Machine Learning: Science and Technology

PAPER • OPEN ACCESS

# Artificial neural network potentials for mechanics and fracture dynamics of two-dimensional crystals**

MACHINE
LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
19 January 2023

REVISED
22 March 2023

ACCEPTED FOR PUBLICATION
14 April 2023

PUBLISHED
4 July 2023

PAPER

# Artificial neural network potentials for mechanics and fracture dynamics of two-dimensional crystals[**]

**Gang Seob Jung[1,*], Hunjoo Myung[2] and Stephan Irle[1]**

[1] Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, United States of America
[2] Division of National Supercomputing, Korea Institute of Science and Technology Information, Daejeon, Republic of Korea
[*] Author to whom any correspondence should be addressed.

**E-mail:** jungg@ornl.gov

**Keywords:** neural network potential, graphene, 2D materials, mechanics, fracture, crack propagation

Supplementary material for this article is available online

## Abstract

Understanding the mechanics and failure of materials at the nanoscale is critical for their engineering and applications. The accurate atomistic modeling of brittle failure with crack propagation in covalent crystals requires a quantum mechanics-based description of individual bond-breaking events. Artificial neural network potentials (NNPs) have emerged to overcome the traditional, physics-based modeling tradeoff between accuracy and accessible time and length scales. Previous studies have shown successful applications of NNPs for describing the structure and dynamics of molecular systems and amorphous or liquid phases of materials. However, their application to deformation and failure processes in materials is still uncommon. In this study, we discuss the apparent limitations of NNPs for the description of deformation and fracture under loadings and propose a way to generate and select training data for their employment in simulations of deformation and fracture simulations of crystals. We applied the proposed approach to 2D crystalline graphene, utilizing the density-functional tight-binding method for more efficient and extensive data generation in place of density functional theory. Then, we explored how the data selection affects the accuracy of the developed artificial NNPs. It revealed that NNP's reliability should not only be measured based on the total energy and atomic force comparisons for reference structures but also utilize comparisons for physical properties, e.g. stress–strain curves and geometric deformation. In sharp contrast to popular reactive bond order potentials, our optimized NNP predicts straight crack propagation in graphene along both armchair and zigzag (ZZ) lattice directions, as well as higher fracture toughness of ZZ edge direction. Our study provides significant insight into crack propagation mechanisms on atomic scales and highlights strategies for NNP developments of broader materials.

## 1. Introduction

Understanding fracture mechanics and crack propagation is key to predicting and controlling mechanical behaviors for materials processing and subsequent materials applications. In many materials, crack propagation under loading is an overriding failure and, therefore, one of the critical problems in materials science. Accurate computational modeling of crack propagation, thus, becomes an essential tool for

[**] Notice: This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

understanding the underlying mechanisms [1–3] and will open pathways for the computational design of advanced materials processing [4].

Crack propagation in covalent crystals is a multiscale phenomenon requiring high atomic-level accuracy but large-length scales for computational modeling. However, physics-based molecular simulations performed on associated atomic-scale potential energy surfaces are often limited by the fundamental tradeoff between the simulation accuracy on one hand and accessible time and length scales on the other. For example, highly accurate quantum mechanics-based electronic structure approaches such as density functional theory (DFT) are computationally demanding and generally impractical for long-timescale simulation of complex and large systems. On the other hand, classical molecular mechanics-based forcefields, including reactive forcefields, can cover larger system sizes, but their accuracy and transferability in specific applications are often inferior. Recently, machine learning (ML) techniques have revolutionized problem-solving approaches in a wide range of fields of science and engineering [5–9]. Significantly, ML potentials have emerged as a possible solution to resolve the conflicts arising from such tradeoffs [10].

There are many types of ML potentials proposed and developed [11–13]. Among them, the so-called 'neural network' (NN) is a ML method that mimics the structures and function of the human brain to process data. NNs have been applied to various problems, such as classification, predictions, and pattern recognition. NN-based potentials (NNPs) utilize NNs to fit the interatomic interaction energies [14], which have rapidly emerged due to their flexibility, accuracy, and efficiency, and they are more suitable for large and complex systems than other types of ML potentials as they can handle large data sets with many training data points.

NNPs rely on mapping or encoding the local chemical environment of atoms in a molecular or bulk system to a unique 'input feature.' Recent advances expand these input features from local descriptors to include long-range and dispersion interactions [11, 12]. Such efforts and achievements are promising for NNPs to describe complex systems accurately and efficiently. One of the most common approaches to the local atomic environment descriptors, symmetry functions, goes back to the early work of Behler and Parrinello (BP) [15]. BP symmetry functions use translational and rotational invariant representations of atomic geometry and can be readily expanded to multiple chemical element types [16, 17]. More recently, convolutional NNPs have been proposed [18], showing overall better accuracy than BP-NNPs but requiring a higher computational cost. Another class of ML potentials, namely kernel-based approaches, including Gaussian approximation potential [19], spectral neighbor analysis potential [20], and moment tensor potential [21], offer good performance with a small number of data sets. Still, the computational time increases with the number of reference data. NNPs can achieve high accuracy, comparable to the kernel-based methods, but faster with a large number of data [13] (e.g. RMSE of energy under 1 kcal mol$^{-1}$ with extended MD17 data set [22]).

While many studies have used NNPs for various applications [10], mechanical properties related to the deformation and fracture of materials were less commonly explored. Recent reports clearly show that the generation of training data for a wide range of configurations and phases for NNPs is not straightforward and requires careful evaluation of the representability contained in the data set [23, 24]. Thus, a critical issue in addressing fracture and deformation from NNPs is the generation and selection of the training data set. Configurations driven by mechanical forces, e.g. tensile or shear loadings, cannot be obtained from the equilibrium molecular dynamics (MDs) simulations or normal mode analysis [16]. Instead, one should perform deformation simulations until failure occurs to obtain appropriate training data that will allow the NNP to accurately reproduce these processes. Another important challenge stems from the fact that the moment of failure or fracture is a rare event in the entire loading history, so naturally, a data imbalance problem emerges.

In this study, we developed an NNP for one of the most famous two-dimensional (2D) materials, crystalline graphene, to describe its crack propagation, and compared the results with previous experimental observations. Graphene frequently plays a role as a popular 'test bed' when new interatomic potentials are developed, especially related to the prediction of mechanical properties and vibrational properties to examine their reliability. Many-body potentials and empirical reactive forcefields such as adptive intermolecular reactive empirical bond order (AIREBO) [25, 26] have been widely utilized for crack propagation and fracture dynamics [27].

There are well-known limitations of empirical potentials for bond breaking, and therefore, many ways to correct them have been proposed. Most approaches are based on avoiding stiffening effects from the switching function in potentials [28–30]. A more detailed example of the switching function in AIREBO on bond breaking is well described in a previous review [31]. If we choose the suitable cutoffs of the switching function, AIREBO can predict the mechanical strength of graphene and stress–strain curves along both zigzag (ZZ) and armchair (AC) directions. However, crack propagation from AIREBO along the AC

