

<https://doi.org/10.1038/s41524-025-01623-4>

Cartesian atomic moment machine learning interatomic potentials

Mingjian Wen¹✉, Wei-Fan Huang², Jin Dai² & Santosh Adhikari²

Machine learning interatomic potentials (MLIPs) have substantially advanced atomistic simulations in materials science and chemistry by balancing accuracy and computational efficiency. While leading MLIPs rely on representing atomic environments using spherical tensors, Cartesian representations offer potential advantages in simplicity and efficiency. Here, we introduce the Cartesian Atomic Moment Potential (CAMP), an approach to building MLIPs entirely in Cartesian space. CAMP constructs atomic moment tensors from neighboring atoms and employs tensor products to incorporate higher body-order interactions, providing a complete description of local atomic environments. Integrated into a graph neural network (GNN) framework, CAMP enables physically motivated, systematically improvable potentials. The model demonstrates excellent performance across diverse systems, including periodic structures, small organic molecules, and two-dimensional materials, achieving accuracy, efficiency, and stability in molecular dynamics simulations that rival or surpass current leading models. CAMP provides a powerful tool for atomistic simulations to accelerate materials understanding and discovery.

Machine learning interatomic potentials (MLIPs) have been widely employed in molecular simulations to investigate the properties of all kinds of materials, ranging from small organic molecules to inorganic crystals and biological systems. MLIPs offer a balance between accuracy and efficiency^{1–4}, making them a suitable choice between first-principles methods such as density functional theory (DFT) and classical interatomic potentials such as the embedded atom method (EAM)⁵.

Since the introduction of the pioneering Behler–Parrinello neural network (BPNN) potential⁶ and the Gaussian approximation potential (GAP)⁷, many valuable MLIPs have been developed^{18–19}. At the same time, techniques have emerged to assess the reliability of MLIP predictions by quantifying uncertainties and evaluating their influence on downstream properties^{20–24}. Despite this progress, the pursuit of enhancing and developing new MLIPs remains ongoing. The current leading MLIPs in terms of accuracy and stability in molecular dynamics (MD) simulations are those based on graph neural networks (GNNs). These models represent an atomic structure as a graph and perform message passing between atoms to propagate information²⁵. The passed messages can be either scalars^{10,12}, vectors¹³, or, more generally, tensors^{16–19}. Using GNNs, a couple of universal MLIPs have recently been developed^{26–32}, which can cover a wide range of materials with chemical species across the periodic table.

At the core of any MLIP lies a description of the atomic environment, which converts the information contained in an atomic neighborhood to a numerical representation that can then be updated and used in an ML

regression algorithm. The representation must satisfy certain symmetries (invariance to permutation of atoms, and invariance or equivariance to translation, rotation, and inversion) inherent to atomic systems. Despite the diverse choice of ML regression algorithms, the atomic environment representation is typically constructed by expanding atomic positions on some basis functions. Early works employ manually crafted basis functions for the bond distances and bond angles, such as the atom-centered symmetry functions used in BPNN⁶ and ANI-1³³. More systematic representations have been developed and adopted in GAP⁷, SNAP³⁴, ACE³⁵, NequIP¹⁷, and MACE¹⁶, among others. In these models, a representation is obtained by first expanding the atomic environment using a radial basis (e.g., Chebyshev polynomials) and the spherical harmonics angular basis to obtain tensorial equivariant features. The equivariant features are then updated and finally contracted to yield the scalar potential energy.

Another approach seeks to develop atomic representations and MLIPs entirely in Cartesian space, bypassing the need for spherical harmonics. This can potentially offer greater simplicity and reduced complexity compared to models using spherical representations. The MTP model by Shapeev³⁶ adopted this approach, which constructs Cartesian moment tensors to capture the angular information of an atomic environment. It has been shown that MTP and ACE are highly related given that a moment tensor can be expressed using spherical harmonics³⁷. TeaNet¹⁸ and TensorNet¹⁹ have explored the use of Cartesian tensors limited to the second rank to create MLIPs. The recent CACE model³⁸ has extended the atomic cluster expansion

¹Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu, China. ²Department of Chemical and Biomolecular Engineering, University of Houston, Houston, TX, USA. ✉e-mail: mjwen@uestc.edu.cn

to Cartesian space and subsequently built MLIPs on this foundation. These works represent excellent efforts toward building MLIPs using a fully Cartesian representation of atomic environments. However, their performance has not yet matched that of models built using spherical tensors.

In this work, we propose the Cartesian Atomic Moment Potential (CAMP), a systematically improvable MLIP developed in Cartesian space under the GNN framework. Inspired by MTP³⁶, CAMP employs physically motivated moment tensors to characterize atomic environments. It then generates hyper moments through tensor products of these moment tensors, effectively capturing higher-order interactions. We have devised specific construction rules for atomic and hyper moment tensors that direct message flow from higher-rank to lower-rank tensors, ultimately to scalars. This approach aligns with our objective of modeling potential energy (a scalar quantity) and significantly reduces the number of moment tensors, enhancing computational efficiency. These moment tensors are subsequently integrated into a message-passing GNN framework, undergoing iterative updates and refinement to form the final CAMP. To evaluate the performance of the model, we have conducted tests across diverse material systems, including periodic LiPS inorganic crystals¹⁷ and bulk water³⁹, non-periodic small organic molecules⁸, and partially periodic two-dimensional (2D) graphene⁴⁰. These comprehensive benchmark studies demonstrate that CAMP is an accurate, efficient, and stable MLIP for molecular simulations.

Results

Model Architecture

CAMP is a GNN model that processes atomic structures, utilizing atomic coordinates, atomic numbers, and cell vectors as input, and predicts the total potential energy, stresses on the structure, and the forces acting on individual atoms.

Figure 1 presents a schematic overview of the model architecture. In the following, we discuss the key components and model design choices.

Atomic graph. An atomic structure is represented as a graph $G(V, E)$ consisting of a set of nodes V and a set of edges E . Each node in V represents an atom, and an edge in E is created between two atoms if they are within a cutoff distance of r_{cut} (Fig. 1a). An atom node i is described by a 3-tuple $(\mathbf{r}^i, z^i, \mathbf{h}^i)$, where \mathbf{r}^i is the position of the atom, z^i is the atomic number, and \mathbf{h}^i is the atom feature. The vector $\mathbf{r}^{ij} = \mathbf{r}^j - \mathbf{r}^i$ associated with the edge from atom i to atom j gives their relative position.

The atom feature \mathbf{h}^i is a set of tensors that carry two indices u and v , becoming \mathbf{h}_{uv}^i in its full form, where u and v denote the *channel* and *rank* of each feature tensor, respectively. The use of channels allows for multiple

copies of tensors of the same rank, helping to increase the expressiveness of the feature.

The atom features are mixed across different channels using a linear mapping:

$$\mathbf{h}_{uv}^i = \sum_{u'} W_{uu'} \mathbf{h}_{u'v}^i, \quad (1)$$

where $W_{uu'}$ are trainable parameters.

Radial basis. The edge length $r^{ij} = \|\mathbf{r}^{ij}\|$ is expanded using a set of radial basis functions, $B_u(r^{ij}, z^i, z^j)$, indexed by the channel u . Following MTP^{36,41}, each basis function is a linear combination of the Chebyshev polynomials of the first kind Q^β :

$$B_u(r^{ij}, z^i, z^j) = \sum_{\beta=0}^{N_\beta} W_{uz^iz^j}^\beta Q^\beta\left(\frac{r^{ij}}{r_{\text{cut}}}\right), \quad (2)$$

where N_β denotes the maximum degree of the Chebyshev polynomials. Separate trainable parameters $W_{uz^iz^j}^\beta$ are used for different atom pairs, enabling customized radial basis dependent on their atomic numbers z^i and z^j . Moreover, the radial basis B_u allows the model to scale to any number of chemical species without increasing the model size.

