



## Full Length Article

## Prediction of potential energy profiles of molecular dynamic simulation by graph convolutional networks



Kota Noda, Yasushi Shibuta\*

Department of Materials Engineering, The University of Tokyo 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

## ARTICLE INFO

## Keywords:

Molecular dynamics  
Machine learning  
Graph convolutional networks  
Graph representation  
Solid-liquid coexisting system

## ABSTRACT

A graph convolutional networks (GCN)-based machine learning (ML) model is constructed to predict physical properties of metallic materials from graph representation of atomic configuration of molecular dynamics (MD) simulation. The developed ML model is employed for the prediction of time variation of the potential energy of a solid–liquid biphasic system of nickel. The learned ML model gives a good prediction on the property of training data. Moreover, it is confirmed that the ML model has generalization performance sufficient to make adequate predictions on unknown graph structures despite the lack of information on interatomic distances in the graph representation. It is significant in this study to show that the graph representation can be a good notation for the prediction of various properties from MD simulations since there is no established notation for atomic configuration of MD simulation especially for large-scale system of metallic materials.

## 1. Introduction

Molecular dynamics (MD) simulation is a powerful tool for investigating kinetic aspect of materials processes since it can trace the motion of atoms in target materials directly. With the benefit of recent dramatic progress in the environment of high-performance computing, MD simulations can now handle large systems with more than billion atoms [1,2]. Moreover, machine learning (ML) methods have rapidly become popular in combination with the high-performance computing and are being used in a variety of fields. Therefore, it is in a natural step for ML approaches to be utilized in current problems of MD simulations [3]. The most popular and successful implementation of the ML approach in the MD simulation is the development of accurate interatomic potentials called the ML potential or neural network potential (NNP) [4–6]. The NNP represents the multidimensional potential energy surface of target system accurately by learning the reference data obtained from high-level electronic structure calculations [5]. Up to now, a large number of NNPs have been proposed [5] and a recently proposed NNP can treat any combination of 45 elements [6].

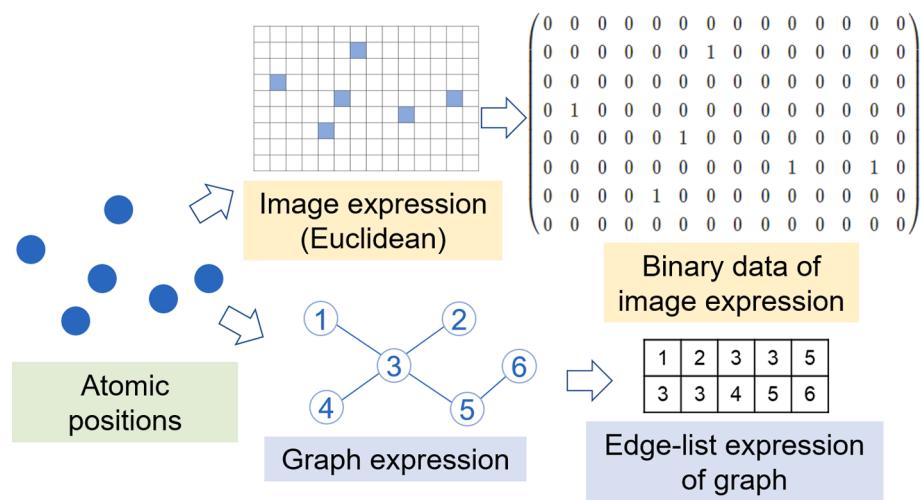
In addition to NNPs, ML approaches are also used to analyze results of MD simulations. In general, the primary information obtained from MD simulations is only positions and velocities of atoms (i.e., trajectories), and it is essential to derive the necessary physical properties from the primary information of MD simulations by the post-processing

[7]. Therefore, MD trajectories are often employed as input data of the ML analysis [3,8] to classify atomic structures and to predict unknown physical properties. For example, local atomic configuration in metallic materials was analyzed using the convolutional neural network (CNN) [9], which classified local atomic structures more accurately than conventional methods [10,11]. ML approaches contributed to the classification of organic molecules [12], biomolecules [13,14] and catalytic materials [15] with specific functionality. Moreover, mechanical [16–18] and thermal properties [19,20] of unknown materials were predicted by MD trajectories in combination with ML approaches. Thus, ML-based studies using MD trajectories as input data have been actively performed and have steadily achieved plenty of new knowledge.

It is common to perform some preprocessing of MD trajectories for the use in ML methods rather than using the coordinate data as it is from a practical viewpoint. For example, in the case of molecule systems, simplified molecular input line entry system (SMILES) [21] is one of the common chemical notations which represents a chemical structure uniquely. SMILES is commonly used in the field of cheminformatics. On the other hand, there is no such established notation for atomic configuration of metallic systems except for perfect symmetric crystals. Therefore, atomic coordinates are often converted into grid point information [10] to apply the CNN-based model since CNN is commonly applied to analyze visual imagery consisting of grid point information in the Euclidean space [9]. However, the grid point expression of atomic

\* Corresponding author.

E-mail address: [shibuta@material.t.u-tokyo.ac.jp](mailto:shibuta@material.t.u-tokyo.ac.jp) (Y. Shibuta).