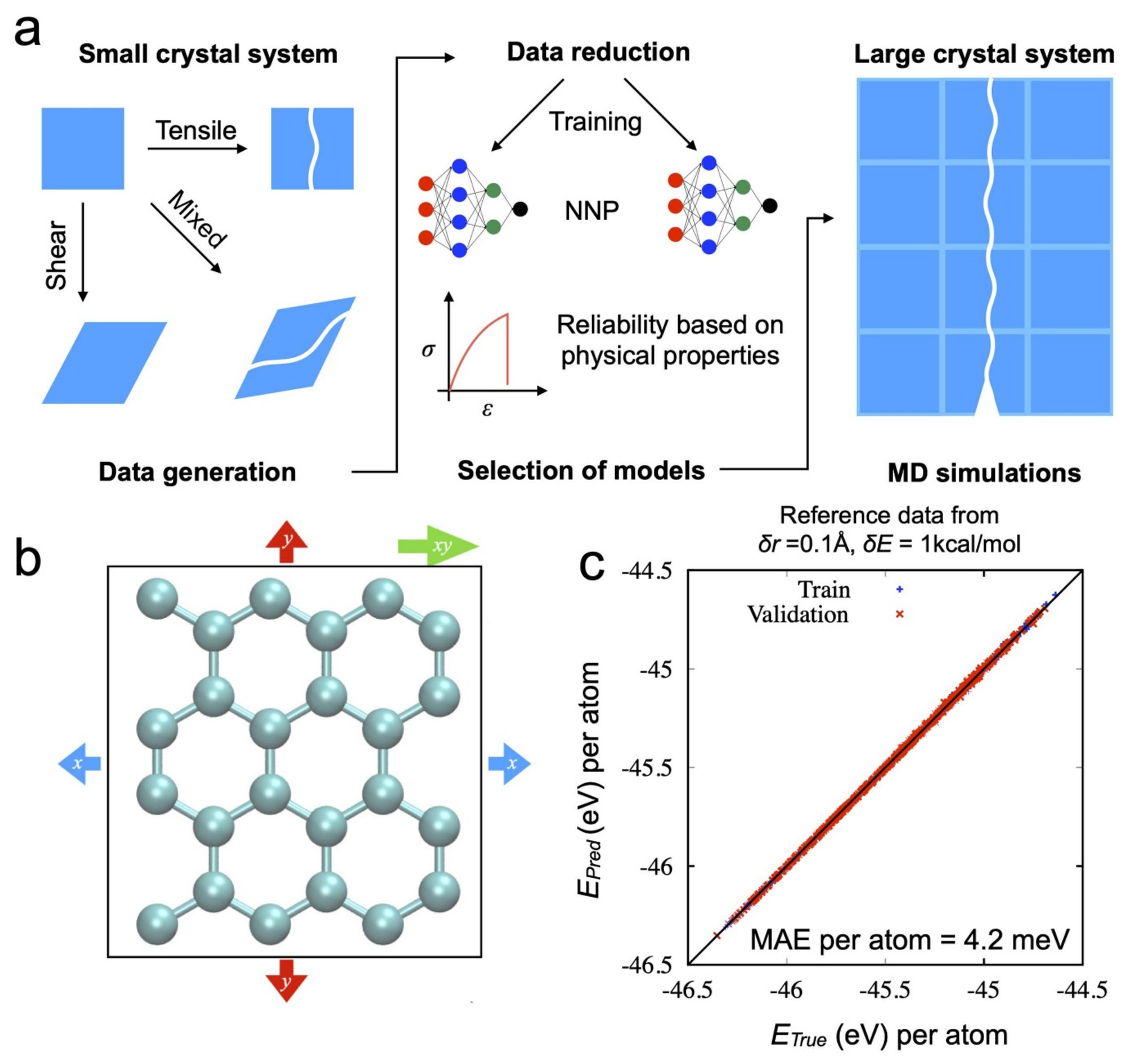


**Figure 1.** (a) Schematic of current study from the data generation to MD simulations. Data are generated for possible fracture scenarios of graphene with a small size. The data are reduced and selected based on physical properties. The selected NNP is utilized for the crack propagation of graphene with a sharp crack. (b) Small crystal system of 24 atoms under three main loadings. The size was selected to avoid self-interaction when we consider the radius cutoff of local descriptors of NNP (c) Energy comparison from the training and validation sets.

direction does not occur in a straight line, even though a previous high-resolution transmission electron microscope (TEM) study reported both straight AC- and ZZ-torn graphene [32].

Two-dimensional materials are ideal for validating atomistic models of materials failure under the effects of vacancies, bilayer, crack directions, and folding by comparing the fracture patterns observed through advanced TEM techniques [33–36]. A recent *in-situ* TEM with reactive molecular simulation has shown that lattice distortion at the crack tip in 2D materials, $WS_2$ with a hexagonal lattice, can drastically change the entire path of crack propagation [35]. Also, theoretical comparisons between DFT and AIREBO show that the lattice distortion under tensile loading is significantly different [37]. Therefore, we hypothesized that developing an NNP of graphene for lattice deformation and fracture under various loadings could provide a more faithful prediction of crack propagation in graphene, especially along the AC direction. Furthermore, establishing a data preparation process can provide a foundational workflow to develop NNPs for mechanics and fractures of other materials.

A schematic of the current study is shown in figure 1(a). In a previous study [37], we showed that DFTB could correctly predict both mechanical behaviors and lattice deformation of graphene under various loading conditions, in good agreement with DFT calculations. Therefore, we here utilized DFTB to generate extensive data for possible fracture scenarios of graphene under various loadings by mixing two tensile and one shear deformation. The data are reduced and selected based on deformation and energy differences to improve the generalization during the training. Then, we explored how the data selection affects the accuracy of the trained NNPs and the reliability evaluated based on physical properties such as deformation and stress–strain curves. In the end, we performed simulations for the crack propagation of graphene with a sharp crack using the trained NNP. We compared the results with an approach based on a popular empirical bond order potential, AIREBO. The results show better agreement with previous experiments regarding the resulting edge structures and the frequency of the type of torn edges.

## 2. Method

### 2.1. Generation of data

The MDs simulations with DFTB for data generation were performed via the large-scale atomic/molecular massively parallel simulator (LAMMPS) package [38]. DFTB calculations were performed at each time step through the DFTB+ package [39], utilizing a previously developed interface for LAMMPS [40]. We employed the 3OB [41] C–C parameters with the DFTB3 scheme because the stress–strain behaviors are well-matched with those from DFT calculations with the PBE functional [40]. We generated 10 000 data points through the NVT ensemble at 400 K with the time step of 0.5 fs to obtain reference RMSE of the relative energy from the canonical data generation.

Then, we obtained data points for the deformation and fracture under various loading. Static loading or quasi-static loading with full energy minimization has a limitation for the training data set because all atoms are located in the energy minimized positions. We need slightly perturbated coordinates to train the NNP to distinguish the contribution of a single atomic energy. Therefore, we utilized dynamics loading for the data generation. We consider different loading directions by mixing the loadings along *x*, *y* (pure tensile), and *xy* (pure shearing) directions as ($v_x$, $v_y$, $v_{xy}$). We prepared 361 directions with constant velocity 400 m s$^{-1}$ with 0.5 fs time step. Each direction has 2000 steps, so the total data number is 722 000. Here, the loading speed is much faster than what is desired to provide reliable behaviors, which is under 20 m s$^{-1}$ [40]. The effect of the loading speed becomes critical when the speed is too fast for the system to have enough time to relax the structures under the deformation. So, after every 20 steps, we included a small step of energy minimization to overcome the delay. The number of steps was tuned to match the stress–strain curve under shear loading to the results from the quasi-static loading. At each step, we deformed the simulation box by about 0.002 Å, resulting in a total deformation for each loading of about 4 Å.

### 2.2. Selection of data

From the 722 000 data points, we built neighbor lists between data points based on the deformation of the simulation box (d*x*, d*y*, d*xy*). We consider the distance ($\delta r$) between data as an indicator of the deformation similarity. Then, we sequentially deleted the data but saved it if the energy difference in the list is larger than $\delta E$. We utilized vales of $\delta r$ (0.01 Å–0.2 Å) and $\delta E$ (1 kcal mol$^{-1}$–50 kcal mol$^{-1}$) from the original 722 000 data points, and the number of reduced data from ($\delta r$, $\delta E$) is listed in table 1.

### 2.3. Training

We utilized TorchANI library and its setting for training the NNs. For training, 80% of data was used, while 20% of data was utilized for validation with a small mini-batch size of 64. We note that we did not explicitly prepare a specific test set in this study because we utilized most hyperparameters suggested in the previous study of TorchANI [42]. Instead, we evaluated the performance of each model by the same reference data ($\delta r = 0.01$ Å, $\delta E = 1$ kcal mol$^{-1}$, ∼450 k). Therefore, our selected model from the data set (e.g. $\delta r = 0.1$ Å, $\delta E = 1$ kcal mol$^{-1}$, ∼250 k) was trained without 200 k data in both training and validation for their final performance.

The loss function is defined as

$$Loss = \frac{1}{N_{\text{data}}}\sum\frac{\left(E_{\text{NNP}}-E_{\text{ref}}\right)^2}{\sqrt{N_{\text{atom}}}} + \frac{\alpha}{N_{\text{data}}}\sum\frac{\left(\vec{F}_{\text{NNP}}-\vec{F}_{\text{ref}}\right)^2}{N_{\text{atom}}}, \tag{1}$$

where $\alpha$ is a parameter to determine the contribution of forces, and we used 0.1. The Adam optimizer was utilized with weight decay for the weights [43, 44] (weight decays for two hidden layers are set $1 \times 10^{-5}$ and $1 \times 10^{-6}$, respectively, others are default values) and stochastic gradient descent (SGD) [45] for the biases (learning rate = 0.001, others are default values), as suggested in the previous study. The weights were initialized by Kaiming initialization [46] with the normal distribution, and the initial values of biases were zeros. We utilized a learning rate scheduler (a function 'reduced lr on plateau' in Pytorch) for both Adam and SGD with factor = 0.5, patience = 100, and threshold = 0. Others are default values). We took the best model for the validation set during the entire epochs.

### 2.4. MDs simulations for crack propagation

We prepared the pristine graphene of 10 nm by 20 nm with a 2 nm sharp crack along the *y* direction with periodic boundary conditions. To avoid the interaction between image cells, 20 nm space along the *y*

**Table 1.** The numbers of data selected from the initial 722 000 data points.

| | $\delta E$ (kcal mol$^{-1}$) | | | | |
|---|---|---|---|---|---|
| $\delta r$ (Å) | 1 | 2 | 5 | 10 | 20 |
| 0.01 | 450 k | 270 k | 130 k | 88 k | 80 k |
| 0.02 | 430 k | 260 k | 120 k | 66 k | 40 k |
| 0.05 | 360 k | 200 k | 100 k | 57 k | 29 k |
| 0.1 | 250 k | 150 k | 78 k | 37 k | 20 k |
| 0.2 | 120 k | 70 k | 33 k | 18 k | 9 k |

direction and 3.35 nm space along the $z$ direction were inserted. Instead of full dynamic loading, we applied a 0.01 strain at each iteration to stretch to a 0.04 strain along the $x$ direction with structural relaxations. Then, we applied dynamic tensile loading along the $x$ direction at low temperature, 10 Kelvin. The loading speed was 2.0 m s$^{-1}$, and the time step was set to 1 fs.