Angular part. The angular information of an atomic environment is contained in the normalized edge vectors, $\hat{\mathbf{r}} = \mathbf{r}/r$, where we have omitted the atom indices i and j in \mathbf{r}^{ij} and r^{ij} for simplicity. Unlike many existing models that expand it on spherical harmonics, we directly adopt the Cartesian basis. The rank- v *polyadic tensor* of $\hat{\mathbf{r}}$ is constructed as

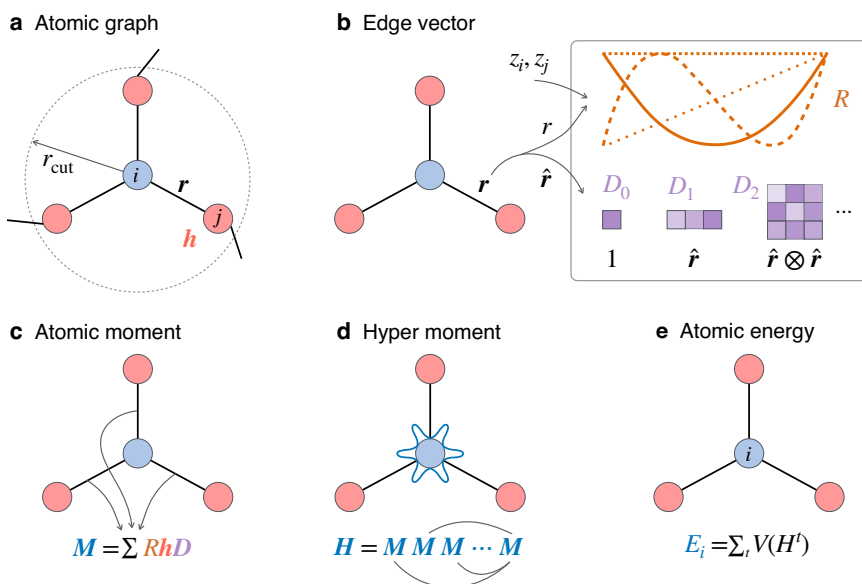
$$\mathbf{D}_v = \hat{\mathbf{r}} \otimes \hat{\mathbf{r}} \otimes \cdots \otimes \hat{\mathbf{r}} \quad (v \text{ of } \hat{\mathbf{r}}), \quad (3)$$

where \otimes denotes a tensor product. For example, $\mathbf{D}_0 = 1$ is a scalar, $\mathbf{D}_1 = \hat{\mathbf{r}} = [\hat{r}_x, \hat{r}_y, \hat{r}_z]$ is a vector, and $\mathbf{D}_2 = \hat{\mathbf{r}} \otimes \hat{\mathbf{r}}$ is a rank-2 tensor, which can be written in a matrix form as

$$\mathbf{D}_2 = \begin{bmatrix} \hat{r}_x^2 & \hat{r}_x \hat{r}_y & \hat{r}_x \hat{r}_z \\ \hat{r}_y \hat{r}_x & \hat{r}_y^2 & \hat{r}_y \hat{r}_z \\ \hat{r}_z \hat{r}_x & \hat{r}_z \hat{r}_y & \hat{r}_z^2 \end{bmatrix}, \quad (4)$$

Fig. 1 | Schematic overview of the CAMP model architecture.

a An atomic structure is represented as a graph. Each atom is associated with a feature \mathbf{h} , and all atoms within a distance of r_{cut} to a center atom i constitute its local atomic neighborhood. **b** Atomic numbers z_i and z_j of a pair of atoms and their relative position vector \mathbf{r} are expanded into the radial part \mathbf{R} and angular part \mathbf{D} . **c** Atomic moment \mathbf{M} of the central atom is obtained by aggregating information from neighboring atoms. **d** Hyper moment \mathbf{H} of the central atom is computed by self-interactions between atomic moments. **e** Hyper moments \mathbf{H}^t of different layers t are used to construct the energy of the central atom via some mapping function V .



where \hat{r}_x , \hat{r}_y , and \hat{r}_z are the Cartesian components of \hat{r} (Fig. 1b). Any D_v is symmetric, as can be seen from the definition in Eq. (3). In CAMP, the angular information of an atomic environment is captured by D_0 , D_1 , D_2 , and so on, which are straightforward to evaluate. In full notation with atom indices, a polyadic tensor is written as D_v^j .

Atomic moment. A representation of the local atomic environment of an atom is constructed from the radial basis, angular part, and atom feature, which we call the *atomic moment*:

$$M_{uv,p}^i = \sum_{j \in \mathcal{N}_i} R_{uvv_1 v_2} h_{uv_1}^j \odot^c D_{v_2}^{jj}, \quad (5)$$

where \mathcal{N}_i denotes the set of neighboring atoms within a distance of r_{cut} to atom i . The radial part is obtained by passing the radial basis through a multilayer perceptron (MLP), $R_{uvv_1 v_2} = \text{MLP}(B_u)$. For different combinations of v , v_1 , and v_2 , different MLPs are used. The symbol \odot^c denotes a degree- c contraction between two tensors. For example, for tensors A and B of rank-2 and rank-3, respectively, $A \odot^2 B$ means $C_k = \sum_{ij} A_{ij} B_{ijk}$, resulting in a rank-1 tensor. In other words, c is the number of indices contracted away in each of the two tensors involved in the contraction; thus, the output tensor has a rank of $v = v_1 + v_2 - 2c$. We denote the combination of v_1 , v_2 , and c as a *path*: $p = (v_1, v_2, c)$. In Eq. (5), multiple paths can result in output tensors of the same rank v . For example, $p = (2, 3, 2)$ and $p = (1, 2, 1)$ both lead to atomic moment rank $v = 1$. This is why the index p is used in $M_{uv,p}^i$ to denote the path from which the atomic moment is obtained.

The atomic moment carries significant physical implications. For a physical quantity Q at a distance r from a reference point, the n th moment of Q is defined as $r^n Q$. For example, when Q represents a force and $n = 1$, rQ corresponds to the torque. Eq. (5) generalizes this concept, with the feature $h_{uv_1}^j$ of atom j acting as Q , and $D_{v_2}^{jj}$ serving as the “distance” r . The radial part $R_{uvv_1 v_2}$ functions as a weighting factor, modulating the relative contribution of different atoms j . This is why we refer to the output in Eq. (5) as the atomic moment.

By construction, the atomic moment is symmetric. This is because, first, both $h_{uv_1}^j$ (explained below around Eq. (9)) and $D_{v_2}^{jj}$ are symmetric; second, an additional constraint $c = v_1 < v_2$ is imposed. The constraint means that the indices of $h_{uv_1}^j$ and thus the output atomic moment has $v = v_2 - v_1$ indices from $D_{v_2}^{jj}$. A detailed example of evaluating Eq. (5) is provided in the Supplementary Information (SI).

Atomic moment tensors of the same rank v from different paths are combined using a linear layer:

$$M_{uv} = \sum_p W_{uv,p} M_{uv,p}, \quad (6)$$

in which $W_{uv,p}$ are trainable parameters, and the atom index i is omitted for simplicity.

Hyper moment. From atomic moments, we create the *hyper moment* as

$$H_{uv,p} = M_{uv_1} \odot^{c_1} M_{uv_2} \odot^{c_2} \dots \odot^{c_{n-1}} M_{uv_n}. \quad (7)$$

Atomic moments capture two-body interactions between a center atom and its neighbors. Three-body, four-body, and higher-body interactions are incorporated into the hyper moments via the self-interactions between atomic moments in Eq. (7). The hyper moments can provide a complete description of the local atomic environment by increasing the order of interactions^{36,37}, which is crucial for constructing systematically improvable MLIPs. The hyper moments are analogous to the B-basis used in ACE³⁵ and MACE¹⁶.

The primary target of an MLIP model is the potential energy, a scalar quantity. Aligned with this, we construct hyper moments such that information flows from higher-rank atomic moment tensors to lower-rank ones, ultimately to scalars. Specifically, we impose the following rule:

let M_{uv_n} be the tensor of the highest rank among all atomic moment tensors in Eq. (7), then possible contractions are only between it and the other atomic moments, but not between M_{uv_1} and M_{uv_2} , and so forth. This design choice substantially reduces the number of hyper moments, which simplifies both the model architecture and the overall training process. Moreover, similar to the atomic moment, the indices of lower-rank atomic tensors $M_{uv_1}, M_{uv_2}, \dots$ are completely contracted away, and therefore we have $v = v_n - (v_1 + v_2 + \dots + v_{n-1})$. This guarantees that $H_{uv,p}$ is constructed to be symmetric. Multiple contractions can result in hyper moments of the same rank v ; they are indexed by p . A detailed example of evaluating Eq. (7) to create hyper moments is provided in the SI.

