore, in practice, we often incorporate regularization terms into the loss function. This is especially necessary when the training dataset is small relative to the number of model parameters, as there is a risk that the parameters may fall on specific values that cause overfitting.[66,67] Such a surface may exhibit very steep slopes and provide poor predictions on configurations away from the data points. Restricting the magnitude of the potential function parameters helps to prevent these abrupt changes, ensuring a certain smoothness of the potential energy surface and, thus, mitigating overfitting to some extent. Consequently, the loss function, with the inclusion of the regularization term, can be reformulated in the following form:

$$L = \kappa \Delta_E + (1-\kappa)\Delta_F + L_1 \sum_{c\ in\ \boldsymbol{c}} |c| + L_2 \sum_{c\ in\ \boldsymbol{c}} |c|^2, \tag{13}$$

where Δ_E and Δ_F denote the energy and force terms in Eq. (12), respectively. The weighted absolute values of the magnitude and squared magnitude of the model's coefficient are named L1 and L2 regularization, respectively, acting as penalty terms to the loss function.[68]

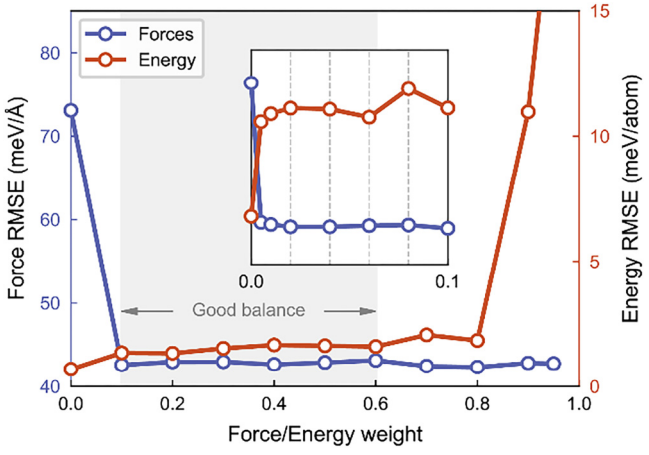

FIG. 2. The trade-off relationship between the accuracy of energy and force in MLIP training.

Subsequently, utilizing the established objective loss function, parameters optimization can be carried out using widely used techniques such as Adam[69,70] or LBFGS,[71,72] for training the potential function. Additional elaboration on these optimization techniques is omitted in this context.

D. Pareto optimal and hyperparameters selection

Given a specific dataset and model setting, the training of MLIPs determines the predictive accuracy of energy and force. By refining the model's hyperparameters, such as using a larger cutoff radius or increasing the order of many-body terms for higher-order interactions, the accuracy of MLIPs may be enhanced. However, these adjustments can lead to an increase in the computational cost, which is particularly detrimental for MD simulations and high-throughput computations where computational efficiency is as vital as the potential's accuracy itself.

The accuracy and speed of MLIPs typically present a trade-off in the optimization process without a single optimal solution.[73,74] Therefore, it is necessary to select from a range of alternatives to balance accuracy and efficiency. As shown in Fig. 3, by performing a grid search on the hyperparameters of ACE including the cutoff radius, order of expansion, and number of basis functions, a series of potentials can be trained, offering a spectrum of efficiency and precision pairs. The vertices of their convex hull form the Pareto optimal,[75] signifying that any improvement in accuracy will inevitably accompany an increase in the computational cost. These Pareto optimal points provide optimal solutions for hyperparameters' selection.

In the thermal calculation practice, the physical accuracy of MLIPs requires further consideration beyond mere numerical precision. Employing these Pareto optimal points, properties such as cell length, phonon frequency, and thermal conductivity of crystalline silicon are predicted, as shown in Figs. 3(b)–3(d). Convergence of these properties' calculations is observed when the computational cost surpasses 8×10^{-3} ms/timestep/atom, with the force root-mean-square error (RMSE) falling below 50 meV/Å. This convergence indicates that the potential's accuracy has met the computational requirements. Therefore, the potential marked on the dashed line can satisfy the computational needs with a balance of performance and efficiency. This convergence behavior of Pareto optimal potentials can assist in the selection of hyperparameters.

III. EXAMPLES FOR APPLICATIONS

State-of-the-art MLIP models, including the ACE potential, offer an interface to the ASE or LAMMPS (as listed in Table I), allowing convenient LD calculations and MD simulations. In the following, we will demonstrate how the ACE is used to calculate the thermal conductivity of crystalline silicon and amorphous carbon. All the data, codes, and scripts necessary to reproduce the results can be accessed at https://github.com/Dio2k/Tutorial-ACE4LDSim.