### 2.5. PyTorch interface with LAMMPS

We utilize the python functions in LAMMPS (v. 29OCT20) to utilize the python code in the python environment. Through the python environment, we can easily call python library. Utilizing TorchANI, we calculate the atomic environmental vectors (AEV) for NNs. Forces and stress are calculated with the given coordinates and simulation box through the autograd engine in PyTorch [47] and updated at each time step.

## 3. Results and discussions

One of the most critical parts of NNP development is the training data set. The data for training should cover the essential features of the problem-specific configurations. Previous NNPs have been trained from the data usually generated from first principles DFT-based MDs simulations and conformal searches based on the normal mode analysis [15, 16, 48]. The initial training set is generally not sufficient for the desired accuracy. Therefore, adaptive and active learning approaches have been proposed and applied [49, 50]. The basic principle of such methods is to detect new data not contained in the initial training data set by analyzing the data using configuration fingerprints or comparing values from multiple models of NNPs, an ensemble, or a so-called committee. Iteratively searching the new data, training, and sampling provide better data sets in the end. However, these approaches are not sufficiently good for fracture dynamics if the initial data set is not accurate enough to describe the dynamics during the failure.

We first tested previously trained models of ANI-1x [16], ANI-1cxx [51], and ANI-2x [52] as provided in the TorchANI library [42], where the data sets include the deformed geometries of small organic molecules from normal mode sampling. We examined the reliability with a single-layer graphene system consisting of 24 atoms under three different loadings (shown in figure 1(b)) by evaluating stress–strain curves and the deformation of bond length ($l_1$, $l_2$, and $l_3$) and angles ($\theta_1$, $\theta_2$, and $\theta_3$). Since we utilized local descriptors with a radius cutoff ∼5 Å in TorchANI, the unit cell of 24 atoms with system size (∼7.4 Å × 8.5 Å) was selected as a compromise to avoid self-interaction effects on one hand and allow sufficiently large data creation with DFTB for training the NNP. A newly developed PyTorch interface in LAMMPS was utilized for communicating energy, forces, and stress with the TorchANI python library. Figures S1–S3 show the results of stress–strain curves and the deformation through ANI models.

We note that the data sets do not explicitly include graphene information, but interestingly, the NNPs from ANIs can well describe graphene's behaviors under small deformation. As expected, however, they clearly fail for fracture behaviors and large deformation. Therefore, we designed data generation by mixing three loading directions along the $x$, $y$, and $xy$ directions and generated more than 700 000 data points. Then, we trained new NNPs from scratch using the TorchANI [42] tool. The structure of the utilized NNP in the current study is shown in figure 2 and the comparison between other ANI models is in table S1. From the actual coordinates ($\boldsymbol{q}$), AEV (also known as a symmetry function, $\boldsymbol{G}$) are utilized as input to be invariant under translation and rotational transformation and the permutation of the same atom types [15]. There are two parts in the AEV: radial and angular terms from two atoms ($i$ and $j$ with distance $R_{ij}$) and three atoms ($i$, $j$, and $k$ with two distances $R_{ij}$, $R_{ik}$, and one $\theta_{ijk}$), respectively [42]:

$$G_m^R = \sum_{j\neq i}^{\text{all atoms}} e^{-\eta\left(R_{ij}-R_s^{(k)}\right)^2} f_C\left(R_{ij}\right), \tag{2}$$

$$G_m^A = 2^{1-\zeta}\sum_{j,k\neq i}^{\text{all atoms}}\left(1+\cos\left(\theta_{ijk}-\theta_s^{(q)}\right)\right)^{\zeta}\times e^{-\eta\left(0.5\left[R_{ij}+R_{ik}\right]-R_s^{(p)}\right)^2} f_C\left(R_{ij}\right) f_C\left(R_{ik}\right), \tag{3}$$

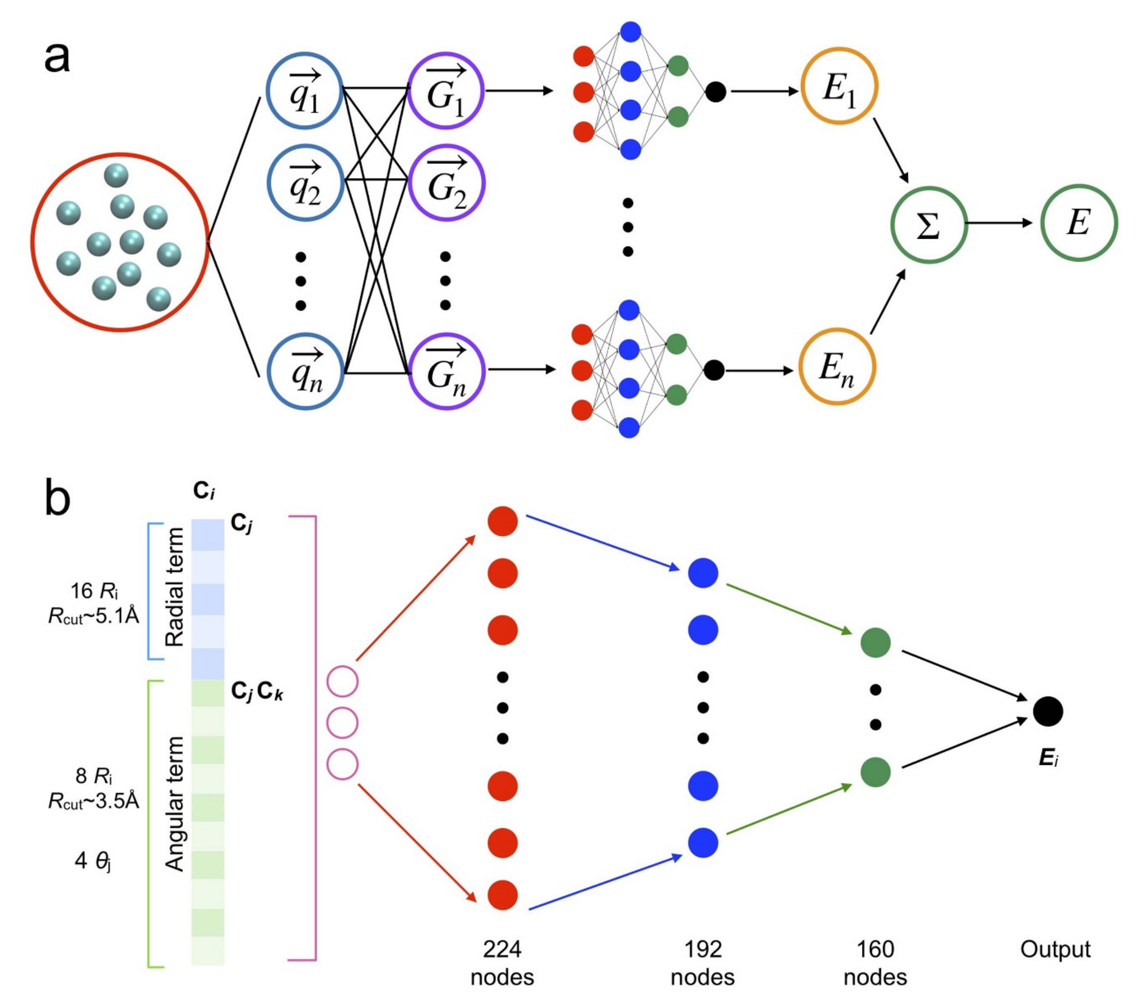


**Figure 2.** (a) Structure of neural network potential based on the symmetry functions or atomic environmental vector (AEV). AEV ($\boldsymbol{G}$) is calculated from the three-dimensional atomic coordinates ($\boldsymbol{q}$) and used as an input for the neural network to obtain atomic energy. (b) Dimension of AEV with radial and angular terms from carbon system and details of neural network structure with layers and nodes.

where $\eta$ controls the width of Gaussian function with multiple $R_s$ for probing specified radial environments ($m$ is an index for $R_s$); $\zeta$ controls the width of probing as $\eta$; $\theta_s$ decides the specific region in the angular environments as $R_s$. $f_C$ is a cutoff function to change values to zero at $R_C$ smoothly, defined as $f_C(R) = 0.5\left[\cos\frac{\pi R}{R_C} + 1\right]$ for $R \leqslant R_C$ and 0 for $R > R_C$. AEV from each atom is an input for the NN to estimate the single atomic energy. We followed the parameters of AEV and the structures of nodes and layers from ANI-2x: the radial term has 16 different radii, and the angular term has 4 angles and 8 radii [52] shown in figure 2(b). The NN has three hidden layers with Gaussian error linear unit activation function [53] to add non-linearity.

Next, we checked the performance of the NN structures and training processes from the data generated from graphene's MD simulation at equilibrium states at 400 Kelvin. We utilized a root mean square error (RMSE) for the evaluation of the accuracy for both energy and force components as a kcal mol$^{-1}$ energy unit. Because some previous works utilized a mean absolute error (MAE) with meV atom$^{-1}$ energy unit as metrics, we also added the MAE values with meV atom$^{-1}$. A RMSE in the relative energy lower than 0.8 kcal mol$^{-1}$ (RMSE = 1.5 meV atom$^{-1}$ $\sim$ 0.034 kcal mol$^{-1}$ atom$^{-1}$, MAE = 0.6 meV atom$^{-1}$ $\sim$ 0.014 kcal mol$^{-1}$ atom$^{-1}$) is achieved around 200 epochs, which we consider a very high accuracy from the perspective of computational chemistry, where a threshold 1 kcal mol$^{-1}$ (for small molecules with $\sim$20 atoms) in accuracy is commonly considered a 'gold standard'. However, our result clearly shows that an agreement between NNP and ground truth data in terms of relative energy does not guarantee correct physical properties. As shown in figure S4, the failure behaviors are not correctly described.