Message Passing. The messages to an atom i from all neighboring atoms are chosen to be a linear expansion over the hyper moments:

$$m_{uv} = \sum_p W_{uv,p} H_{uv,p}. \quad (8)$$

Atom features are then updated using a residual connection⁴² by linearly combining the message and the atom feature of the previous layer:

$$h_{uv}^t = \sum_{u'} W_{1,uu'} m_{u'v} + \sum_{u'} W_{2,uu'} h_{u'v}^{t-1}, \quad (9)$$

where t is the layer index. Atom features in the input layer h_{u0}^0 consist only of scalars h_{u0}^0 , obtained as a u -dimensional embedding of the atomic number z . By construction, h_{uv}^t is symmetric.

Output. The message passing is performed multiple times T . We find that typical values of two or three are good enough based on benchmark studies to be discussed in the following sections. The atomic energy of atom i is then obtained from the scalar atom features $h_{i0}^{i,t}$ from all layers:

$$E^i = \sum_{t=1}^T V(h_{i0}^{i,t}). \quad (10)$$

V is set to an MLP for the last layer T , and a linear function for others layers, that is, $V(h_{i0}^{i,t}) = \sum_u W_u^t h_{i0}^{i,t}$ for $t < T$. The atomic energies of all atoms are summed to get the total potential energy

$$E = \sum_i E^i. \quad (11)$$

Forces on atom i can then be computed as

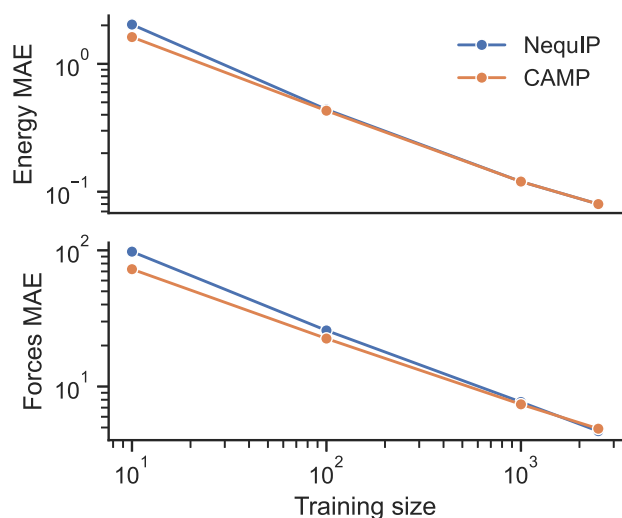
$$F_i = -\frac{\partial E}{\partial r_i}. \quad (12)$$

Computational complexity. The most computationally demanding part of CAMP is creating the atomic moments and hyper moments. Let v_{max} represent the maximum rank of tensors used in the model. This implies that all features h , atomic moments M , and hyper moments H have ranks up to v_{max} . The time complexity for constructing both the atomic moment in Eq. (5) and the hyper moment in Eq. (7) is $\mathcal{O}(3^{v_{\text{max}}})$ (detailed analysis in the SI). In contrast, for models based on spherical tensors (such as NequIP¹⁷ and MACE¹⁶), the time complexity of the Clebsch–Gordan tensor product is $\mathcal{O}(L^6)$, where L is the maximum degree of the irreducible spherical tensors. A Cartesian tensor (e.g., M and H) of rank v can be decomposed as the sum of spherical tensors of degrees up to v ⁴³; therefore, as far as the computational cost is concerned, v_{max} and L can be thought to be equivalent to each other. Asymptotically, as L (or v_{max}) increases, the exponential-scaling Cartesian models are computationally less efficient than the polynomial-scaling spherical models. Nevertheless, empirical evidence suggests that, for many material systems, the adopted values of L or (v_{max}) are below five^{16,17,44}. In such cases,

Table 1 | Performance of CAMP on the LiPS dataset

Training size		NequIP	CAMP
10	Energy	2.03	1.63 (0.16)
	Forces	97.8	76.4 (3.3)
100	Energy	0.44	0.44 (0.02)
	Forces	25.8	21.6 (0.6)
1000	Energy	0.12	0.12 (0.00)
	Forces	7.7	7.4 (0.4)
2500	Energy	0.08	0.08 (0.00)
	Forces	4.7	4.9 (0.1)

MAEs of energy and forces are in the units of meV/atom and meV/Å, respectively. Bold values represent the smallest MAEs in each row. Errors shown in parentheses are obtained as the standard deviations from five independently trained CAMP models.

**Fig. 2 | Learning curve of CAMP.** On a log-log scale, the MAEs of both energy (meV/atom) and forces (meV/Å) decrease linearly with the training set size.

Cartesian models should be more computationally efficient. We note that there exist techniques to reduce the computational cost of the Clebsch–Gordan tensor product in spherical models from $\mathcal{O}(L^6)$ to $\mathcal{O}(L^3)$, such as reducing SO(3) convolutions to SO(2)⁴⁵ and replacing SO(3) convolutions by matrix multiplications⁴⁶. When such techniques are adopted, spherical models can be more efficient.

Inorganic crystals

We first test CAMP on a dataset of inorganic lithium phosphorus sulfide (LiPS) solid-state electrolytes¹⁷. Multiple models were trained using varying numbers of training samples (10, 100, 1000, and 2500), validated on 1000 samples, and tested on 5000 samples (training details in Methods). Mean absolute errors (MAEs) of energy and forces on the test set are listed in Table 1. Compared with NequIP¹⁷, CAMP yields smaller or equal errors on seven of the eight tasks. The learning curve in Fig. 2 exhibits a linear decrease in both energy and forces MAEs on a log-log scale as the number of training samples increases. These results demonstrate CAMP's accuracy and its capacity for further improvement with increased data.

We further examine the performance in MD simulations, focusing on computing the diffusivity of lithium ions (Li^+) in LiPS. Diffusivity is a crucial characteristic for assessing the potential of solid materials as electrolytes in next-generation solid-state batteries. Following existing benchmark studies¹⁷, we trained another model with 19000 structures with the same validation and test sets discussed above. Five structures were randomly selected from the test set and an independent MD simulation was performed

for each structure. As seen in Fig. 3, CAMP accurately captures the structural information of LiPS, reproducing the radial distribution function (RDF) and the angular distribution function (ADF) of the S–P–S tetrahedral angle (Fig. S1 in the SI) from ab initio molecular dynamics (AIMD) simulations.

CAMP can yield stable MD simulations. It is known that MLIPs that perform well on energy and forces do not necessarily guarantee high-quality MD simulations. In particular, MD simulations can collapse due to model instability. Therefore, before calculating the diffusivity, we checked the simulation stability. Stability is measured as the difference between the RDF averaged over a time window and the RDF of the entire simulation (see Methods). CAMP shows excellent stability, reaching the entire simulation time of 50 ps in all five simulations (Table 2). In ref. 47, a small timestep of 0.25 fs was adopted in the MD simulations using the models in Table 2. We also tested CAMP with a large timestep of 1 fs and found that all five simulations are still stable up to 50 ps (see Fig. S2 in the SI).

The diffusion coefficient D of Li^+ in LiPS calculated from CAMP agrees well with the AIMD result. After confirming the stability, D was calculated using the Einstein equation by fitting a linear line between the mean squared displacement (MSD) of Li^+ and the correlation time (see Methods). The MSDs of the five MD runs are shown in Fig. 3c, from which the diffusion coefficient is calculated as $D = (1.08 \pm 0.08) \times 10^{-5} \text{ cm}^2/\text{s}$. This agrees reasonably well with AIMD result of $1.37 \times 10^{-5} \text{ cm}^2/\text{s}$ ¹⁷.