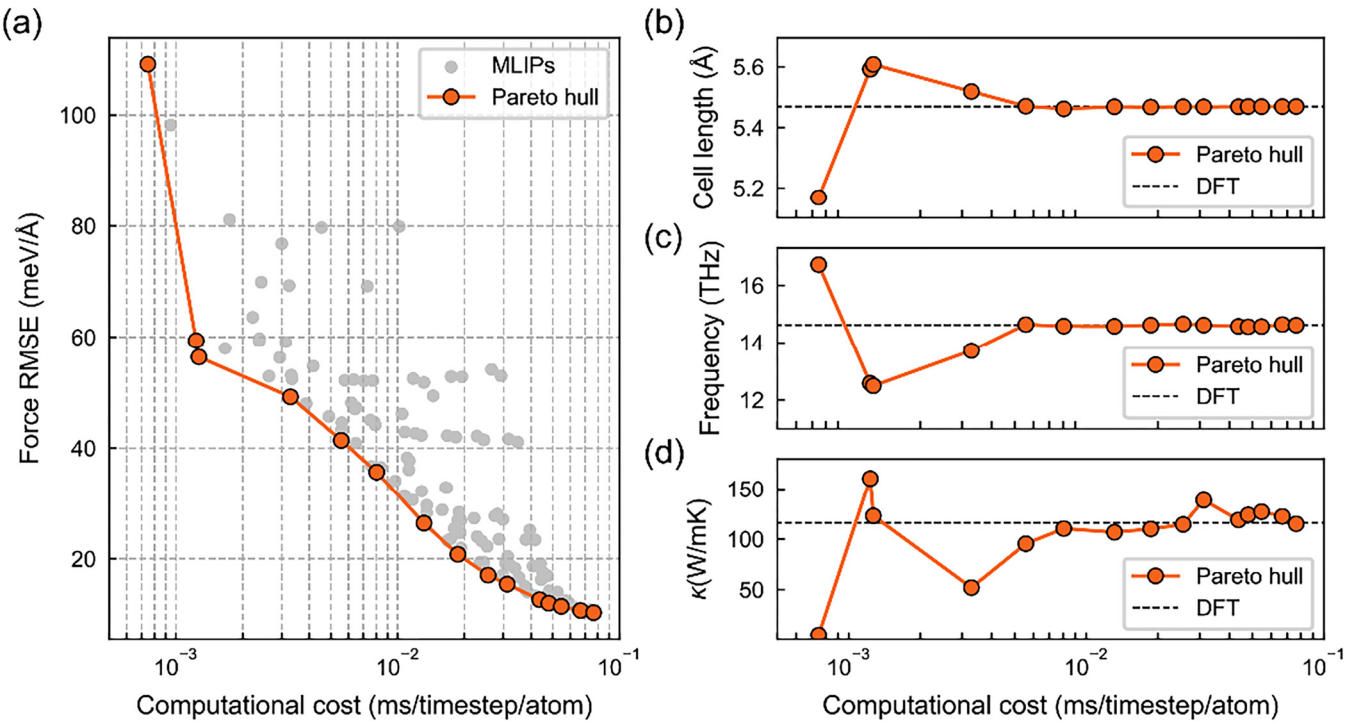

FIG. 3. (a) Pareto optimal via grid search. (b) and (d) Physical accuracy exemplification of cell length, phonon cutoff frequency, and thermal conductivity, respectively, on Pareto optimal points.

A. Diamond silicon

LD is a fundamental approach for normal modes, i.e., phonons in crystal, the quantized units of vibrational energy that serve as the dominating heat carriers in insulators and semiconductors. Harmonic LD calculation takes second FCs to generate phonon dispersion, group velocities, and other harmonic or quasi-harmonic phonon properties, while anharmonic LD calculation takes third or higher-order FCs to generate phonon scattering rates, which can provide scattering rates. With these phonon properties, thermal conductivities can be calculated by solving PBTE. Here, we show the workflow with Phonopy and Phono3py for LD calculations and PBTE solving,[43,76] as illustrated in Fig. 4, where MLIPs can replace DFT in structural relaxation and the generation of FCs to reduce the computational cost. We will exemplify this process using the calculation of crystalline silicon.

While there have already been several publicly available datasets for crystalline silicon, we elect to construct a new dataset from the ground up for pedagogical purposes. Here, we employ hand-built approaches, incorporating both perturbation and AIMD methods for data construction. First, we generate 300 structures by introducing random displacements with a deviation of 0.03 Å on atomic positions and simultaneously applying cell strains ranging from −2% to +2% for each degree of freedom. This process is conducted using a $2 \times 2 \times 2$ supercell of crystal silicon comprising 64 atoms. These reference configurations are then subjected to single-point DFT calculations using the Perdew–Burke–Ernzerhof functional with the generalized gradient approximation,[77] a cutoff energy of 600 eV, and a k-point mesh of $4 \times 4 \times 4$ in Vienna *Ab initio* Simulation Package (VASP).[78,79] Second, we perform ten 10 ps AIMD simulations at a temperature range from 100 to 1000 K in 100 K increments, using a supercell of the same size and a time step of 10 fs. Snapshots are sampled every 0.1 ps, yielding other 1000 data. In total, we obtain 1300 reference configurations for our dataset.

After obtaining the datasets, the ACE potential can be fit by utilizing pacemaker[34,46,47] software. For parameterization, the site energy is directly read out from a single atom in a linear form, and the cutoff distance (r_{cut}) is set to 7.0 Å. A total of 700 basis functions are parameterized with a maximum body order corresponding to $\nu = 6$. Ten percent of the total dataset is randomly selected for testing, not involved in the training procedure. Parameter optimization results in a fit with an energy mean-averaged error (MAE) of 2.82 meV/atom and a force root-mean-square error (RMSE) of 12.24 meV/Å on the testing datasets. Note that the RMSE metric is used for force since it is sensitive to undesirable outliers. This provides a more stringent estimation of the model's performance in predicting forces.[80]

We proceed to demonstrate the application of the trained ACE potential to thermal calculations. Upon completion of the ACE potential training, a YAML file recording all model parameters is automatically created in the training directory. This file can then be

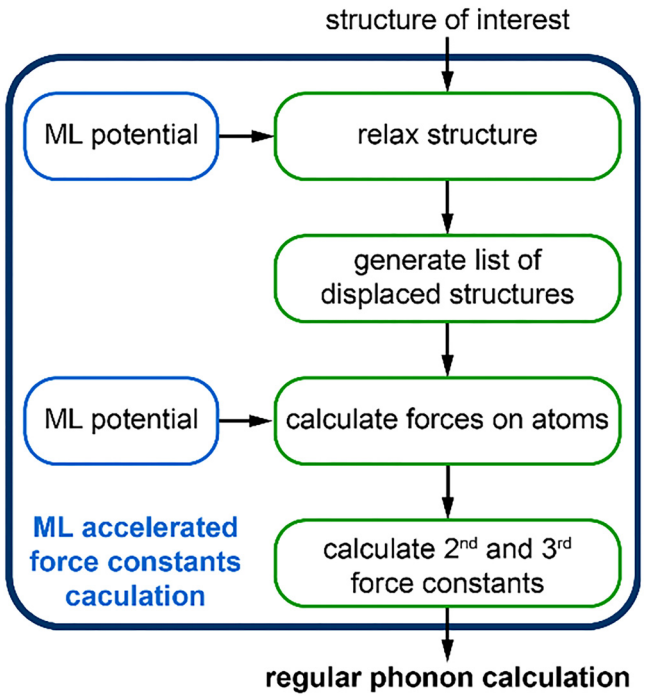

FIG. 4. Workflow of ML accelerated force constant calculations in lattice dynamics.

used as an ASE calculator, enabling the convenient prediction of structural energies and atomic forces (this interface is similar to the other Python-supported MLIPs listed in Table I). Listing 1 shows an illustrative code extract to perform structural relaxation with the ACE potential in Python, where the PreconLBFGS optimizer[81] is employed. The variability of the cell shape and volume during optimization (variable_cell) and the force convergence criterion (fmax) can be further specified. The optimized equilibrium structure is a prerequisite for the subsequent force constant calculations.