Figure 3(a) shows a naïve way to save data under loading, recording data based on constant deformation ($\delta r$) or at a constant time interval. There are two apparent problems with this approach. First, it is likely to

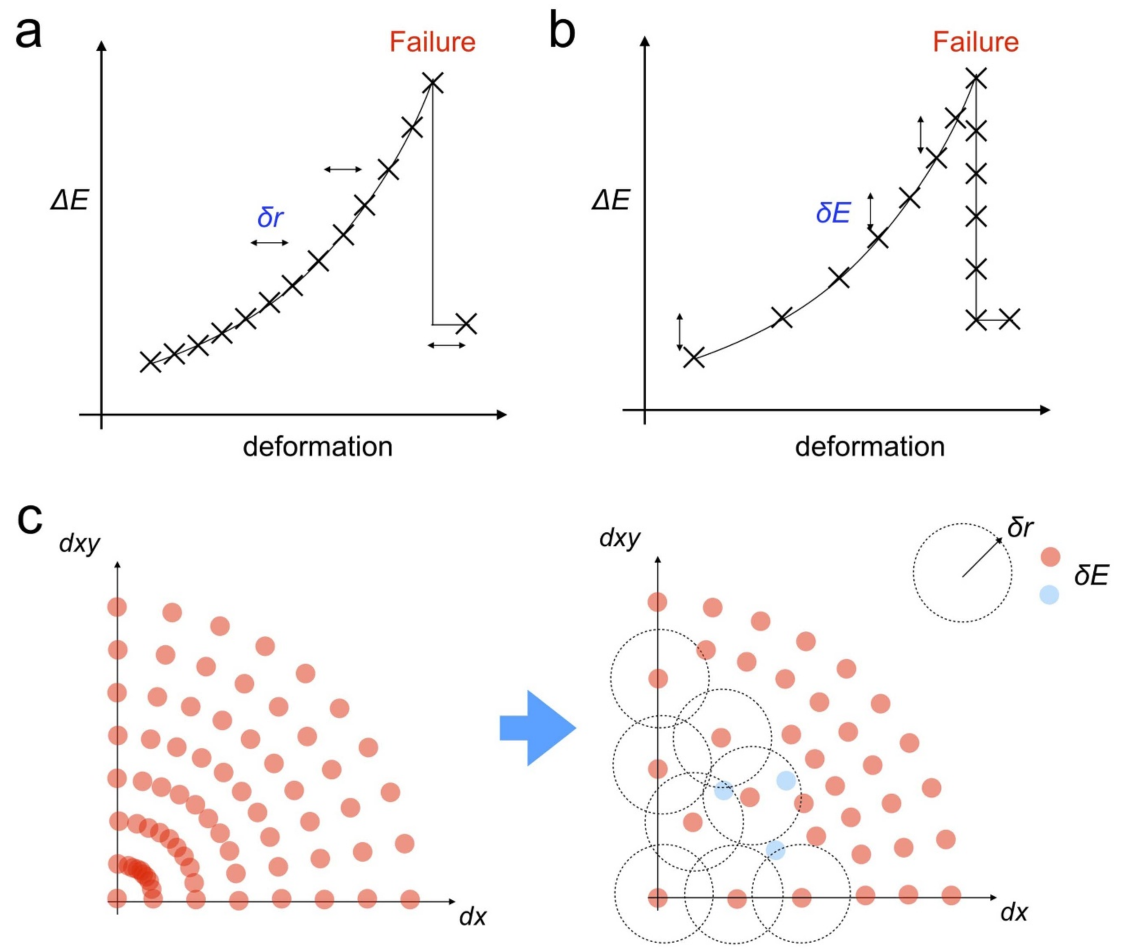


**Figure 3.** (a) and (b) Schematics of data distribution under loading with $\delta r$ and $\delta E$. Data with a constant deformation or space ($\delta$r), it would lose the important data of the failure. Instead, energy difference ($\delta E$) based data selection will conserve the key data and reduce data near the equilibrium state. (c) Example of data distributions with the two loading directions to show a practical method to achieve data selection based on $\delta E$. First, neighbor lists are built with a defined cut-off, $\delta r$. The data of which the energy difference is smaller than $\delta E$ are deleted from the list.

miss essential data during the failure process, where the configurations drastically change in a very short time. Second, the many similar data are close to each other near the non-deformed structures, which probably hinders the training due to the data imbalance [54]. Therefore, we utilized a constant energy difference ($\delta E$) to select data, as shown in figure 3(b). It would have better chances to capture the key data during the failure process even with the same number of data points. Figure 3(c) shows the schematic to represent the data distributions with the two main directions of the loadings: $x$ and $xy$. From the total data, we built neighbor lists of each data point with a defined cutoff in the deformation space, $\delta r$. Then, the data in the neighbor list is sequentially removed if the energy difference is not bigger than the predefined criterion, $\delta E$. Table 1 lists the parameters: $\delta r$ and $\delta E$ with the screened data numbers.

We checked the stress–strain curves of the trained model from all data without any reduction, as shown in figure S5. As expected, it shows much better behaviors than ANIs or the trained model from NVT ensemble trajectories because the current data set explicitly includes various fracture scenarios. However, it fails to describe the fracture patterns and stress–strain after fracture along the $x$ direction loading in figure S5(a). The energy minimization during the quasi-static loading should result in complete bond breaking between broken edges. Figures 4(a) and (b) show the RMSE of energy and force components from training, validation, and total data from each data set selected from the above-mentioned approach as varying $\delta r$ with fixed $\delta E = 1$ kcal mol$^{-1}$. We note that the data from larger $\delta r$ is selected from the data set of smaller $\delta r$, which means the smaller data sets always belong to the larger data sets. Therefore, the difference of RMSE, e.g. better accuracy of $\delta r = 0.1$ Å than that of $\delta r = 0.05$ Å, does not come from the data difference but from better generalization with reducing the overfitting (See SI discussion). We assume the original data set has too many similar data to prevent generalization. So, we evaluate the accuracy of all trained models based on the first selected data set from $\delta r = 0.01$ Å and $\delta E = 1$ kcal mol$^{-1}$ as shown in figures 4(c) and (d).