Bulk water

To evaluate CAMP's ability to model complex liquid systems, we test it on a dataset of bulk water³⁹. See Methods Dataset and training details are in Root-mean-square errors (RMSEs) of energy and forces on the test set are listed in Table 3. In general, GNN models (REANN⁴⁸, NequIP¹⁷, MACE¹⁶, and CACE³⁸) have smaller errors than single-layer ACE³⁵ and descriptors-based BPNN³⁹. Our CAMP achieves the best performance on both energy and forces, with RMSEs of 0.59 meV/atom and 34 meV/Å, respectively. We observe that CACE³⁸ demonstrates comparable RMSEs in energy when compared to CAMP. CACE³⁸ is a recent model also developed entirely in Cartesian space. Similar to CAMP, it also first constructs the atomic basis (Eq. (5)), then builds the product basis on top of the atomic basis (Eq. (7)), and iteratively updates the features under a GNN framework to refine the representations. A major difference is that CACE is formulated based on the atomic cluster expansion, while CAMP is designed using the atomic moment. The connections between CAMP and CACE, as well as other models, are further elaborated in Discussion.

CAMP accurately reproduces experimental structural and dynamical properties of water. Figure 4 shows the oxygen–oxygen RDF of water from MD simulations using CAMP (computation details in Methods), together with experimental observations obtained from neutron diffraction⁴⁹ and x-ray diffraction⁵⁰. The RDF by CAMP almost overlaps with the x-ray diffraction result, demonstrating its ability to capture the delicate structural features of water. We also calculated the diffusion coefficient of water (1 g/cm³) at 300 K to be $D = (2.79 \pm 0.18) \times 10^{-5} \text{ cm}^2/\text{s}$, which is in excellent agreement with AIMD results of $2.67 \times 10^{-5} \text{ cm}^2/\text{s}$ ⁵¹. The model shows superior stability for water, producing stable MD simulations at high temperatures up to 1500 K. The RDF and the diffusion coefficient at various high temperatures are presented in Fig. S3 in the SI.

We examine the efficiency of CAMP by measuring the number of completed steps per second in MD simulations. It is not a surprise that DeePMD⁹ runs much faster due to its simplicity in model architecture, although its accuracy falls short when compared with more recent models (Table 3). For those more accurate models, CAMP is about 1.4 and 3.6 times faster than CACE³⁸ and NequIP¹⁷, respectively, on the water system of 192 atoms. More running speed data of other models are provided in Table S1 in the SI.

Small organic molecules

The MD17 dataset consists of AIMD trajectories of several small organic molecules⁸. For each molecule, we trained CAMP on 950 random samples, validated on 50 samples, and the rest were used for testing (dataset and

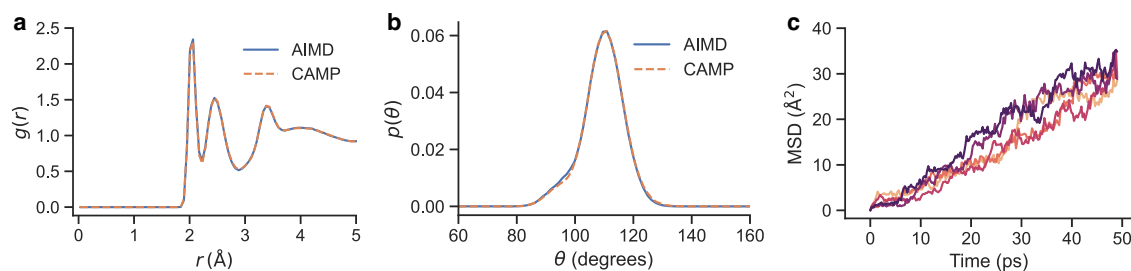


Fig. 3 | MD simulation results on LiPS. a Radial distribution function. **b** Angular distribution function for the S–P–S tetrahedral angle. **c** Mean squared displacement of Li^+ in LiPS. Five MD simulations are performed, and the calculated Li^+ diffusion coefficient is $D = (1.08 \pm 0.08) \times 10^{-5} \text{ cm}^2/\text{s}$.

Table 2 | Results on LiPS for models developed with 19000 training samples

	DeepPot-SE ⁵²	SchNet ¹⁰	DimeNet ¹²	PaiNN ¹³	SphereNet ⁵⁴	ForceNet ⁷⁰	GemNet-T ⁵³	NequIP ¹⁷	CAMP
Forces	40.5	28.8	3.2	11.7	8.3	12.8	1.3	3.7	3.6
Diffusivity		0.38	0.30	0.40	0.40		0.24	0.34	0.18
Stability	4 (3)	50 (0)	48 (4)	50 (0)	50 (0)	26 (8)	50 (0)	50 (0)	50 (0)

MAE of forces and MAE of Li^+ diffusivity are reported in $\text{meV}/\text{\AA}$ and $10^{-5} \text{ cm}^2/\text{s}$, respectively. The reference diffusivity is $1.35 \times 10^{-5} \text{ cm}^2/\text{s}$. Stability is measured by MD simulation time in picoseconds, with standard deviations obtained from five independent runs shown in parentheses. All results are obtained from ref. 47, except those for CAMP.

Table 3 | Model performance on the water dataset

	BPNN ³⁹	ACE ³⁵	REANN ⁴⁸	DeePMD ⁹	NequIP ¹⁷	MACE ¹⁶	CACE ³⁸	CAMP
Energy	2.3	1.7	0.8	2.1	0.94	0.63	0.59	0.59 (0.002)
Forces	120	99	53	92	45	36	47	34 (0.1)
Speed				61.8	3.9		9.8	14.2

RMSEs of energy and forces are in the units of meV/atom and $\text{meV}/\text{\AA}$, respectively. Speed is measured as completed steps per second in MD simulations of the water system of 192 atoms, using a single NVIDIA V100 GPU. All energy and forces RMSE results, except CAMP, are from ref. 38. DeePMD and NequIP running speed results are from ref. 47. Errors shown in parentheses are obtained as the standard deviations from five independently trained CAMP models.

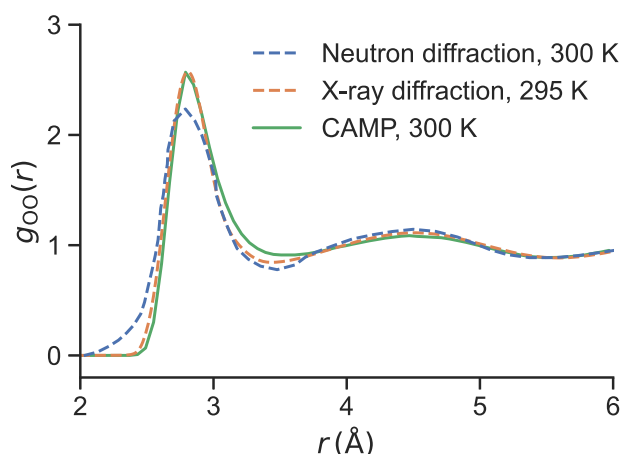


Fig. 4 | Oxygen–Oxygen RDF of water. Experimental neutron diffraction⁴⁹ and x-ray diffraction⁵⁰ results are shown for comparison.

training details in Methods). MAEs of energy and forces by CAMP and other MLIPs are listed in Table 4. While NequIP¹⁷ generally maintains high performance, CAMP demonstrates high competitiveness. It surpasses NequIP in energy predictions for three out of seven molecules and ranks second in both energy and force predictions across various cases.

We further tested the ability of CAMP to maintain long-time stable MD simulations. Following the benchmark study in ref. 47, 9500, 500, and 10,000 molecules were randomly sampled for training, validation, and testing, respectively. With the trained models, we performed MD simulations at 500 K for 300 ps and examined the stability by measuring the change

in bond length during the simulations (see Methods). Figure 5 presents the results for Aspirin and Ethanol (numerical values in Table S2 in the SI). In general, a small forces MAE does not guarantee stable MD simulations⁴⁷, as seen from a comparison between ForceNet and GemNet-T. GemNet-T has a much smaller MAE of forces, but its stability is not as good as ForceNet. For CAMP, its MAE of forces is slightly larger than that of, e.g. GemNet-T and SphereNet, but it can maintain stable MD simulations for the entire simulation time of 300 ps, a significant improvement over most of the existing models.

In terms of running speed, similar patterns are observed here as those reported in the water system. Models based on simple scalar features such as SchNet¹⁰ and DeepPot-SE⁵² are faster than those based on tensorial features, although they are far less accurate (Fig. 5). For both Aspirin and Ethanol, CAMP can complete about 33 steps per second in MD simulations on a single NVIDIA V100 GPU, which is approximately 1.2, 1.4, and 3.9 times faster than GemNet-T⁵³, SphereNet⁵⁴, and NequIP¹⁷, respectively.