Listing 1. Code extract of structural optima with the ACE potential.

```
from pyace import PyACECalculator
ase_atoms.calc = PyACECalculator('/path/to/potential')
from ase.optimize.precon import PreconLBFGS
optimizer = PreconLBFGS(ase_atoms, variable_cell = True)
optimizer.run(fmax = 1e-5)
```

For FC calculations, a practical approach is to apply finite displacements to atoms within a given supercell.[43] With a sufficient number of atomic displacement and force pairs, FCs can be approximated mathematically. This can be achieved with the help of the phono3py package as shown in Listing 2. In the first block, calculation settings are specified, which include the size of the supercell (supercell_matrix), the finite displacement distance applied to atoms (distance), and the cutoff distance on displacement pairs to reduce computational budget (cutoff_pair_distance). In the second block, forces on the displaced structures are predicted with the ASE calculator iteratively, and then, FCs are estimated. Here, only second and third FCs are considered.

Listing 2. Code extract of force constant calculations with the ACE potential.

```
from phono3py import Phono3py
phono3py = Phono3py(phpy_atoms, supercell_matrix = [3, 3, 3])
phono3py.generate_displacements(distance = 0.03, cutoff_pair_distance =
6.21)
# Adapted to ASE.Atoms formats for ML potentials calculator.
supercell_fc2 = PhonopyAtoms2AseAtoms(phono3py.phonon_supercell)
supercell_fc2.calc = PyACECalculator('/path/to/potential')
for dataset in phono3py.phonon_dataset['first_atoms']:
        supercell_fc2.positions[dataset['number']] + = dataset
['displacement']
        dataset['forces'] = supercell_fc2.get_forces()
        supercell_fc2.positions[dataset['number']] — = dataset
['displacement']
phono3py.produce_fc2()
fc2 = phono3py.fc2
# the third force constant can be obtained with a similar process.
fc3 = …
```

With the FCs prepared using the developed ACE model, we proceed with regular phonon computations to derive the thermal conductivity of crystalline silicon. Figure 5 presents the calculated phonon band structure, density of states (DOS), and temperature-dependent thermal conductivity, with comparisons to the DFT results under the same calculation settings. It is evident that the LD computations accelerated by MLIP align with DFT in high agreement, indicating the capability of MLIPs to predict thermal transport properties with comparable accuracy to DFT.

1. Amorphous carbon

Amorphous materials are valuable for applications such as thermal barrier coatings and thermoelectric, where their inherently low thermal conductivity due to disorder is desired.[82] However, the complex local environments and interactions in amorphous materials require large atomic structures, which pose significant challenges for *ab initio* calculations, particularly in thermal computations.[13,83] To address this, MLIPs can accurately capture the complex interatomic interactions in amorphous systems, allowing for reliable atomic model generation and force constant calculations. In this case, we use amorphous carbon (a-C) to exemplify the advantage of MLIPs in handling thermal transport calculations for complex systems. As shown in Fig. 6, the melt-quench protocol, structure relaxation, and FC calculations for further thermal transport modeling can all be accelerated by MLIPs.

Since the purpose of the target MLIP is to correctly simulate the melt-quench process and calculate the FCs required for thermal conductivity calculations, the training dataset must cover a wide range of configurations as comprehensively as possible. Manually creating a reasonable dataset in a convincing manner is challenging. Therefore, we opt to utilize the QbC technique from active learning to construct the dataset instead of using hand-built approaches.

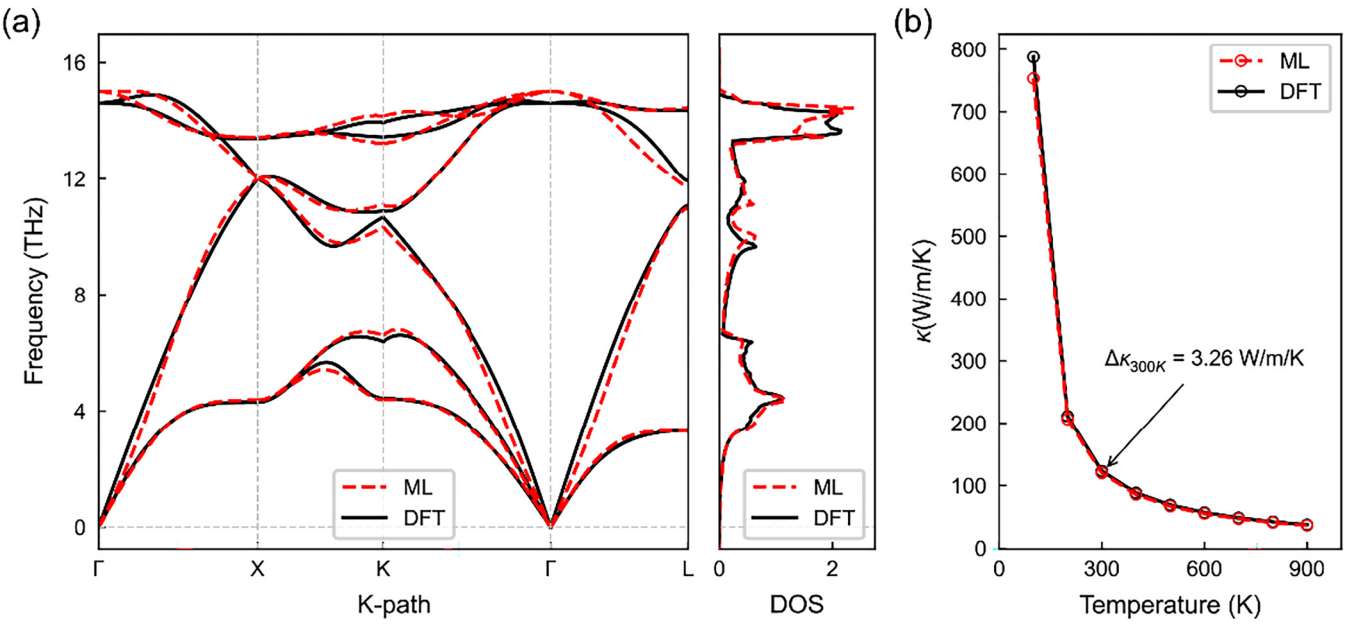

FIG. 5. Comparison of DFT and ML predictions on (a) band structure and DOS, and (b) thermal conductivity.