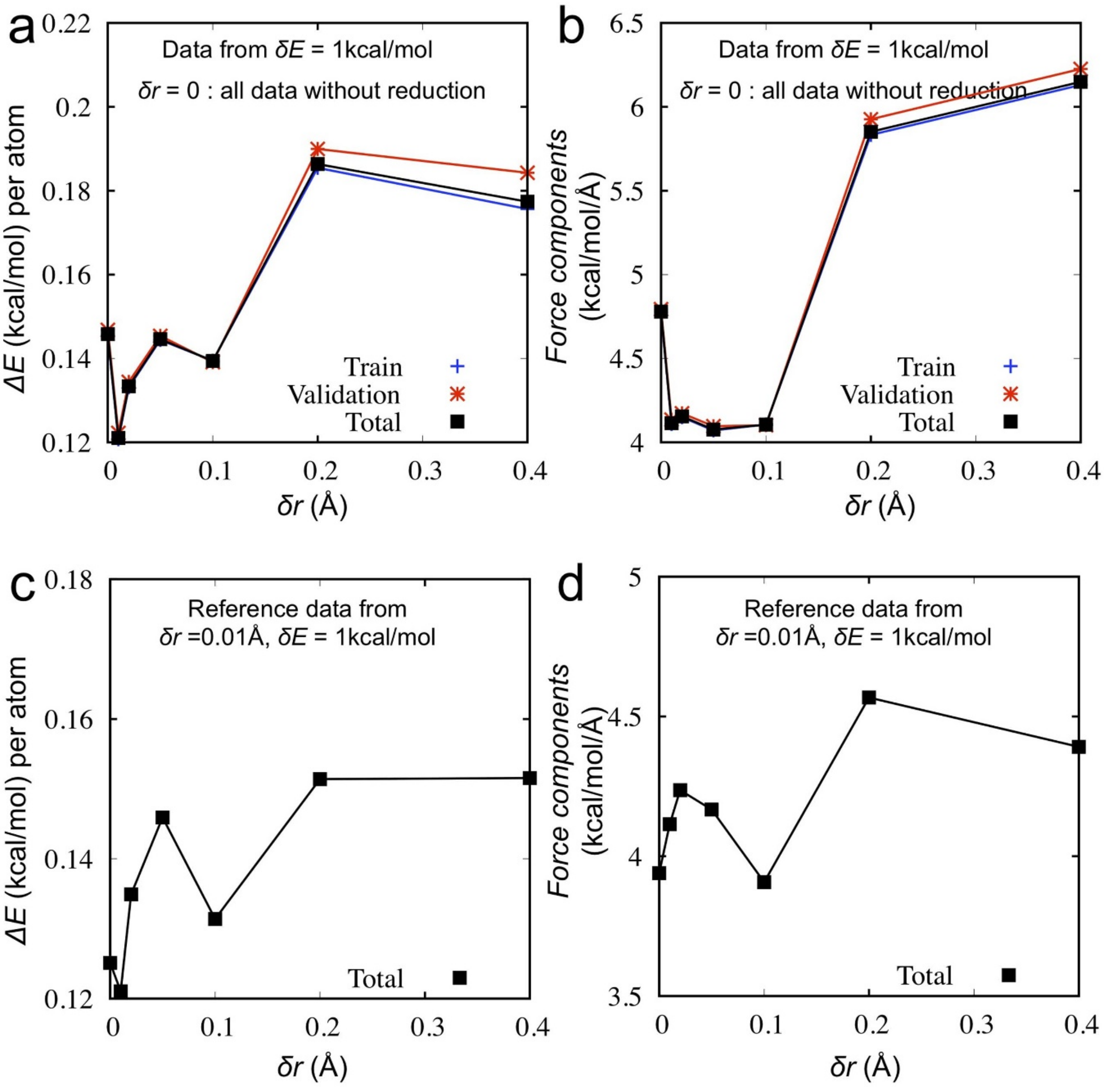


**Figure 4.** RMSE of relative energy per atom (a) and force components (b) from training, validation, and total data from each data set selected from $\delta r$ with fixed $\delta E = 1$ kcal mol$^{-1}$. $\delta r = 0$ indicates the entire data points without reduction. Reevaluation RMSE of relative energy (c) and force components (d) with the same data set ($\delta r = 0.01$ Å, $\delta E = 1$ kcal mol$^{-1}$). RMSEs depend on data set, and even smaller numbers of data can have higher accuracy. Also, RMSE does not guarantee the physical properties of the models.

The RMSE of relative energy from the model ($\delta r = 0.01$ Å, $\delta E = 1$ kcal mol$^{-1}$) shows the lowest value, but RMSE of force components from the model ($\delta r = 0.1$ Å, $\delta E = 1$ kcal mol$^{-1}$) show the lowest value. Also, we investigated the stress–strain curves and deformation for the reliability of the trained NNPs. The models trained from $\delta r = 0.01$ Å to $\delta r = 0.05$ Å data have problems near the fracture point, as indicated with arrows in figures S6–S8 (See figures S9-S11 for other conditions). In terms of the stress–strain curves and deformation, the trained model from $\delta r = 0.2$ Å looks better than that from $\delta r = 0.1$ Å, but the model from $\delta r = 0.1$ Å was selected for the next stage because the RMSE of force components exhibits the lowest value. This is a reasonable choice because the RMSE of atomic force is a good indicator for overfitting (see SI discussion). We also tested how the choice of $\delta E$ affects the accuracy of models with $\delta r$ lower than 0.1, as shown in figure S12. Since the number of data decreases as the value of $\delta E$ increases, it is expected to lose accuracy. However, this also does not monotonically decrease, and data with $\delta r = 0.1$ Å and $\delta E = 5$ kcal mol$^{-1}$ is reasonably optimal with the number of data points, 78 000 (= 78 k). Considering the fact that the reference data (∼450 k) is six times larger, the loss of accuracy of energy and force components are only 0.02 kcal mol$^{-1}$ atom$^{-1}$ (RMSE = 0.9 meV atom$^{-1}$) and 0.12 kcal mol$^{-1}$ Å$^{-1}$ (5 meV Å$^{-1}$), respectively. Also, the NNP describes the stress–strain and deformation under three loadings very well, as shown in figure S13. This kind of data reduction without losing the essential data is important for active

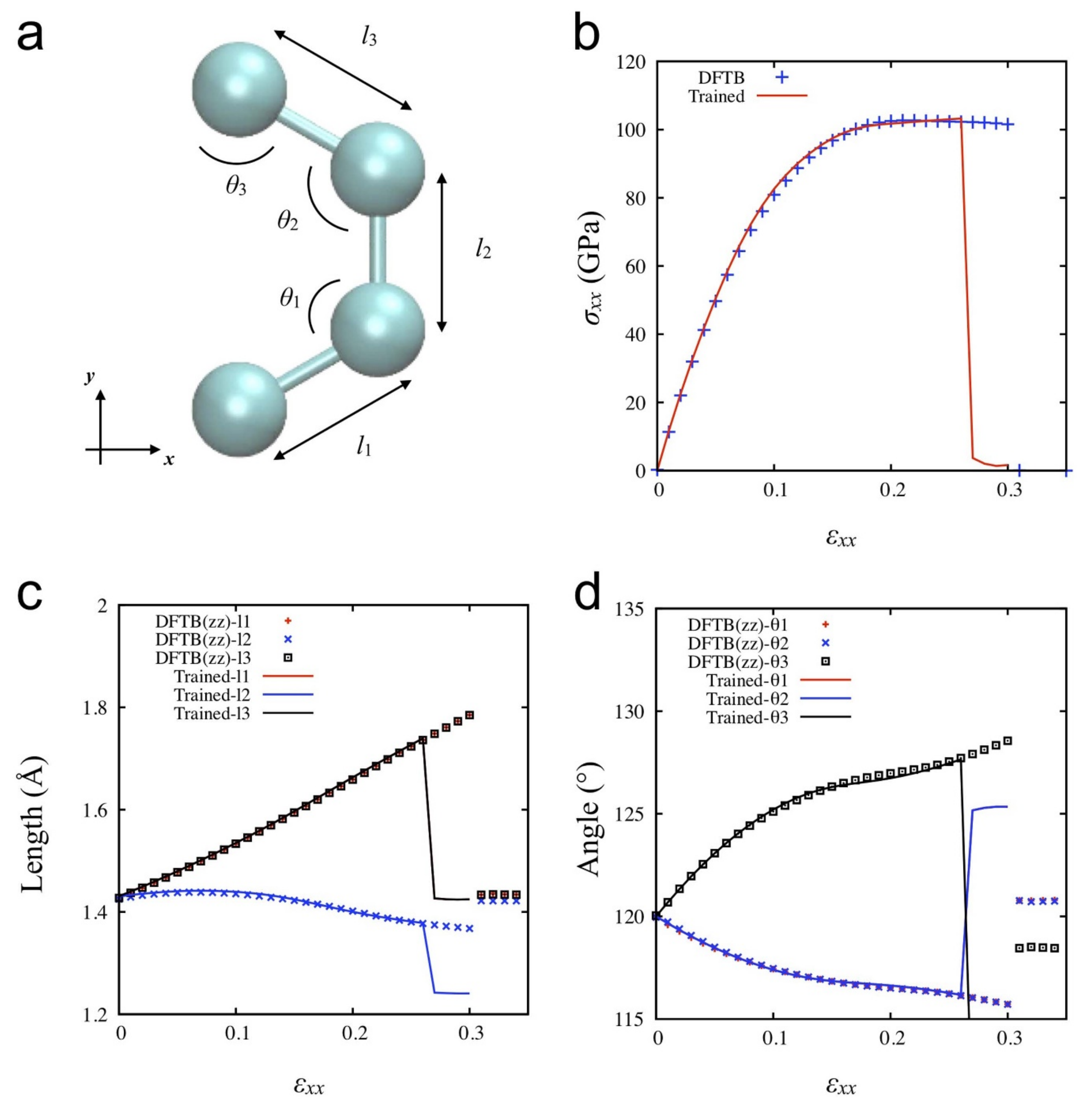


**Figure 5.** (a) Definition of three different bond lengths and angles to evaluate the deformation under loadings (b) Stress–strain curves from the trained model with the reduced data ($\delta r = 0.1$ Å, $\delta E = 1$ kcal mol$^{-1}$) with DFTB reference points under the loading along the $x$ direction from selected NNP and DFTB calculations. (c) The bond lengths during the deformation under the $x$ direction in panel **a**. The bond ($l_2$) gets shorter as the strain increases ($\varepsilon_{xx}$). (d) The angles during the deformation under the $x$ direction. The other directions are shown in figure S9.

learning. However, such data augmentation is out of the current scope, and we selected the model from the data ($\delta r = 0.1$ Å, $\delta E = 1$ kcal mol$^{-1}$) for the next simulations. Figure 5 shows the stress-strain curves and deformation of the selected model under the loading along the x direction.

A previous microscopy study reported both straight AC and ZZ torn graphene edges through the high-resolution TEM [32]. Also, torn lines along the AC edge are twice more frequently observed than the torn lines along the ZZ edge [55]. In previous theoretical studies, ReaxFF and AIREBO have been utilized to describe the mechanics and crack propagation from atomistic modeling. However, ReaxFF has some limitations in describing mechanics and stress–strain curves near failure compared to DFT calculations [56]. Especially, brittle crack propagation is hindered by the stiffening effect near the point of failure. Instead, AIREBO is preferred because the stress–strain curves of pristine graphene are well-matched with DFT calculations once its bond order switching function is controlled [31]. MD simulations of pristine graphene with AIREBO consistently show that the fracture toughness along the ZZ edge is lower than the AC edge [57–59]. Also, once the crack propagates along the AC direction, the fracture pattern from the AIREBO shows the ZZ torn edge preference. Although nanopore formation through the electron beam prefer to form the ZZ edge [60], the configuration does not come from the mechanics or crack preference but from the kinetic stability during the reconstructions [61]. This shows that empirical forcefields are limited for predicting pristine graphene's torn edge configuration and the dynamics of crack propagation.