The MD17 dataset was adopted due to the excellent benchmark study by Fu et al.⁴⁷, which provides a consistent basis for comparing different models. However, it has been observed that the MD17 dataset contains significant numerical errors, prompting the introduction of the rMD17 dataset to address this issue⁵⁵. We evaluated CAMP on the aspirin and ethanol molecules from the rMD17 dataset, and it demonstrates competitive performance. While CAMP has errors higher than those of leading spherical models like NequIP¹⁷, Allegro⁵⁶, and MACE¹⁶, it outperforms models such as GAP⁷, FCHL⁵⁷, ACE³⁵, and PaiNN¹³. Detailed results are presented in Table S3 of the SI.

Two-dimensional materials

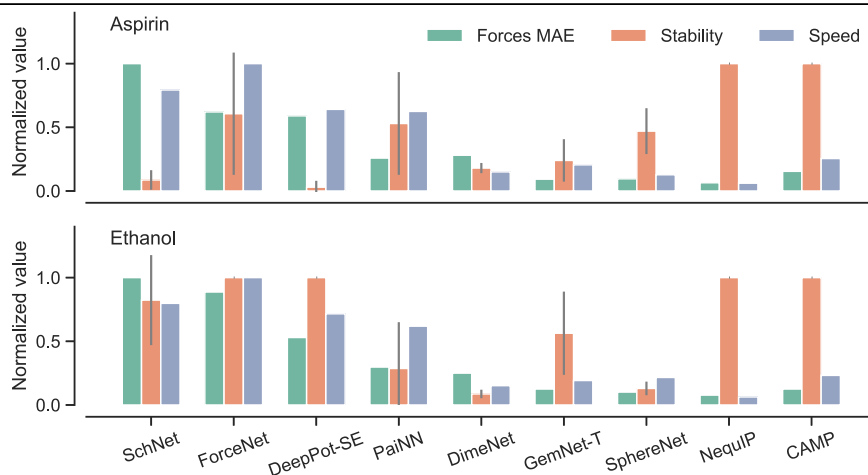
In addition to periodic systems and small organic molecules, we evaluate the performance of CAMP on partially periodic 2D materials. These materials

Table 4 | Model performance on the MD17 dataset

Molecule		SchNet ¹⁰	DimeNet ⁷¹	sGDML ⁸	PaiNN ¹³	NewtonNet ⁷²	NequiP ¹⁷	CAMP
Aspirin	Energy	16.0	8.8	8.2	<u>6.9</u>	7.3	5.7	8.7 (0.7)
	Forces	58.5	21.6	29.5	<u>14.7</u>	15.1	8.0	20.5 (1.5)
Ethanol	Energy	3.5	2.8	3.0	2.7	2.6	2.2	<u>2.5</u> (0.1)
	Forces	16.9	10.0	14.3	9.7	9.1	3.1	<u>7.6</u> (0.4)
Malonaldehyde	Energy	5.6	4.5	4.3	<u>3.9</u>	4.2	3.3	<u>3.9</u> (0.0)
	Forces	28.6	16.6	17.8	13.8	14.0	5.6	<u>12.7</u> (0.3)
Naphthalene	Energy	6.9	5.3	5.2	5.0	5.1	<u>4.9</u>	4.2 (0.0)
	Forces	25.2	9.3	4.8	<u>3.3</u>	3.6	1.7	4.0 (0.3)
Salicylic acid	Energy	8.7	5.8	5.2	<u>4.9</u>	5.0	4.6	4.6 (0.3)
	Forces	36.9	16.2	12.1	<u>8.5</u>	<u>8.5</u>	3.9	11.0 (2.1)
Toluene	Energy	5.2	4.4	4.3	<u>4.1</u>	<u>4.1</u>	4.0	4.2 (0.2)
	Forces	24.7	9.4	6.1	<u>4.1</u>	<u>3.8</u>	2.0	4.6 (0.6)
Uracil	Energy	6.1	5.0	4.8	4.5	4.6	<u>4.5</u>	4.4 (0.0)
	Forces	24.3	13.1	10.4	<u>6.0</u>	6.5	3.3	7.2 (0.3)

MAEs of energy and forces are in the units of meV and meV/Å, respectively. **Bold**: smallest; underline: second smallest. Errors shown in parentheses are obtained as the standard deviations from five independently trained CAMP models.

Fig. 5 | Comparison of MAE of forces, MD stability, and running speed on small molecules. The forces MAE, stability, and running speed are normalized by their maximum values for easier visualization. All numerical values are provided in Table S2 in the SI. Error bars indicate the standard deviation from five MD runs using different starting atomic positions. Computational speed is measured on a single NVIDIA V100 GPU.

**Table 5 | Results on the bilayer graphene dataset**

	AIREBO ⁵⁸	AIREBO-M ⁵⁹	LCBOP ⁶⁰	hNN ⁴⁰	CAMP
Energy	659.9	671.2	718.3	1.4	0.35 (0.05)
Forces	194.6	186.9	166.1	46.0	6.26 (0.04)

MAEs of the energy and forces are in the units of meV/atom and meV/Å, respectively. Errors shown in parentheses are obtained as the standard deviations from five independently trained CAMP models.

exhibit distinct physical and chemical properties. Despite their significance, to the best of our knowledge, no standardized benchmark dataset exists for MLIPs for 2D materials. We have thus constructed a new DFT dataset of bilayer graphene, building upon our previous investigations of carbon systems⁴⁰. See Methods for detailed information on this new dataset.

Widely used empirical potentials for carbon systems, such as AIREBO⁵⁸, AIREBO-M⁵⁹, and LCBOP⁶⁰ have large errors in predicting both the energy and forces of bilayer graphene. The MAEs on the test set against DFT results are on the order of hundreds of meV/atom for energy and meV/Å for forces (Table 5). This is not too surprising given that: first, these potentials are general-purpose models for carbon systems, which are not specifically designed for multilayer graphene systems; second, their training data were experimental properties and/or DFT calculations using different density functionals from the test set. MLIPs such as hNN⁴⁰ (a hybrid model

that combines BPNN⁶ and Lennard-Jones⁶¹), which was trained on the same data as used here, can significantly reduce the errors to 1.4 meV/atom for energy and 46.0 meV/Å for forces. CAMP can further drive the MAEs down to 0.3 meV/atom for energy and 6.3 meV/Å for forces.

It is interesting to investigate the interlayer interaction between the graphene layers, which controls many structural, mechanical, and electronic properties of 2D materials^{62,63}. Here, we focus on the energetics in different stacking configurations: AB, AA, and saddle point (SP) stackings (Fig. 6a). Empirical models such as LCBOP cannot distinguish between the different stacking states at all. The interlayer energy versus layer distance curves are almost identical between AB and AA (Fig. 6b), and the generalized stacking fault energy surface is nearly flat (not shown), with a maximum value on the order of 0.01 meV/atom. This is also the case for AIREBO and AIREBO-M (we refer to ref. 40 for plots). On the contrary, CAMP and hNN can clearly