First, a small initial dataset is built by 500 structures sampled from an AIMD simulation performed using the VASP with the Perdew–Burke–Ernzerhof exchange-correlation functional under the generalized gradient approximation used along with projector-augmented wave (PAW) pseudo-potentials.[77,84,85] The simulation starts with a (unstable) simple-cubic lattice of carbon containing 125 atoms with density of 2.5 g/cm^3. The system is first held at 8000 K for 3 ps, then maintained in the liquid state at 5000 K for another 3 ps, quenched to 300 K at a constant rate of over 1 ps, and finally, annealed at 300 K for 3 ps. The time step is set to 1 fs along with a k-point grid of $3 \times 3 \times 3$ in all simulations. Second, we use a committee of eight potentials to iteratively select new snapshots to enrich the initial dataset. For each iteration, 50 unselected structures with the highest uncertainties are added to the dataset, and the protocol stops after 10 iterations. This procedure ultimately yields a dataset comprising 1000 structures for training the final ACE potential.

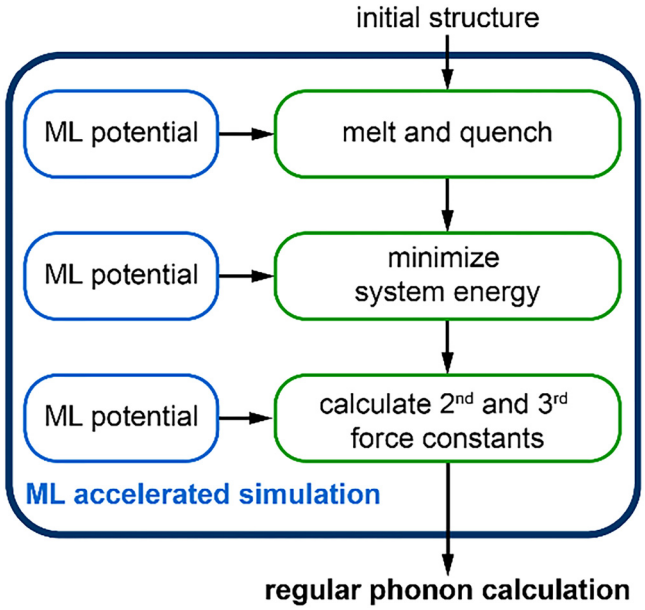

FIG. 6. Workflow of ML accelerated LD simulation on amorphous materials.

For potential parameterization, the site energy is directly read out from two single atomic properties in the Finnis Sinclair form and the cutoff distance is set as 5.5 Å. A total of 700 basis functions are parameterized with a maximum body order corresponding to $\nu = 6$. Ten percent of the total dataset is randomly selected for testing. Parameter optimization leads to a fit with an energy MAE of 40.03 meV/atom and a force RMSE of 540.07 meV/Å for the testing datasets.

Listing 3. Code extract of ACE potential definition in the LAMMPS.

```
pair_style pace
pair_coeff * * /path/to/potential
```

Leveraging the speed advantage of the ACE potential, we generate an a-C sample using a larger cell containing more atoms and a longer melt-quench process.[86] This MD simulation is performed

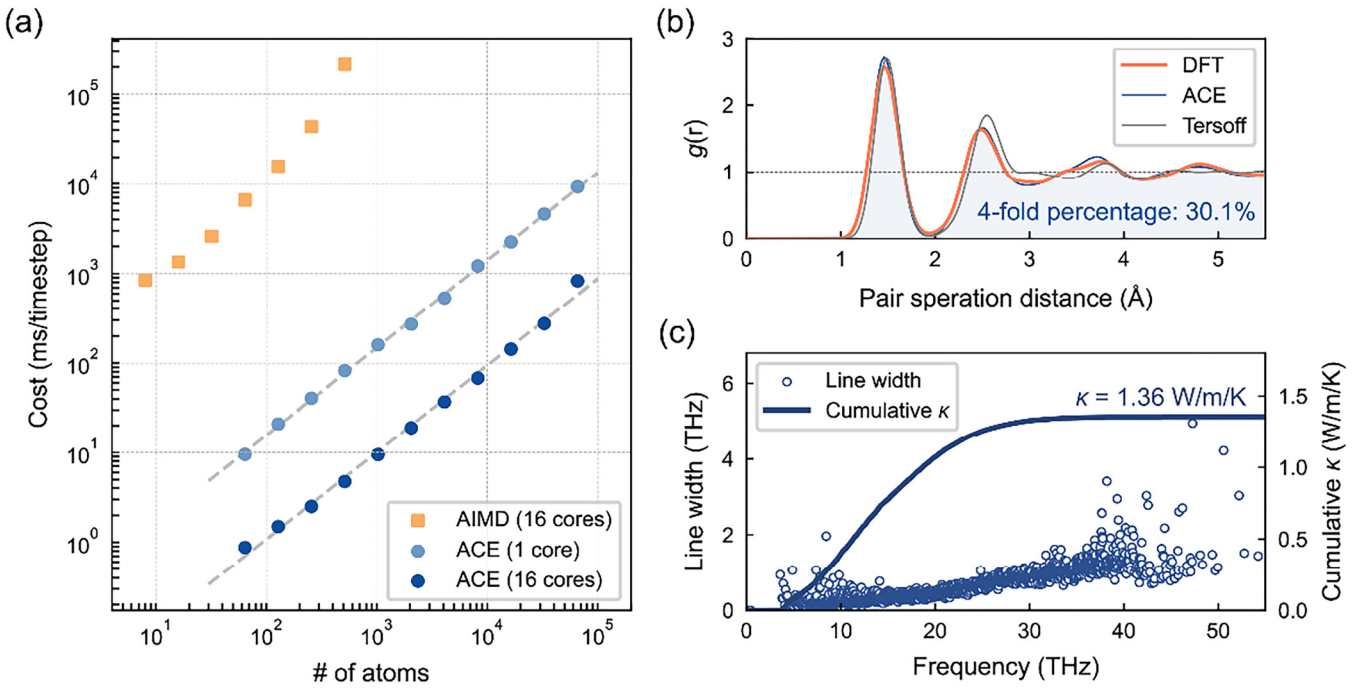

FIG. 7. (a) Comparison of computation efficiency on AIMD and MLIP MD. (b) RDF of a-C. (c) Scattering rate distribution and cumulative thermal conductivity of a-C.