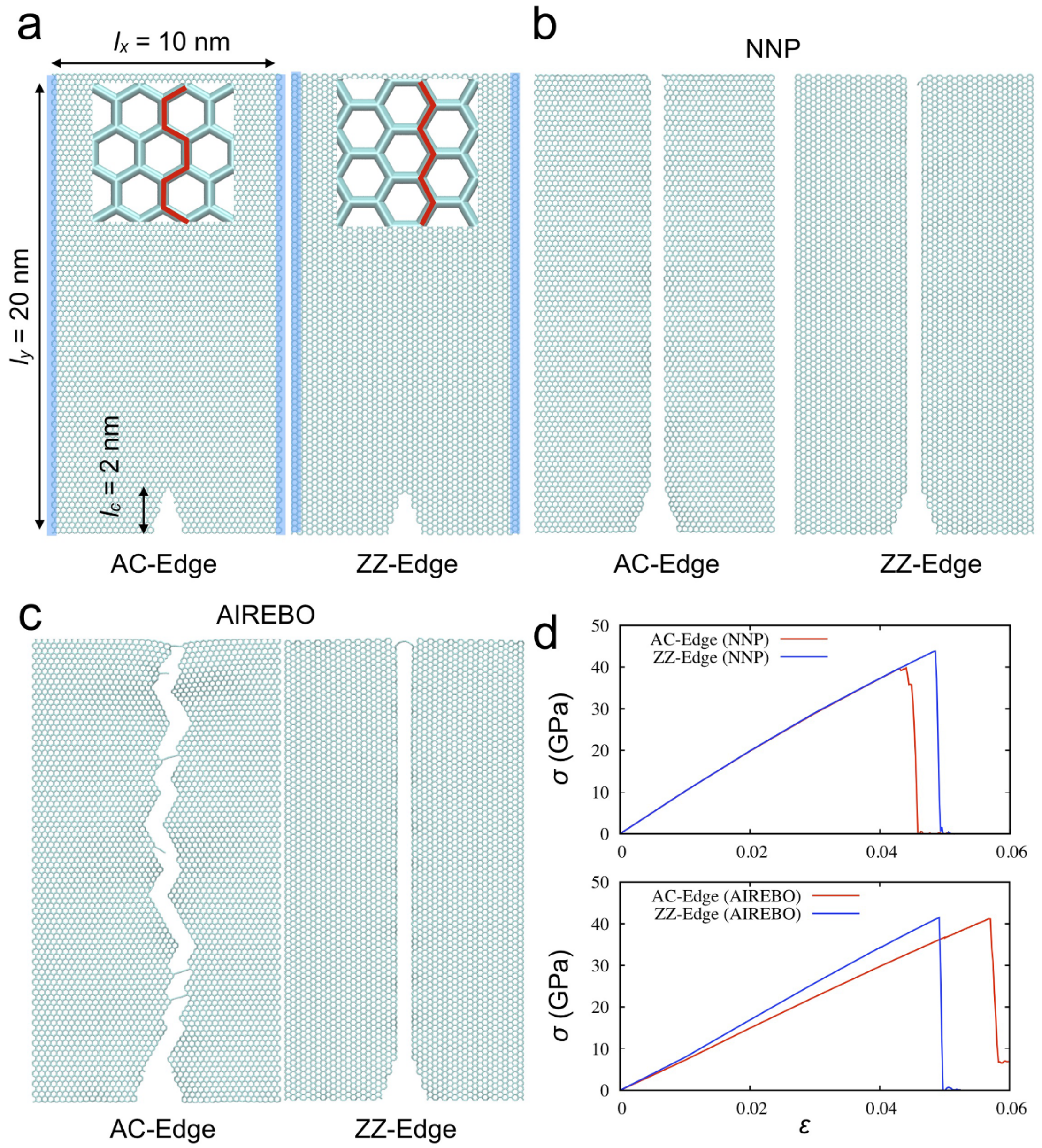


**Figure 6.** (a) Initial geometries of crack inserted graphene along the armchair and zigzag edges to evaluate the crack propagation (b) Fracture pattern of NNP after crack propagations (see movies 1 and 2). (c) Fracture pattern of NNP after crack propagations (see movies 3 and 4). (d) The obtained stress–strain curves from NNP and AIREBO.

Finally, we performed MD simulations using both selected NNP and AIREBO to test our hypothesis for the crack propagation along the AC direction with a rectangular system with 10 nm × 20 nm with a crack of 2 nm, as shown in figure 6. We performed the crack propagation simulations by combining quasi-static loading and dynamic loading to reduce the computational cost. The NNP results in a straight and clean torn edge in both AC and ZZ crack direction in figure 6 (see movies 1 and 2). As described above, AIREBO predicts that the straight propagation along the AC direction is less likely to occur. Instead, it shows meandering crack paths in figure 6 (see movies 3 and 4) with ZZ crack edges. The obtained stress–strain curves from NNP and AIREBO are shown in figure 6. The notable difference here is the fracture toughness. We estimated the critical energy release rate and fracture toughness in both AC and ZZ crack directions. Considering the brittle behaviors of the stress–strain curve, we can directly estimate the external work as the critical energy release rate ($G_c$) from the critical stress ($\sigma_c$, GPa) and the critical elongation ($\Delta l$, nm) as

$$G_c = 0.5\frac{\sigma_c l_y \Delta l}{\Delta a}\left(\mathrm{Jm}^{-2}\right), \tag{4}$$

**Table 2.** Critical elongation, critical stress, critical energy release rate, and fracture toughness of both NNP and AIREBO in the current study.

| | $\Delta a$ (nm) | $\Delta l$ (nm) | $\sigma_c$ (GPa) | $G_c$ (J m$^{-2}$) | $K_c$ (MPa m$^{1/2}$) |
|---|---|---|---|---|---|
| AC-edge (NNP) | 18 | 0.436 | 39.8 | 9.6 | 3.10 |
| ZZ-edge (NNP) | 18 | 0.479 | 43.8 | 11.7 | 3.41 |
| AC-edge (AIREBO) | 18 | 0.548 | 41.2 | 12.5 | 3.54 |
| ZZ-edge (AIREBO) | 18 | 0.474 | 41.2 | 10.9 | 3.29 |

**Table 3.** Elastic constants and Poisson's ratio ($\nu$) obtained from DFTB, reactive FFs, and trained neural network potential from the different data set selected by $\delta r$ with the same $\delta E = 1$ kcal mol$^{-1}$.

| | $E$ (GPa) | ($C_{11}$-$C_{12}$)/2 (GPa) | $\nu$ |
|---|---|---|---|
| DFT [62] | 1010 | | 0.203 |
| AIREBO | 828.8 | 304.1 | 0.362 |
| ReaxFF | 787.2 | 256.2 | 0.536 |
| DFTB (reference) | 1108.2 | 451.9 | 0.226 |
| $\delta r = 0.01$ Å | 1168.3 | 471.0 | 0.240 |
| $\delta r = 0.02$ Å | 1173.0 | 461.6 | 0.270 |
| $\delta r = 0.05$ Å | 1192.5 | 474.4 | 0.257 |
| $\delta r = 0.1$ Å | 1141.1 | 452.6 | 0.261 |
| $\delta r = 0.2$ Å | 1132.0 | 453.1 | 0.249 |

**Table 4.** Elastic constants obtained from neural network potential from the different data set selected by $\delta E$ with the same $\delta r = 0.1$ Å.

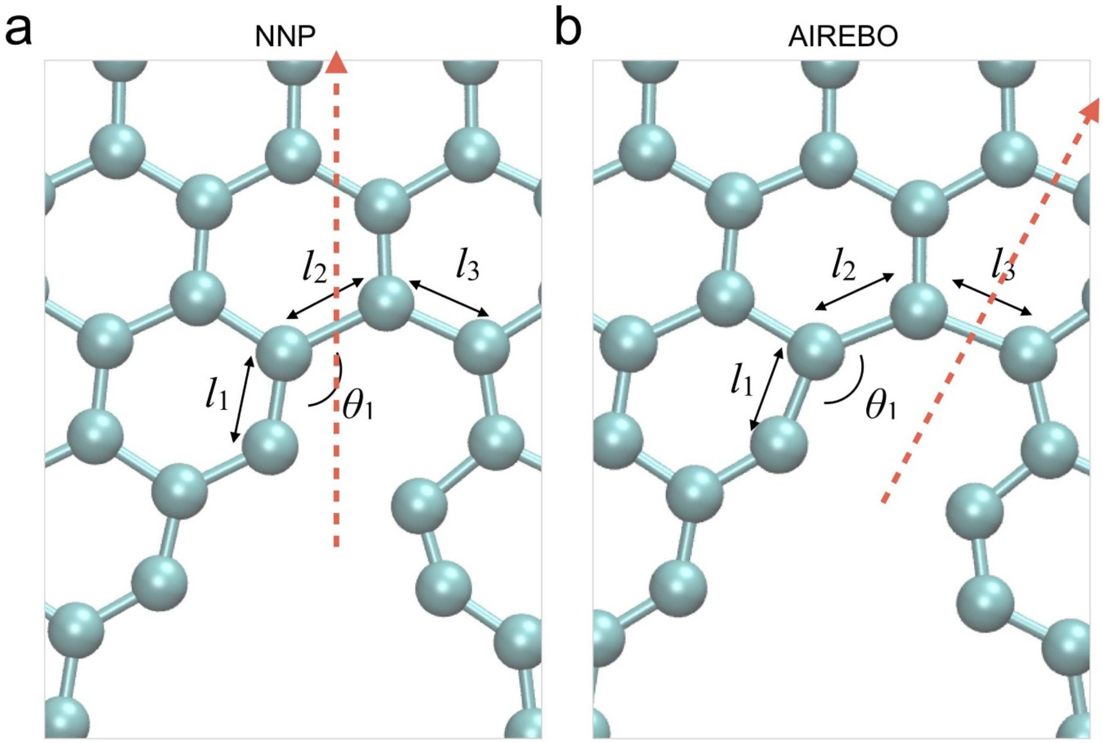
| | $E$ (GPa) | ($C_{11}$-$C_{12}$)/2 (GPa) | $\nu$ |
|---|---|---|---|
| $\delta E = 1$ kcal mol$^{-1}$ | 1141.1 | 452.6 | 0.261 |
| $\delta E = 2$ kcal mol$^{-1}$ | 1093.2 | 485.6 | 0.126 |
| $\delta E = 5$ kcal mol$^{-1}$ | 1112.6 | 445.9 | 0.248 |
| $\delta E = 10$ kcal mol$^{-1}$ | 1143.6 | 467.0 | 0.224 |