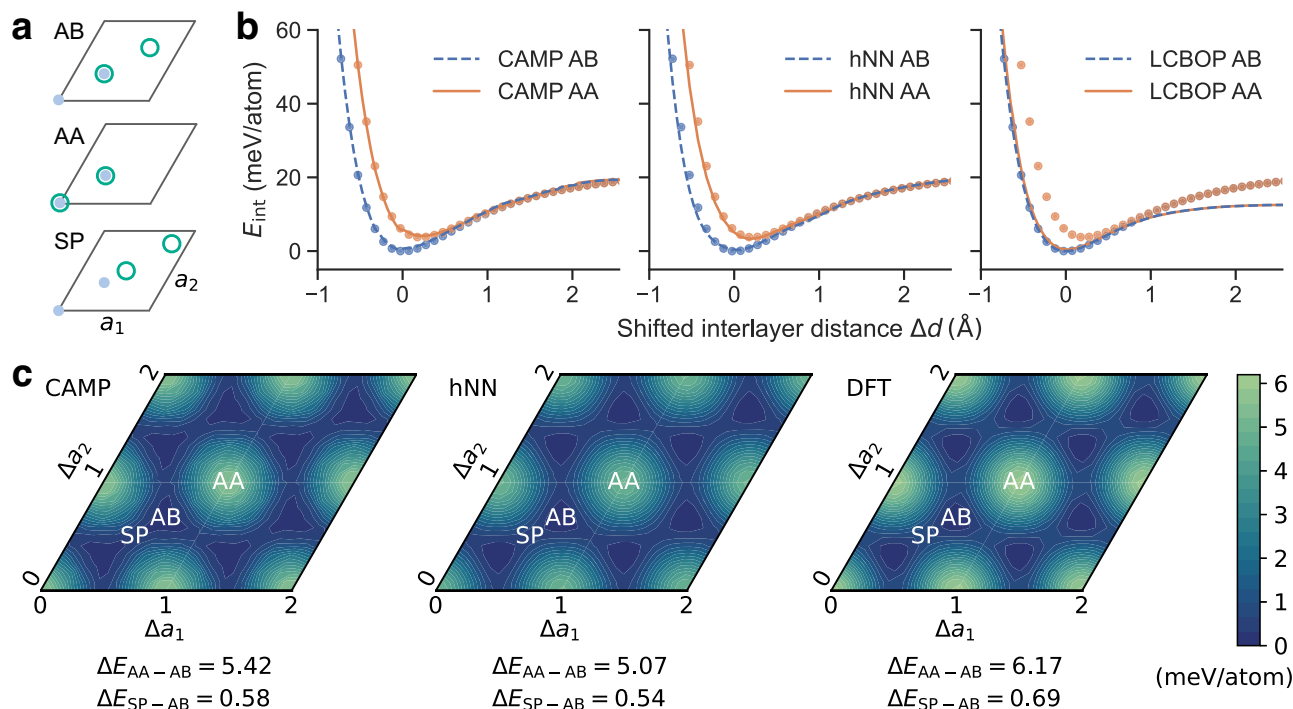


Fig. 6 | Interlayer energetics of bilayer graphene. **a** Bilayer graphene in AB, AA, and saddle point (SP) stacking, where the blue solid dots represent atoms in the bottom layer, and the green hollow dots represent atoms in the top layer. **b** Interlayer energy E_{int} versus layer distance d of bilayer graphene in AB and AA stacking. The dots represent DFT results. The layer distance is shifted such that $\Delta d = d - d_0$, where $d_0 =$

3.4 Å is the equilibrium layer distance. **c** Generalized stacking fault energy of bilayer graphene. This is obtained by sliding the top layer against the bottom layer at a fixed layer distance of d_0 , where Δa_1 and Δa_2 indicate the shifts of the top layer against the bottom layer, along the lattice vectors a_1 and a_2 , respectively.

distinguish between the stackings. Both the interlayer energy versus layer distance curves and the generalized stacking fault energy surface agree well with DFT references, and CAMP has a slightly better prediction of the energy barrier ($\Delta E_{\text{SP-AB}}$) and the overall energy corrugation ($\Delta E_{\text{AA-AB}}$). The hNN model is specifically designed as an MLIP for 2D materials, and its training process is complex, requiring separate training of the Lennard-Jones and BPNN components. In contrast, CAMP does not require such special treatment and is straightforward to train.

Discussion

In this work, we develop CAMP, a new class of MLIP that is based on atomic moment tensors and operates entirely in Cartesian space. CAMP is designed to be physically inspired, flexible, and expressive, and it can be applied to a wide range of systems, including periodic structures, small organic molecules, and 2D materials. Benchmark tests on these systems demonstrate that CAMP achieves performance surpassing or comparable to current leading models based on spherical tensors in terms of accuracy, efficiency, and stability. It is robust and straightforward to train, without the need to tune a large number of hyperparameters. In all the tested systems, we only need to set four hyperparameters: the number of channels u , the maximum tensor rank v_{max} , the number of layers T , and the cutoff radius r_c to achieve good performance.

CAMP is related to existing models in several ways. As shown in ref. 37, ACE³⁵ and MTP³⁶ can be viewed as the same model, with the former constructed in spherical space while the latter in Cartesian space. Both are single-layer models without iterative feature updates. Loosely speaking, MACE¹⁶ and CAMP can be regarded as a generalization of the spherical ACE and the Cartesian MTP, respectively, to multilayer GNNs with iterative feature updates and refinement. The atomic moment in Eq. (5) and hyper moment in Eq. (7) are related to the A-basis and B-basis in ACE and MACE. The recent CACE model is related to MACE and CAMP. Both MACE and CACE are based on atomic cluster expansion, but MACE implements this expansion in spherical space while CACE builds it in Cartesian space. Both

CAMP and CACE develop features in Cartesian space, but CAMP uses atomic moments while CACE employs atomic cluster expansion. Moreover, CAMP generalizes existing Cartesian tensor models such as TeaNet¹⁸ and TensorNet¹⁹ that use at most second-rank tensors to tensors of arbitrary rank. Despite these connections, CAMP is unique in its design of the atomic moment and hyper moment, and the selection rules that govern the tensor contractions. These characteristics make CAMP a physically inspired, flexible, and expressive model.

Beyond energy and forces, CAMP can be extended to model tensorial properties. The atom features used in CAMP are symmetric Cartesian tensors, which can be used to output tensorial properties with slight modifications to the output block. For example, NMR tensors can be modeled by selecting and summing the scalar, vector, and symmetric rank-2 tensor components of the hyper moments, rather than using only the scalar component for potential energies. However, the current implementation of CAMP in PyTorch has certain limitations. All feature tensors are stored in full form, without exploiting the fact that they are symmetric. In addition, it is possible to extend CAMP to use irreducible representations of Cartesian tensors (basically symmetric traceless tensors), analogous to spherical irreducible representations used in spherical tensor models. Leveraging these symmetries and irreducible representations could further enhance model accuracy and improve time and memory efficiency. These are directions for future work.

Methods

Dataset

The LiPS dataset¹⁷ consists of 250001 structures of lithium phosphorus sulfide ($\text{Li}_{6.75}\text{P}_3\text{S}_{11}$) obtained from an AIMD trajectory. Each structure has 83 atoms, with 27 Li, 12 P, and 44 S. The data are randomized before splitting into training, validation, and test sets. The water dataset³⁹ consists of 1593 water structures, each with 192 atoms, generated with AIMD simulations at 300 K. The data are randomly split into training, validation, and test sets with a ratio of 90:5:5. The MD17 dataset⁸ consists of AIMD trajectories of

several small organic molecules. The number of data points for each molecule ranges from 133,000 to 993,000. Our new bilayer graphene dataset is derived from ref. 40, with 6178 bilayer graphene configurations from stressed structures, atomic perturbations, and AIMD trajectories. The data were generated using DFT calculations with the PBE functional⁶⁴, and the many-body dispersion correction method⁶⁵ was used to account for van der Waals interactions. The data are split into training, validation, and test sets with a ratio of 8:1:1.

Model training

The models are trained by minimizing a loss function of energy and forces. For an atomic configuration \mathcal{C} , the loss is

$$l(\theta; \mathcal{C}) = w_E \left(\frac{E - \hat{E}}{N} \right)^2 + w_F \frac{\sum_{i=1}^N \| \mathbf{F}_i - \hat{\mathbf{F}}_i \|^2}{3N}, \quad (13)$$

where N is the number of atoms, E and \hat{E} are the predicted and reference energies, respectively, and \mathbf{F}_i and $\hat{\mathbf{F}}_i$ are the predicted and reference forces on atom i , respectively. Both the energy weight w_E and force weight w_F are set to 1. The total loss to minimize at each optimization step is the sum of the losses of multiple configurations,

$$L(\theta) = \sum_{j=1}^B l(\theta; \mathcal{C}_j), \quad (14)$$

where B is the mini-batch size.

The CAMP model is implemented in PyTorch⁶⁶ and trained with PyTorch Lightning⁶⁷. We trained all models using the Adam optimizer⁶⁸ with an initial learning rate of 0.01, which is reduced by a factor of 0.8 if the validation error does not decrease for 100 epochs. The training batch size and allowed maximum number of training epochs vary for different datasets. Training is stopped if the validation error does not reduce for a certain number of epochs (see Table S4 in the SI). We use an exponential moving average with a weight 0.999 to evaluate the validation set as well as for the final model.