in the LAMMPS, with the implementation of ACE potential shown in Listing 3. Using the same time step of 1 fs, we start with a 512-atom orthogonal box held at 8000 K and then cooled down to 5000 K for 100 ps. We then perform a rapid quench from 5000 to 300 K within 1 ps. Finally, the system is equilibrated at 300 K for 100 ps, followed by an energy minimization process for the equilibrium structure. We compare the radial distribution function (RDF) of the final generated structure with those from AIMD and the Tersoff potential.[87] As shown in Fig. 7(b), the RDF of ACE shows good agreement with the result of AIMD simulation, whereas the Tersoff potential does not capture the right locations and values of the second and the third RDF peaks. We also calculate the fourfold coordinated atom percentage to be 30.1%, which aligns with the previous works reported in Ref. 88. Additionally, we assess the computational efficiency of the ACE potential. As illustrated in Fig. 7(a), simulations accelerated by MLIPs show a speed of 3 to 4 orders of magnitude faster than AIMD, with linear scaling relative to the system size and the number of CPU cores used in the calculation. In contrast, the computational cost of AIMD grows exponentially with the number of atoms, leading to a sharp increase in computational expense, which restricts its application to large systems (e.g., over 10^3 atoms).

With the relaxed atomic model, we calculate the second and third FCs and, subsequently, the thermal conductivity for a-C. The FCs are derived using the PHONON package[89] in the LAMMPS. For thermal conductivity, we use the Quasi-harmonic Green's Function (QHGK) method[90] implemented in kALDo,[44] versatile and scalable open-source software to compute phonon transport in crystalline and amorphous solids. As illustrated in Fig. 7(c), the thermal conductivity of a-C is 1.36 W/m/K, which is close to the value of 1.61 W/m/K using the linear fit to experimental data reported in Ref. 91, demonstrating the capability of MLIPs in thermal calculations for complex systems. We noted that the propagation contribution is likely to be underestimated due to the limitation of the QHGK method and a proper extrapolation is needed for the case in which propagation mechanism matters,[83] like a-C with higher density[92] and amorphous silicon.[93]

IV. SUMMARY

In this tutorial, we offer a comprehensive guide to develop MLIPs for modeling thermal transport using the LD method. This guide covers the introduction of basic concepts and the procedure of MLIP development from data construction, potential training to the optimal and the selection of model hyperparameters. We also present two application examples, i.e., crystalline silicon and amorphous carbon, to demonstrate the superiority of speed with the accuracy of MLIPs. In addition, we provide practical code implementations for beginners to follow conveniently. It is noted that while we use ACE potential as the illustrative backbone model in this tutorial, the underlying principle, practical strategies, and guidelines can be easily transferred to the other MLIP models. We believe that our tutorial will promote the extensive applications of MLIPs in thermal transport research, encouraging innovative discoveries and their practical applications in the industry.

ACKNOWLEDGMENTS

This study was financially supported by the National Natural Science Foundation of China (NNSFC) (Nos. 52425601, 52327809, 52250273, U20A20301, and 82361138571), National Key Research and Development Program of China (No. 2023YFB4404104), and Beijing Natural Science Foundation (No. L233022).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Liben Guo: Conceptualization (equal); Investigation (lead); Validation (equal); Visualization (lead); Writing – original draft (lead); Writing – review & editing (equal). **Yuanbin Liu:** Conceptualization (lead); Investigation (equal); Validation (equal); Visualization (equal); Writing – original draft (supporting); Writing – review & editing (equal). **Lei Yang:** Conceptualization (equal); Investigation (equal); Validation (equal); Visualization (supporting); Writing – original draft (supporting); Writing – review & editing (equal). **Bingyang Cao:** Conceptualization (equal); Funding acquisition (lead); Investigation (equal); Supervision (lead); Validation (equal); Visualization (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are openly available at in Github repository Tutorial-ACE4LDSim at https://github.com/Dio2k/Tutorial-ACE4LDSim, Ref. 94.