where $l_y$ (= 20 nm) is the length of the system in the $y$ direction, and $\Delta a$ (nm) is the total length of the crack propagation. Fracture toughness is then can be obtained by

$$K_c = \sqrt{G_c E}, \tag{5}$$

where $E$ is Young's modulus, and it is assumed as 1 TPa. The obtained values are listed in table 2. The obtained fracture toughness values from AIREBO for AC-edge (zz loading direction, 3.54 MPa m$^{1/2}$) and ZZ-edge (ac loading direction, 3.29 MPa m$^{1/2}$) agree with those from the previous studies with AIREBO [59]. The NNP predicts the fracture toughness along the AC-edge (zz loading direction, 3.10 MPa m$^{1/2}$) is lower than the ZZ-edge (ac loading direction, 3.41 MPa m$^{1/2}$), which shows an opposite trend to the results of the AIREBO. We note that our focus is on the difference in fracture toughness between ZZ and AC directions. The main factor of our estimations comes from the stored elastic energy before the crack propagation. Therefore, a correction of Young's modulus (calculated elastic constants are listed in table 3 and table 4) estimation without assuming 1 TPa does not affect the relative fracture toughness, e.g. ratio.

The frequency of torn edges in the suspended polycrystalline graphene monolayer depends on the fracture toughness of pristine graphene. The prediction from the NNP agrees well with those previous observations, while AIREBO predicts opposite behaviors in terms of the frequency of torn edge observation and torn AC edge configuration. The limitation of AIREBO comes from the softened angle stiffness under tensile loading, which also has been compared with DFT and DFTB calculations in the previous study [40]. Figure 7 shows graphene's bond lengths and angles at the crack tip just after the first bond breaking during the crack propagation. The angular deformation of NNP shows a lower angle (∼124°) than that of AIREBO (133°), which results in the elongation of the inner side bond length, $l_2$ (∼1.7 Å) than the outer side bond length, $l_3$ (∼1.6 Å). The relative bond lengths between $l_2$ and $l_3$ determine the crack path, and AIREBO prefers ZZ crack paths because $l_3$ (1.72 Å) is longer than $l_2$ (1.67 Å). In another 2D material, $WS_2$, these lattice distortions can result in anisotropic crack dynamics even with the same surface energy [35].



| | $l_1$ | $l_2$ | $l_3$ | $\theta_1$ |
|---|---|---|---|---|
| NNP | 1.41 Å | 1.71 Å | 1.58 Å | 124.1° |
| AIREBO | 1.47 Å | 1.67 Å | 1.72 Å | 133.0° |

**Figure 7.** Bond and angle at the crack tip after the first bond breaking along the AC edge direction from NNP (**a**) and AIREBO (**b**). Dotted red arrows represent the crack propagation. The angular deformation of trained NNP shows a lower angle than that of AIREBO, which results in the elongation of the inner side bond length, $l_2$, than the outer side bond length, $l_3$, from NNP. AIREBO shows propagation along the zigzag edge because $l_3$ is longer than $l_2$.

## 4. Conclusion

In summary, we proposed generating and selecting the training data for the deformation and fracture of crystals and applied it to 2D crystal graphene through DFTB calculations. We utilized the previously developed PyTorch library, TorchANI, to train the models of the Behler-Parrinello type's NNP. For MD simulation, we developed a PyTorch interface with LAMMPS, which can be expanded to other ML potential libraries through PyTorch. The proposed data reduction process improves the generalization of NNP training by mitigating overfitting. We show that the low RMSEs of energy and force do not automatically guarantee the reliable behaviors of the trained models[3]. The selected model considering physical properties can describe the torn edge configuration observed in the previous studies and explains well the higher frequency of torn AC edge occurrence, which is not possible with other reactive potentials. The proposed work frame can be applied to understand fracture dynamics of 2D and bulk crystals using NNPs.

We wish to emphasize that the current NNPs are limited as they should only be used for the simulation of stress-induced fracture and failure of pristine graphene. The training data set does not explicitly have failure dynamics of bilayer graphene, diamond, amorphous carbon network, carbon nanotube, graphyne, grain boundary, vacancies, folding, etc. Therefore, new data should be generated, or transfer learning/active learning is required to provide a quick path for developing NNPs of other applications. However, NNP is more useful for a large system in terms of computational speed than *first principles*-based electronic structure approaches. Also, the NNP is very flexible in capturing non-linear deformation-stress behaviors well (figure 5), which is challenging with the fixed functional form, such as harmonic equations in classical forcefields. In the previous study, we observed that it is challenging to match both deformation and non-linearity of stress–strain behaviors of graphene simultaneously with those from DFT even through

[3] While this manuscript was in review, we noticed a recent preprint with a similar finding [63].

reactive forcefields [37]. Crack dynamics is one of the exciting applications for NNPs due to its intrinsic multiscale feature. In this study, we only focus on the data generation and selection from the mixed loading and data reduction using DFTB as the reference method. However, the selected data can be utilized for high throughput calculation with more accurate methods. Also, active learning and transfer learning from the selected data would be interesting topics in the future.

## Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/gsjung0419/LammpsANI.

## Acknowledgments

The authors acknowledge helpful comments and discussion with J Warner. G S J and S I acknowledge support by the U. S. Department of Energy Fossile Energy and Carbon Management Program, Advanced Coal Processing Program, C4WARD project (FWP No. FEAA155). G S J acknowledges support for method developments by the Laboratory Directed Research and Development (LDRD), Eugene P Wigner Fellowship, Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the US Department of Energy. S I acknowledges partial support from the Artificial Intelligence Initiative as part of the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory. This research used resources of the Compute and Data Environment for Science (CADES) at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

## Author contribution statements

G S J conceived the idea, developed codes, and performed experiments. G S J and H M optimized codes. G S J performed DFTB calculations under supervision of S I. G S J, H M, and S I involved in writing and editing the manuscript.

## ORCID iDs

Gang Seob Jung https://orcid.org/0000-0002-8047-6505
Stephan Irle https://orcid.org/0000-0003-4995-4991

## References

[1] Buehler M J and Gao H 2006 Dynamical fracture instabilities due to local hyperelasticity at crack tips *Nature* **439** 307–10

[2] Kermode J R, Albaret T, Sherman D, Bernstein N, Gumbsch P, Payne M C, Csányi G and De Vita A 2008 Low-speed fracture instabilities in a brittle crystal *Nature* **455** 1224–7

[3] Livne A, Bouchbinder E, Svetlizky I and Fineberg J 2010 The near-tip fields of fast cracks *Science* **327** 1359

[4] Wang H, Yao Z, Jung G S, Song Q, Hempel M, Palacios T, Chen G, Buehler M J, Aspuru-Guzik A and Kong J 2021 Frank-van der Merwe growth in bilayer graphene *Matter* **4** 3339–53

[5] Jumper J *et al* 2021 Highly accurate protein structure prediction with AlphaFold *Nature* **596** 583–9

[6] Schmidt J, Marques M R G, Botti S and Marques M A L 2019 Recent advances and applications of machine learning in solid-state materials science *npj Comput. Mater.* **5** 83

[7] Butler K T, Davies D W, Cartwright H, Isayev O and Walsh A 2018 Machine learning for molecular and materials science *Nature* **559** 547–55

[8] Pfau D, Spencer J S, Matthews A G D G and Foulkes W M C 2020 *Ab initio* solution of the many-electron Schrödinger equation with deep neural networks *Phys. Rev. Res.* **2** 033429

[9] Vamathevan J *et al* 2019 Applications of machine learning in drug discovery and development *Nat. Rev. Drug Discov.* **18** 463–77

[10] Friederich P, Häse F, Proppe J and Aspuru-Guzik A 2021 Machine-learned potentials for next-generation matter simulations *Nat. Mater.* **20** 750–61

[11] Behler J and Csányi G 2021 Machine learning potentials for extended systems: a perspective *Eur. Phys. J.* B **94** 142