Regarding model structure, Chebyshev polynomials of degrees up to $N_\beta = 8$ are used for the radial basis functions. Other hyperparameters include the number of channels u , maximum tensor rank v_{\max} , number of layers T , and cutoff distance r_{cut} . Optimal values of these hyperparameters are searched for each dataset, and typical values are around $u = 32$, $v_{\max} = 3$, $T = 3$, and $r_{\text{cut}} = 5 \text{ \AA}$. These result in small, parameter-efficient models with fewer than 125k parameters. Detailed hyperparameters for each dataset are provided in Table S5 in the SI.

Diffusivity

The diffusivity can be computed from MD simulations via the Einstein equation, which relates the MSD of particles to their diffusion coefficient:

$$D = \lim_{t \rightarrow \infty} \frac{\langle \frac{1}{N} \sum_i |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle}{2nt}, \quad (15)$$

where the MSD is computed as the ensemble average (denoted by $\langle \cdot \rangle$) using diffusing atoms, $\mathbf{r}_i(t)$ represents the position of atom i at time t , N is the total number of considered diffusion atoms, n denotes the number of dimensions (three here), and D is the diffusion coefficient. To solve for the diffusion coefficient D , we employ a linear fitting approach as implemented in ASE⁶⁹, where D is obtained as the slope of the MSD versus $2nt$.

The diffusivity of Li^+ in LiPS was computed from MD simulations under the canonical ensemble using the Nosé–Hoover thermostat. Simulations were performed at a temperature of 520 K for a total of 50 ps, with a timestep of 0.25 fs, consistent with the settings reported in the benchmark study in ref. 47. We similarly computed the diffusivity of water; five simulations were performed at 300 K, each using a timestep of 1 fs and running for a total of 50 ps.

Stability criteria

For periodic systems, stability is monitored from the RDF $g(r)$ such that a simulation becoming “unstable” at time t when⁴⁷

$$\int_0^\infty \| \langle g(r) \rangle_t^{t+\tau} - \langle g(r) \rangle \| \, dr > \Delta, \quad (16)$$

where $\langle \cdot \rangle$ denotes the average using the entire trajectory, $\langle \cdot \rangle_t^{t+\tau}$ denotes the average in a time window $[t, t + \tau]$, and Δ is a threshold. In other words, when the difference of the area under the RDF obtained from the time window and the entire trajectory exceeds Δ , the simulation is considered unstable.

This criterion cannot be applied to characterize the stability of MD simulations where large structural changes are expected, such as in phase transitions or chemical reactions. However, for the LiPS system studied here, no such events are expected, and the RDF-based stability criterion is appropriate. We adopted $\tau = 1 \text{ ps}$ and $\Delta = 1$, as proposed in ref. 47.

For molecular systems, the stability is monitored through the bond lengths, and a simulation is considered “unstable” at time T when⁴⁷

$$\max_{(i,j) \in \mathcal{B}} |r^{ij}(T) - b^{ij}| > \Delta, \quad (17)$$

where \mathcal{B} denotes the set of all bonds, Δ is a threshold, $r^{ij}(T)$ is the bond length between atoms i and j at time T , and b^{ij} is the corresponding equilibrium bond length, computed as the average bond length using the reference DFT data. For the MD17 dataset, we adopted $\Delta = 0.5 \text{ \AA}$ as in ref. 47, and the MD simulations were performed at 500 K for 300 ps with a timestep of 0.5 fs, using the Nosé–Hoover thermostat.

Data availability

The new bilayer graphene dataset is provided at https://github.com/wengroup/camp_run. The other datasets used in this work are publicly available; the LiPS dataset: <https://archive.materialscloud.org/record/2022.45>, the water dataset: <https://doi.org/10.1073/pnas.1815117116>, and the MD17 dataset: <http://www.sgdm.org>.

Code availability

The code for the CAMP model is available at <https://github.com/wengroup/camp>. Scripts for training models, running MD simulations, and analyzing the results are at https://github.com/wengroup/camp_run.

Received: 18 November 2024; Accepted: 23 April 2025;

Published online: 09 May 2025

References

- Huang, B. & von Lilienfeld, O. A. Ab initio machine learning in chemical compound space. *Chem. Rev.* **121**, 10001–10036 (2021).
- Unke, O. T. et al. Machine learning force fields. *Chem. Rev.* **121**, 10142–10186 (2021).
- Deringer, V. L. et al. Gaussian process regression for materials and molecules. *Chem. Rev.* **121**, 10073–10141 (2021).
- Wen, M., Afshar, Y., Elliott, R. S. & Tadmor, E. B. Kliff: A framework to develop physics-based and machine learning interatomic potentials. *Comput. Phys. Commun.* **272**, 108218 (2022).
- Daw, M. S. & Baskes, M. I. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B* **29**, 6443–6453 (1984).
- Behler, J. & Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **98**, 146401 (2007).
- Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **104**, 136403 (2010).