REFERENCES

[1]Y.-C. Hua, H.-L. Li, and B.-Y. Cao, "Thermal spreading resistance in ballistic-diffusive regime for GaN HEMTs," IEEE Trans. Electron Devices **66**(8), 3296–3301 (2019).
[2]Y. Shen, X.-S. Chen, Y.-C. Hua, H.-L. Li, L. Wei, and B.-Y. Cao, "Bias dependence of non-Fourier heat spreading in GaN HEMTs," IEEE Trans. Electron Devices **70**(2), 409–417 (2023).
[3]Q. Yan and M. G. Kanatzidis, "High-performance thermoelectrics and challenges for practical devices," Nat. Mater. **21**(5), 503–513 (2022).
[4]H. Wu, H. Fan, and Y. Hu, "*Ab initio* determination of ultrahigh thermal conductivity in ternary compounds," Phys. Rev. B **103**(4), L041203 (2021).
[5]Z. Liu, X. Wu, V. Varshney, J. Lee, G. Qin, M. Hu, A. K. Roy, and T. Luo, "Bond saturation significantly enhances thermal energy transport in two-dimensional pentagonal materials," Nano Energy **45**, 1–9 (2018).
[6]S.-Y. Yue, G. Qin, X. Zhang, X. Sheng, G. Su, and M. Hu, "Thermal transport in novel carbon allotropes with sp^2 or sp^3 hybridization: An *ab initio* study," Phys. Rev. B **95**(8), 085207 (2017).
[7]G. Fugallo, A. Cepellotti, L. Paulatto, M. Lazzeri, N. Marzari, and F. Mauri, "Thermal conductivity of graphene and graphite: Collective excitations and mean free paths," Nano Lett. **14**(11), 6109–6114 (2014).
[8]J. E. Turney, E. S. Landry, A. J. H. McGaughey, and C. H. Amon, "Predicting phonon properties and thermal conductivity from anharmonic lattice dynamics calculations and molecular dynamics simulations," Phys. Rev. B **79**(6), 064301 (2009).
[9]D. A. Broido, M. Malorny, G. Birner, N. Mingo, and D. A. Stewart, "Intrinsic lattice thermal conductivity of semiconductors from first principles," Appl. Phys. Lett. **91**(23), 231922 (2007).
[10]P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," Phys. Rev. **136**(3B), B864–B871 (1964).
[11]W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects," Phys. Rev. **140**(4A), A1133–A1138 (1965).
[12]T. Feng and X. Ruan, "Quantum mechanical prediction of four-phonon scattering rates and reduced thermal conductivity of solids," Phys. Rev. B **93**(4), 045202 (2016).
[13]M. Simoncelli, F. Mauri, and N. Marzari, "Thermal conductivity of glasses: First-principles theory and applications," npj Comput. Mater. **9**(1), 1–22 (2023).
[14]E. A. Carter, "Challenges in modeling materials properties without experimental input," Science **321**(5890), 800–803 (2008).
[15]L. Hung and E. A. Carter, "Accurate simulations of metals at the mesoscale: Explicit treatment of 1 million atoms with quantum mechanics," Chem. Phys. Lett. **475**(4–6), 163–170 (2009).
[16]V. L. Deringer, M. A. Caro, and G. Csányi, "Machine learning interatomic potentials as emerging tools for materials science," Adv. Mater. **31**(46), 1902765 (2019).
[17]P. Friederich, F. Häse, J. Proppe, and A. Aspuru-Guzik, "Machine-learned potentials for next-generation matter simulations," Nat. Mater. **20**(6), 750–761 (2021).
[18]G. Wang, C. Wang, X. Zhang, Z. Li, J. Zhou, and Z. Sun, "Machine learning interatomic potential: Bridge the gap between small-scale models and realistic device-scale simulations," iScience **27**(5), 109673 (2024).
[19]Y. Zhou, W. Zhang, E. Ma, and V. L. Deringer, "Device-scale atomistic modelling of phase-change memory materials," Nat. Electron **6**(10), 746–754 (2023).
[20]A. Musaelian, S. Batzner, A. Johansson, L. Sun, C. J. Owen, M. Kornbluth, and B. Kozinsky, "Learning local equivariant representations for large-scale atomistic dynamics," Nat. Commun. **14**(1), 579 (2023).
[21]J. Tiwari and T. Feng, "Accurate prediction of thermal conductivity of Al_2O_3 at ultrahigh temperatures," Phys. Rev. B **109**(7), 075201 (2024).
[22]X. Yang, Y.-S. Chen, Y.-H. Zheng, C.-W. Wu, G.-F. Xie, Y.-J. Zeng, and W.-X. Zhou, "Understanding the importance of four-phonon scattering in low-symmetry monolayer 1T′-ReS_2 using machine learning potential," Appl. Phys. Lett. **124**(7), 072204 (2024).
[23]Y. Wang, W. Huang, J. Che, and X. Wang, "Four-phonon scattering significantly reduces the predicted lattice thermal conductivity in penta-graphene: A machine learning-assisted investigation," Comput. Mater. Sci. **229**, 112435 (2023).
[24]J. Tang, G. Li, Q. Wang, J. Zheng, L. Cheng, and R. Guo, "Effect of four-phonon scattering on anisotropic thermal transport in bulk hexagonal boron nitride by machine learning interatomic potential," Int. J. Heat Mass Transfer **207**, 124011 (2023).
[25]G. Yang, Y.-B. Liu, L. Yang, and B.-Y. Cao, "Machine-learned atomic cluster expansion potentials for fast and quantum-accurate thermal simulations of wurtzite AlN," J. Appl. Phys. **135**(8), 085105 (2024).
[26]Z. Fan, J. Ding, and E. Ma, "Machine learning bridges local static structure with multiple properties in metallic glasses," Mater. Today **40**, 48–62 (2020).
[27]G. C. Sosso, D. Donadio, S. Caravati, J. Behler, and M. Bernasconi, "Thermal transport in phase-change materials from atomistic simulations," Phys. Rev. B **86**(10), 104301 (2012).
[28]D. Campi, D. Donadio, G. C. Sosso, J. Behler, and M. Bernasconi, "Electron-phonon interaction and thermal boundary resistance at the crystal-amorphous interface of the phase change compound GeTe," J. Appl. Phys. **117**(1), 015304 (2015).
[29]C. Mangold, S. Chen, G. Barbalinardo, J. Behler, P. Pochet, K. Termentzidis, Y. Han, L. Chaput, D. Lacroix, and D. Donadio, "Transferability of neural network potentials for varying stoichiometry: Phonons and thermal conductivity of Mn_xGe_y compounds," J. Appl. Phys. **127**, 244901 (2020).
[30]Y.-B. Liu, J.-Y. Yang, G.-M. Xin, L.-H. Liu, G. Csányi, and B.-Y. Cao, "Machine learning interatomic potential developed for molecular simulations on thermal properties of β-Ga_2O_3," J. Chem. Phys. **153**(14), 144501 (2020).