[12] Kocer E, Ko T W and Behler J 2022 Neural network potentials: a concise overview of methods *Annu. Rev. Phys. Chem.* **73** 163–86

[13] Pinheiro M, Ge F, Ferré N, Dral P O and Barbatti M 2021 Choosing the right molecular machine learning potential *Chem. Sci.* **12** 14396–413

[14] Lorenz S, Groß A and Scheffler M 2004 Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks *Chem. Phys. Lett.* **395** 210–5

[15] Behler J and Parrinello M 2007 Generalized neural-network representation of high-dimensional potential-energy surfaces *Phys. Rev. Lett.* **98** 146401

[16] Smith J S, Isayev O and Roitberg A E 2017 ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost *Chem. Sci.* **8** 3192–203

[17] Behler J 2015 Constructing high-dimensional neural network potentials: a tutorial review *Int. J. Quantum Chem.* **115** 1032–50

[18] Schütt K T, Kessel P, Gastegger M, Nicoli K A, Tkatchenko A and Müller K-R 2018 SchNetPack: a deep learning toolbox for atomistic systems *J. Chem. Theory Comput.* **15** 448–55

[19] Bartók A P, Payne M C, Kondor R and Csányi G 2010 Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons *Phys. Rev. Lett.* **104** 136403

[20] Thompson A P, Swiler L P, Trott C R, Foiles S M and Tucker G J 2015 Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials *J. Comput. Phys.* **285** 316–30

[21] Shapeev A V 2016 Moment tensor potentials: a class of systematically improvable interatomic potentials *Multiscale Model. Simul.* **14** 1153–73

[22] Chmiela S, Tkatchenko A, Sauceda H E, Poltavsky I, Schütt K T and Müller K-R 2017 Machine learning of accurate energy-conserving molecular force fields *Sci. Adv.* **3** e1603015

[23] Smith J S *et al* 2021 Automated discovery of a robust interatomic potential for aluminum *Nat. Commun.* **12** 1257

[24] Niu H, Bonati L, Piaggi P M and Parrinello M 2020 *Ab initio* phase diagram and nucleation of gallium *Nat. Commun.* **11** 2654

[25] Stuart S J, Tutein A B and Harrison J A 2000 A reactive potential for hydrocarbons with intermolecular interactions *J. Chem. Phys.* **112** 6472–86

[26] Brenner D W *et al* 2002 A second-generation reactive empirical bond order (REBO) potential energy expression for hydrocarbons *J. Phys.: Condens. Matter* **14** 783–802

[27] Zhang P *et al* 2014 Fracture toughness of graphene *Nat. Commun.* **5** 3782

[28] Shenderova O A, Brenner D W, Omeltchenko A, Su X and Yang L H 2000 Atomistic modeling of the fracture of polycrystalline diamond *Phys. Rev.* B **61** 3877–88

[29] Mattoni A, Ippolito M and Colombo L 2007 Atomistic modeling of brittleness in covalent materials *Phys. Rev.* B **76** 224103

[30] Perriot R, Gu X, Lin Y, Zhakhovsky V V and Oleynik I I 2013 Screened environment-dependent reactive empirical bond-order potential for atomistic simulations of carbon materials *Phys. Rev.* B **88** 064101

[31] Yeo J, Jung G S, Martín-Martínez F J, Beem J, Qin Z and Buehler M J 2019 Multiscale design of graphyne-based materials for high-performance separation membranes *Adv. Mater.* **31** 1805665

[32] Kim K, Coh S, Kisielowski C, Crommie M F, Louie S G, Cohen M L and Zettl A 2013 Atomically perfect torn graphene edges and their reversible reconstruction *Nat. Commun.* **4** 2723

[33] Wang S, Qin Z, Jung G S, Martin-Martinez F J, Zhang K, Buehler M J and Warner J H 2016 Atomically sharp crack tips in monolayer $MoS_2$ and their enhanced toughness by vacancy defects *ACS Nano* **10** 9831–9

[34] Jung G S, Wang S, Qin Z, Martin-Martinez F J, Warner J H and Buehler M J 2018 Interlocking friction governs the mechanical fracture of bilayer $MoS_2$ *ACS Nano* **12** 3600–8

[35] Jung G S, Wang S, Qin Z, Zhou S, Danaie M, Kirkland A I, Buehler M J and Warner J H 2019 Anisotropic fracture dynamics due to local lattice distortions *ACS Nano* **13** 5693–702

[36] Ryu G H, Jung G S, Park H, Chang R-J and Warner J H 2021 Atomistic mechanics of torn back folded edges of triangular voids in monolayer $WS_2$ *Small* **n/a** 2104238

[37] Jung G S, Irle S and Sumpter B G 2022 Dynamic aspects of graphene deformation and fracture from approximate density functional theory *Carbon* **190** 183–93

[38] Plimpton S 1995 Fast parallel algorithms for short-range molecular dynamics *J. Comput. Phys.* **117** 1–19

[39] Hourahine B *et al* 2020 DFTB+, a software package for efficient approximate density functional theory based atomistic simulations *J. Chem. Phys.* **152** 124101

[40] Jung G S, Stephan A I and Sumpter B Dynamic aspects of graphene deformation and fracture from approximate density functional theory *SSRN* (available at: https://ssrn.com/abstract=3933565)

[41] Gaus M, Goez A and Elstner M 2013 Parametrization and benchmark of DFTB3 for organic molecules *J. Chem. Theory Comput.* **9** 338–54

[42] Gao X, Ramezanghorbani F, Isayev O, Smith J S and Roitberg A E 2020 TorchANI: a free and open source PyTorch-based deep learning implementation of the ANI neural network potentials *J. Chem. Inf. Model.* **60** 3408–15

[43] Kingma D P and Ba J 2017 Adam: a method for stochastic optimization (arXiv:1412.6980)

[44] Loshchilov I and Hutter F 2019 Decoupled weight decay regularization (arXiv:1711.05101)

[45] Ruder S 2016 An overview of gradient descent optimization algorithms (arXiv:1609. 04747)

[46] He K, Zhang X, Ren S and Sun J 2015 Delving deep into rectifiers: surpassing human-level performance on ImageNet classification (arXiv:1502.01852)

[47] Paszke A *et al* 2017 Workshop on autodiff *NIPS 2017*

[48] Wang H, Zhang L, Han J and Weinan E 2018 DeePMD-kit: a deep learning package for many-body potential energy representation and molecular dynamics *Comput. Phys. Commun.* **228** 178–84

[49] Smith J S, Nebgen B, Lubbers N, Isayev O and Roitberg A E 2018 Less is more: sampling chemical space with active learning *J. Chem. Phys.* **148** 241733

[50] Botu V and Ramprasad R 2015 Adaptive machine learning framework to accelerate *ab initio* molecular dynamics *Int. J. Quantum Chem.* **115** 1074–83

[51] Smith J S, Nebgen B T, Zubatyuk R, Lubbers N, Devereux C, Barros K, Tretiak S, Isayev O and Roitberg A E 2019 Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning *Nat. Commun.* **10** 2903

[52] Devereux C, Smith J S, Huddleston K K, Barros K, Zubatyuk R, Isayev O and Roitberg A E 2020 Extending the applicability of the ANI deep learning molecular potential to sulfur and halogens *J. Chem. Theory Comput.* **16** 4192–202

[53] Hendrycks D and Gimpel K 2020 Gaussian error linear units (GELUs) (arXiv:1606.08415)

[54] Fernandez A *et al* 2018 *Learning from Imbalanced Data Sets* vol 10 (Cham: Springer) pp 978–3

[55] Kim K, Artyukhov V I, Regan W, Liu Y, Crommie M F, Yakobson B I and Zettl A 2012 Ripping graphene: preferred directions *Nano Lett.* **12** 293–7

[56] Jensen B D, Wise K E and Odegard G M 2015 Simulation of the elastic and ultimate tensile properties of diamond, graphene, carbon nanotubes, and amorphous carbon using a revised ReaxFF parametrization *J. Phys. Chem.* A **119** 9710–21

[57] Dewapriya M A N and Rajapakse R K N D 2014 Molecular dynamics simulations and continuum modeling of temperature and strain rate dependent fracture strength of graphene with vacancy defects *J. Appl. Mech.* **81** 081010

[58] Jung G, Qin Z and Buehler M J 2015 Molecular mechanics of polycrystalline graphene with enhanced fracture toughness *Extrem. Mech. Lett.* **2** 52–59

[59] Zhang T, Li X and Gao H 2015 Fracture of graphene: a review *Int. J. Fract.* **196** 1–31

[60] Girit Ç Ö *et al* 2009 Graphene at the edge: stability and dynamics *Science* **323** 1705–8

[61] Govind Rajan A, Silmore K S, Swett J, Robertson A W, Warner J H, Blankschtein D and Strano M S 2019 Addressing the isomer cataloguing problem for nanopores in two-dimensional materials *Nat. Mater.* **18** 129–35

[62] Lebedeva I V, Minkin A S, Popov A M and Knizhnik A A 2019 Elastic constants of graphene: comparison of empirical potentials and DFT calculations *Physica* E **108** 326–38

[63] Fu X *et al* 2022 Forces are not enough: benchmark and critical evaluation for machine learning force fields with molecular simulations (arXiv:2210.07237)