8. Chmiela, S. et al. Machine learning of accurate energy-conserving molecular force fields. *Sci. Adv.* **3**, e1603015 (2017).
9. Zhang, L., Han, J., Wang, H., Car, R. & E, W. Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics. *Phys. Rev. Lett.* **120**, 143001 (2018).
10. Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet – a deep learning architecture for molecules and materials. *J. Chem. Phys.* **148**, 241722 (2018).
11. Zubatyuk, R., Smith, J. S., Leszczynski, J. & Isayev, O. Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network. *Sci. Adv.* **5**, eaav6490 (2019).
12. Gastegger, J., Groß, J. & Günnemann, S. Directional message passing for molecular graphs. In *International Conference on Learning Representations (ICLR)* (2020).
13. Schütt, K., Unke, O. & Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, vol. 139, 9377–9388 (2021).
14. Ko, T. W., Finkler, J. A., Goedecker, S. & Behler, J. A fourth-generation high-dimensional neural network potential with accurate electrostatics including non-local charge transfer. *Nat. Commun.* **12**, 398 (2021).
15. Unke, O. T. et al. SpookyNet: Learning force fields with electronic degrees of freedom and nonlocal effects. *Nat. Commun.* **12**, 7273 (2021).
16. Batatia, I., Kovacs, D. P., Simm, G., Ortner, C. & Csányi, G. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Adv. Neural Inf. Process. Syst.* **35**, 11423–11436 (2022).
17. Batzner, S. et al. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 1–11 (2022).
18. Takamoto, S., Izumi, S. & Li, J. Teanet: Universal neural network interatomic potential inspired by iterative electronic relaxations. *Comput. Mater. Sci.* **207**, 111280 (2022).
19. Simeon, G. & De Fabritiis, G. Tensormet: Cartesian tensor representations for efficient learning of molecular potentials. In Oh, A. et al. (eds.) *Advances in Neural Information Processing Systems*, vol. 36, 37334–37353 (2023).
20. Wen, M. & Tadmor, E. B. Uncertainty quantification in molecular simulations with dropout neural network potentials. *npj Comput. Mater.* **6**, 124 (2020).
21. Zhu, A., Batzner, S., Musaelian, A. & Kozinsky, B. Fast uncertainty estimates in deep learning interatomic potentials. *J. Chem. Phys.* **158**, 164111 (2023).
22. Tan, A. R., Urata, S., Goldman, S., Dietschreit, J. C. B. & Gómez-Bombarelli, R. Single-model uncertainty quantification in neural network potentials does not consistently outperform model ensembles. *npj Comput. Mater.* **9**, 1–11 (2023).
23. Vita, J. A., Samanta, A., Zhou, F. & Lordi, V. Ltau-ff: Loss trajectory analysis for uncertainty in atomistic force fields. *ArXiv e-prints* 2402.00853 (2024).
24. Dai, J., Adhikari, S. & Wen, M. Uncertainty quantification and propagation in atomistic machine learning. *Rev. Chem. Eng.* **41**, 333–357 (2024).
25. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*, 1263–1272 (2017).
26. Chen, C. & Ong, S. P. A universal graph deep learning interatomic potential for the periodic table. *Nat. Comput. Sci.* **2**, 718–728 (2022).
27. Takamoto, S. et al. Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements. *Nat. Commun.* **13**, 1–11 (2022).
28. Deng, B. et al. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nat. Mach. Intell.* **5**, 1031–1041 (2023).
29. Batatia, I. et al. A foundation model for atomistic materials chemistry. *ArXiv e-prints* 2401.00096 (2023).
30. Xie, F., Lu, T., Meng, S. & Liu, M. Gptff: A high-accuracy out-of-the-box universal AI force field for arbitrary inorganic materials. *Sci. Bull.* **69**, 3525–3532 (2024).
31. Zhang, D. et al. Dpa-2: a large atomic model as a multi-task learner. *ArXiv e-prints* 2312.15492 (2023).
32. Yang, H. et al. Mattersim: A deep learning atomistic model across elements, temperatures and pressures. *ArXiv e-prints* 2405.04967 (2024).
33. Smith, J. S., Isayev, O. & Roitberg, A. E. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chem. Sci.* **8**, 3192–3203 (2017).
34. Thompson, A. P., Swiler, L. P., Trott, C. R., Foiles, S. M. & Tucker, G. J. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *J. Comput. Phys.* **285**, 316–330 (2015).
35. Drautz, R. Atomic cluster expansion for accurate and transferable interatomic potentials. *Phys. Rev. B* **99**, 014104 (2019).
36. Shapeev, A. V. Moment tensor potentials: A class of systematically improvable interatomic potentials. *Multiscale Model. Multiscale Model. Sim.* **14**, 1153–1173 (2016).
37. Dusson, G. et al. Atomic cluster expansion: Completeness, efficiency and stability. *J. Comput. Phys.* **454**, 110946 (2022).
38. Cheng, B. Cartesian atomic cluster expansion for machine learning interatomic potentials. *npj Comput. Mater.* **10**, 1–10 (2024).
39. Cheng, B., Engel, E. A., Behler, J., Dellago, C. & Ceriotti, M. Ab initio thermodynamics of liquid and solid water. *Proc. Natl Acad. Sci.* **116**, 1110–1115 (2019).
40. Wen, M. & Tadmor, E. B. Hybrid neural network potential for multilayer graphene. *Phys. Rev. B* **100**, 195419 (2019).
41. Novikov, I. S., Gubaev, K., Podryabinkin, E. V. & Shapeev, A. V. The mlip package: moment tensor potentials with mpi and active learning. *Mach. Learn.: Sci. Technol.* **2**, 025002 (2020).
42. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. *ArXiv e-prints* 1512.03385 (2015).
43. Zee, A. *Group Theory in a Nutshell for Physicists* (Princeton University Press, Princeton, NJ, USA, 2016). <https://press.princeton.edu/books/hardcover/9780691162690/group-theory-in-a-nutshell-for-physicists>.
44. Wen, M., Horton, M. K., Munro, J. M., Huck, P. & Persson, K. A. An equivariant graph neural network for the elasticity tensors of all seven crystal systems. *Digital Discov.* **3**, 869–882 (2024).
45. Passaro, S. & Zitnick, C. L. Reducing so(3) convolutions to so(2) for efficient equivariant gnns. In *Proceedings of the 40th International Conference on Machine Learning, ICML'23* (2023).
46. Maennel, H., Unke, O. T. & Müller, K.-R. Complete and efficient covariants for three-dimensional point configurations with application to learning molecular quantum properties. *J. Phys. Chem. Lett.* **15**, 12513–12519 (2024).
47. Fu, X. et al. Forces are not enough: Benchmark and critical evaluation for machine learning force fields with molecular simulations. *Trans. Mach. Learn. Res.* <https://openreview.net/forum?id=A8pqQipwkt> (2023).
48. Zhang, Y., Xia, J. & Jiang, B. Physically motivated recursively embedded atom neural networks: Incorporating local completeness and nonlocality. *Phys. Rev. Lett.* **127**, 156002 (2021).
49. Soper, A., Bruni, F. & Ricci, M. Site-site pair correlation functions of water from 25 to 400 c: Revised analysis of new and old diffraction data. *J. Chem. Phys.* **106**, 247–254 (1997).
50. Skinner, L. B., Benmore, C. J., Neufeind, J. C. & Parise, J. B. The structure of water around the compressibility minimum. *J. Chem. Phys.* **141**, 214507 (2014).
51. Marsalek, O. & Markland, T. E. Quantum dynamics and spectroscopy of ab initio liquid water: The interplay of nuclear and electronic quantum effects. *J. Phys. Chem. Lett.* **8**, 1545–1551 (2017).

52. Zhang, L. et al. End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems. In Bengio, S. et al. (eds.) *Advances in Neural Information Processing Systems*, vol. 31 (2018).
53. Gasteiger, J., Becker, F. & Günnemann, S. Gemnet: Universal directional graph neural networks for molecules. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W. (eds.) *Advances in Neural Information Processing Systems*, vol. 34, 6790–6802 (2021).
54. Liu, Y. et al. Spherical message passing for 3D molecular graphs. In *International Conference on Learning Representations* <https://openreview.net/forum?id=givsRXsOt9r> (2022).
55. Christensen, A. S. & von Lilienfeld, O. A. On the role of gradients for machine learning of molecular energies and forces. *Mach. Learn.: Sci. Technol.* **1**, 045018 (2020).
56. Musaelian, A. et al. Learning local equivariant representations for large-scale atomistic dynamics. *Nat. Commun.* **14**, 1–15 (2023).
57. Faber, F. A., Christensen, A. S., Huang, B. & von Lilienfeld, O. A. Alchemical and structural distribution based representation for universal quantum machine learning. *J. Chem. Phys.* **148**, 241717 (2018).
58. Stuart, S. J., Tutein, A. B. & Harrison, J. A. A reactive potential for hydrocarbons with intermolecular interactions. *J. Chem. Phys.* **112**, 6472–6486 (2000).
59. O'Connor, T. C., Andzelm, J. & Robbins, M. O. AIREBO-m: A reactive model for hydrocarbons at extreme pressures. *J. Chem. Phys.* **142**, 024903 (2015).
60. Los, J. H. & Fasolino, A. Intrinsic long-range bond-order potential for carbon: Performance in monte carlo simulations of graphitization. *Phys. Rev. B* **68**, 024107 (2003).
61. Lennard-Jones, J. E. Cohesion. *Proc. Phys. Soc.* **43**, 461 (1931).
62. Geim, A. K. & Grigorieva, I. V. Van der waals heterostructures. *Nature* **499**, 419–425 (2013).
63. Wen, M., Carr, S., Fang, S., Kaxiras, E. & Tadmor, E. B. Dihedral-angle-corrected registry-dependent interlayer potential for multilayer graphene structures. *Phys. Rev. B* **98**, 235404 (2018).
64. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865–3868 (1996).
65. Tkatchenko, A., DiStasio, R. A., Car, R. & Scheffler, M. Accurate and efficient method for many-body van der Waals interactions. *Phys. Rev. Lett.* **108**, 236402 (2012).
66. Paszke, A. et al. Pytorch: An imperative style, high-performance deep learning library. *ArXiv e-prints* 1912.01703 (2019).
67. pytorch-lightning (2024). <https://github.com/Lightning-AI/pytorch-lightning>. [Online; accessed 2. Oct. 2024].
68. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *ArXiv e-prints* 1412.6980 (2014).
69. Larsen, A. H. et al. The atomic simulation environment—a Python library for working with atoms. *J. Phys.: Condens. Matter* **29**, 273002 (2017).
70. Hu, W. et al. Forcenet: A graph neural network for large-scale quantum calculations. *ArXiv e-prints* 2103.01436 (2021).
71. Gasteiger, J., Yeshwanth, C. & Günnemann, S. Directional message passing on molecular graphs via synthetic coordinates. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W. (eds.) *Advances in Neural Information Processing Systems*, vol. 34, 15421–15433 (2021).
72. Haghighatlari, M. et al. Newtonnet: a newtonian message passing network for deep learning of interatomic potentials and forces. *Digit. Discov.* **1**, 333–343 (2022).

Acknowledgements

This work is supported by the National Science Foundation under Grant No. 2316667, and supported by the Center for HPC at the University of Electronic Science and Technology of China. This work also uses computational resources provided by the Research Computing Data Core at the University of Houston.

Author contributions

M.W.: project conceptualization, model development, data analysis, writing - original draft, writing - review, and supervision. W.F.H, J.D. and S.A.: data analysis and writing - review.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41524-025-01623-4>.

Correspondence and requests for materials should be addressed to Mingjian Wen.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025