[31]A. Hjorth Larsen, J. Jørgen Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. Bjerre Jensen, J. Kermode, J. R. Kitchin, E. Leonhard Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. Bergmann Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen, "The atomic simulation environment—A Python library for working with atoms," J. Phys.: Condens. Matter **29**(27), 273002 (2017).
[32]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).
[33]A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton, "LAMMPS—A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales," Comput. Phys. Commun. **271**, 108171 (2022).
[34]R. Drautz, "Atomic cluster expansion for accurate and transferable interatomic potentials," Phys. Rev. B **99**(1), 014104 (2019).
[35]A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, "Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons," Phys. Rev. Lett. **104**, 136403 (2010).
[36]Z. Fan, W. Chen, V. Vierimaa, and A. Harju, "Efficient molecular dynamics simulations with many-body potentials on graphics processing units," Comput. Phys. Commun. **218**, 10–16 (2017).
[37]Z. Fan, Z. Zeng, C. Zhang, Y. Wang, K. Song, H. Dong, Y. Chen, and T. Ala-Nissila, "Neuroevolution machine learning potentials: Combining high accuracy and low cost in atomistic simulations and application to heat transport," Phys. Rev. B **104**(10), 104309 (2021).
[38]J. Han, L. Zhang, R. Car, and W. E, "Deep potential: A general representation of a many-body potential energy surface," Commun. Comput. Phys. **23**(3), 629–639 (2018).
[39]I. Batatia, D. P. Kovács, G. N. C. Simm, C. Ortner, and G. Csányi, "MACE: Higher order equivariant message passing neural networks for fast and accurate force fields," in *Advances in Neural Information Processing Systems*, edited by S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (MIT Press, 2022), Vol. 35.
[40]S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, "E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials," Nat. Commun. **13**(1), 2453 (2022).
[41]H. Dong, Y. Shi, P. Ying, K. Xu, T. Liang, Y. Wang, Z. Zeng, X. Wu, W. Zhou, S. Xiong, S. Chen, and Z. Fan, "Molecular dynamics simulations of heat transport using machine-learned potentials: A mini-review and tutorial on GPUMD with neuroevolution potentials," J. Appl. Phys. **135**(16), 161101 (2024).
[42]H. Bao, J. Chen, X. Gu, and B. Cao, "A review of simulation methods in micro/nanoscale heat conduction," ES Energy Environ. **1**, 16–55 (2018).
[43]A. Togo, "First-principles phonon calculations with phonopy and Phono3py," J. Phys. Soc. Jpn. **92**(1), 012001 (2023).
[44]G. Barbalinardo, Z. Chen, N. W. Lundgren, and D. Donadio, "Efficient anharmonic lattice dynamics calculations of thermal transport in crystalline and disordered solids," J. Appl. Phys. **128**(13), 135104 (2020).
[45]A. J. H. McGaughey, A. Jain, H.-Y. Kim, and B. Fu, "Phonon properties and thermal conductivity from first principles, lattice dynamics, and the Boltzmann transport equation," J. Appl. Phys. **125**(1), 011101 (2019).
[46]Y. Lysogorskiy, C. van der Oord, A. Bochkarev, S. Menon, M. Rinaldi, T. Hammerschmidt, M. Mrovec, A. Thompson, G. Csányi, C. Ortner, and R. Drautz, "Performant implementation of the atomic cluster expansion (PACE) and application to copper and silicon," npj Comput. Mater. **7**(1), 97 (2021).
[47]A. Bochkarev, Y. Lysogorskiy, S. Menon, M. Qamar, M. Mrovec, and R. Drautz, "Efficient parametrization of the atomic cluster expansion," Phys. Rev. Mater. **6**(1), 013804 (2022).
[48]A. P. Bartók, J. Kermode, N. Bernstein, and G. Csányi, "Machine learning a general-purpose interatomic potential for silicon," Phys. Rev. X **8**(4), 041048 (2018).
[49]P. Rowe, V. L. Deringer, P. Gasparotto, G. Csányi, and A. Michaelides, "An accurate and transferable machine learning potential for carbon," J. Chem. Phys. **153**(3), 034702 (2020).
[50]M. Qamar, M. Mrovec, Y. Lysogorskiy, A. Bochkarev, and R. Drautz, "Atomic cluster expansion for quantum-accurate large-scale simulations of carbon," J. Chem. Theory Comput. **19**(15), 5151–5167 (2023).
[51]W. J. Szlachta, A. P. Bartók, and G. Csányi, "Accuracy and transferability of Gaussian approximation potential models for tungsten," Phys. Rev. B **90**, 104108 (2014).
[52]X. Wu, W. Zhou, H. Dong, P. Ying, Y. Wang, B. Song, Z. Fan, and S. Xiong, "Correcting force error-induced underestimation of lattice thermal conductivity in machine learning molecular dynamics," J. Chem. Phys. **161**(1), 014103 (2024).
[53]D. G. Truhlar, "*Ab initio* molecular dynamics: Basic theory and advanced methods," Phys. Today **63**(3), 54–56 (2010).
[54]F. Eriksson, E. Fransson, and P. Erhart, "The hiPhive package for the extraction of high-order force constants by machine learning," Adv. Theory Simul. **2**(5), 1800184 (2019).
[55]D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," J. Artif. Intell. Res. **4**, 129–145 (1996).
[56]J. Vandermause, S. B. Torrisi, S. Batzner, Y. Xie, L. Sun, A. M. Kolpak, and B. Kozinsky, "On-the-fly active learning of interpretable Bayesian force fields for atomistic rare events," npj Comput. Mater. **6**(1), 1–11 (2020).
[57]Y. Zhai, A. Caruso, S. Gao, and F. Paesani, "Active learning of many-body configuration space: Application to the Cs^+–water MB-nrg potential energy function as a case study," J. Chem. Phys. **152**(14), 144103 (2020).
[58]C. van der Oord, M. Sachs, D. P. Kovács, C. Ortner, and G. Csányi, "Hyperactive learning for data-driven interatomic potentials," npj Comput. Mater. **9**(1), 1–14 (2023).
[59]D. D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *Proceedings of the Eleventh International Conference*, edited by W. W. Cohen and H. Hirsh (Morgan Kaufmann, 1994), pp. 148–156.
[60]H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proceedings of the Fifth Annual ACM Conference on Computational Learning Theory*, edited by D. Haussler (ACM, 1992), pp. 287–294.
[61]A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation, and active learning," in *Advances in Neural Information Processing Systems*, edited by G. Tesauro, D. S. Touretzky, and T. K. Leen (MIT Press, 1994), Vol. 7, pp. 231–238.
[62]C. Schran, K. Brezina, and O. Marsalek, "Committee neural network potentials control generalization errors and enable active learning," J. Chem. Phys. **153**(10), 104105 (2020).
[63]C. Schran, F. L. Thiemann, P. Rowe, E. A. Müller, O. Marsalek, and A. Michaelides, "Machine learning potentials for complex aqueous systems made simple," Proc. Natl. Acad. Sci. **118**(38), e2110077118 (2021).
[64]V. L. Deringer, A. P. Bartók, N. Bernstein, D. M. Wilkins, M. Ceriotti, and G. Csányi, "Gaussian process regression for materials and molecules," Chem. Rev. **121**(16), 10073–10141 (2021).
[65]M. Karabin and D. Perez, "An entropy-maximization approach to automated training set generation for interatomic potentials," J. Chem. Phys. **153**(9), 094110 (2020).
[66]M. Wen and E. B. Tadmor, "Uncertainty quantification in molecular simulations with dropout neural network potentials," npj Comput. Mater. **6**(1), 124 (2020).
[67]J. Kukacka, V. Golkov, and D. Cremers, "Regularization for deep learning: A taxonomy," CoRR abs/1710.10686 (2017).
[68]Y. Tian and Y. Zhang, "A comprehensive survey on regularization strategies in machine learning," Inf. Fusion **80**, 146–166 (2022).
[69]R. Tibshirani, "Regression shrinkage and selection via the lasso," J. R. Stat. Soc. Ser. B: Stat. Methodol. **58**(1), 267–288 (1996).
[70]D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Third International Conference on Learning Representations*, edited by Y. Bengio and Y. LeCun (OpenReview, 2015).

[71]J. Nocedal, "Updating quasi-newton matrices with limited storage," Math. Comput. **35**, 773–782 (1980).
[72]D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," Math. Program. **45**(1–3), 503–528 (1989).
[73]Y. Zuo, C. Chen, X. Li, Z. Deng, Y. Chen, J. Behler, G. Csányi, A. V. Shapeev, A. P. Thompson, M. A. Wood, and S. P. Ong, "Performance and cost assessment of machine learning interatomic potentials," J. Phys. Chem. A **124**(4), 731–745 (2020).
[74]A. Seko, A. Togo, and I. Tanaka, "Group-theoretical high-order rotational invariants for structural representations: Application to linearized machine learning interatomic potential," Phys. Rev. B **99**(21), 214108 (2019).
[75]J. Branke, K. Deb, K. Miettinen, and R. Słowiński, in *Multiobjective Optimization: Interactive and Evolutionary Approaches* (Springer-Verlag, Berlin, 2008).
[76]A. Togo, L. Chaput, T. Tadano, and I. Tanaka, "Implementation strategies in phonopy and phono3py," J. Phys.: Condens. Matter **35**(35), 353001 (2023).
[77]J. P. Perdew, K. Burke, and M. Ernzerhof, "Generalized gradient approximation made simple," Phys. Rev. Lett. **77**(18), 3865–3868 (1996).
[78]G. Kresse and J. Furthmüller, "Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set," Comput. Mater. Sci. **6**(1), 15–50 (1996).
[79]G. Kresse and J. Furthmüller, "Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set," Phys. Rev. B **54**(16), 11169–11186 (1996).
[80]J. D. Morrow, J. L. A. Gardner, and V. L. Deringer, "How to validate machine-learned interatomic potentials," J. Chem. Phys. **158**(12), 121501 (2023).
[81]D. Packwood, J. Kermode, L. Mones, N. Bernstein, J. Woolley, N. Gould, C. Ortner, and G. Csányi, "A universal preconditioner for simulating condensed phase materials," J. Chem. Phys. **144**(16), 164109 (2016).
[82]W.-X. Zhou, Y. Cheng, K.-Q. Chen, G. Xie, T. Wang, and G. Zhang, "Thermal conductivity of amorphous materials," Adv. Funct. Mater. **30**(8), 1903829 (2020).
[83]A. Fiorentino, P. Pegolo, and S. Baroni, "Hydrodynamic finite-size scaling of the thermal conductivity in glasses," npj Comput. Mater. **9**(1), 157 (2023).
[84]P. E. Blöchl, "Projector augmented-wave method," Phys. Rev. B **50**(24), 17953–17979 (1994).
[85]G. Kresse and D. Joubert, "From ultrasoft pseudopotentials to the projector augmented-wave method," Phys. Rev. B **59**(3), 1758–1775 (1999).
[86]B. A. Steele, S. Bastea, and I.-F. W. Kuo, "*Ab initio* structural dynamics of pure and nitrogen-containing amorphous carbon," Sci. Rep. **13**(1), 19657 (2023).
[87]J. Tersoff, "Empirical interatomic potential for carbon, with applications to amorphous carbon," Phys. Rev. Lett. **61**(25), 2879–2882 (1988).
[88]Y. Wang, Z. Fan, P. Qian, M. A. Caro, and T. Ala-Nissila, "Density dependence of thermal conductivity in nanoporous and amorphous carbon with machine-learned molecular dynamics," arXiv:2408.12390 (2024).
[89]L. T. Kong, "Phonon dispersion measured directly from molecular dynamics simulations," Comput. Phys. Commun. **182**(10), 2201–2207 (2011).
[90]L. Isaeva, G. Barbalinardo, D. Donadio, and S. Baroni, "Modeling heat transport in crystals and glasses from a unified lattice-dynamical approach," Nat. Commun. **10**(1), 3853 (2019).
[91]M. Shamsa, W. L. Liu, A. A. Balandin, C. Casiraghi, W. I. Milne, and A. C. Ferrari, "Thermal conductivity of diamond-like carbon films," Appl. Phys. Lett. **89**(16), 161921 (2006).
[92]J. Moon and Z. Tian, "Crystal-like thermal transport in amorphous carbon," arXiv:2405.07298 (2024).
[93]J. Moon, B. Latour, and A. J. Minnich, "Propagating elastic vibrations dominate thermal conduction in amorphous silicon," Phys. Rev. B **97**(2), 024201 (2018).
[94]Dataset: "Tutorial-ACE4LDSim," Github (2025). https://github.com/Dio2k/Tutorial-ACE4LDSim