**Fig. 1.** Schematic image of graph and image expressions of atomic positions.

coordinates is not efficient since it has a lot of sparse area (see the schematic in Fig. 1). Therefore, it is not an efficient way to expand the CNN-based model with grid point expression to the large system of MD simulations. Recently, new types of neural networks that handle graph structures are rapidly gaining popularity, which is called graph convolutional network (GCN) [22]. The graph structure consists of nodes and edges, where the node is a unit representing an entity or object, and edge is a unit representing the relationships or interactions between nodes. Originally, atomic configurations have a high affinity for graph structures since atomic configuration can be represented by atomic positions and interatomic bonds. Therefore, an edge-list representation of the connections between atoms is very efficient and reduces the data size as shown in Fig. 1. On the other hand, there exists one possible drawback to the graph structure, that is, the loss of distance information between nodes. Therefore, the prediction of physical properties from the atomic structure of large-scale MD simulation using GCN is still an unexplored area. To this end, the objective of this study is to construct a GCN-based ML model to predict physical properties of the target MD system. In this study, time variation of the potential energy in the biphasic system is employed as an example of the physical property to be predicted.

## 2. Methodology for data creation and machine learning model

### 2.1. Outline of methodology

In advance of the examination of ML models, we first perform MD simulations of solidification and melting of a solid–liquid coexisting system of nickel to create data for ML models. We extract the time variation of the potential energy of the system as label data of the training step. Then, we construct the graph representation of the system from atomic configuration. In the graph representation, atomic position, velocity, edges per node and attributes of each atom as solid or liquid are assigned for each node as features. After that, we train a GCN-based ML model using node features and graph structures as input data and the potential energy obtained in the MD simulation is employed as label data. Details of each process are described below.

### 2.2. Molecular dynamic simulation

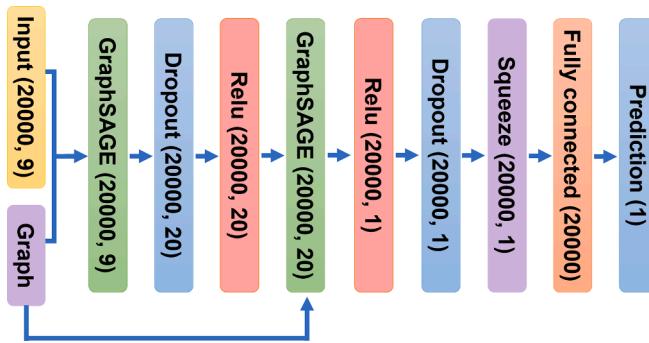
MD simulations are performed to create data for the following ML process. A solid–liquid coexistence system of nickel is prepared by connecting face-centered cubic (fcc) and liquid nickel systems with dimension  $35.2 \times 35.2 \times 88.0 \text{ \AA}^3$  containing 10,000 atoms in each phase. Both phases are connected so that the fcc(100) surface is in

contact with the liquid phase. The solid and liquid phases each consist of 10,000 atoms, and therefore 20,000 atoms exist in total in the solid–liquid coexisting system. The system is then isothermally annealed for 90000 fs with the isobaric-isothermal (i.e., number of atoms, pressure and temperature (NPT)-constant) ensemble using the Nose-Hoover thermostat and barostat [23,24]. Target temperatures are set from 1000 K to 1900 K at 100 K intervals. The embedded atom method (EAM) potential fitted by Purja Pun and Mishin [25] is employed for the interaction between nickel atoms. The classical equation of motion is integrated using the velocity–Verlet method with a time step of 1.0 fs. The large-scale atomic/molecular massively parallel simulator (LAMMPS) [26] is employed for performing MD simulation. The open visualization tool (OVITO) [27] is employed for visualizing MD results and performing the common neighbor analysis (CNA) [28] to identify the local atomic configuration.

### 2.3. Conversion from atomic coordinates into graph structure

To use GCN, it is essential to convert atomic coordinates of the system into the graph structure. There is originally a similarity between interatomic bonds and edge structures as described above. Therefore, each atom is regarded as each node in a one-to-one relationship. That is, 20,000 nodes exist in a graph representation of one snapshot of the MD configuration. The edge is created between two atoms when the interatomic distance is shorter than the threshold distance. The threshold distance is set to be 2.7 Å, which is between the first and second nearest neighbor interatomic distance of fcc-nickel, since edges usually connect adjacent nodes in a graph structure. The features consist of 9 dimensions: three components of position ( $x, y, z$ ) and velocity ( $v_x, v_y, v_z$ ), the absolute value of the velocity, the number of edges connected to the node, and the analytical value of CNA (solid or liquid). These features are assigned to each node. In the MD simulation, the interatomic potential represents a conserved force-field, and its derivative represents the conserved force, which is directly related to the motion of atoms. Moreover, the square of the velocity is proportional to the kinetic energy. Since a close relationship exists between kinetic and potential energies, the information of the velocity is added in features for the prediction of the potential energy even though the kinetic energy is not predicted directly.

The array size of the edge-list representation for the initial configuration of the MD simulation with 20,000 atoms is (2, 99799), which is approximately 1.5 MB in the npy format (a standard binary file format in NumPy). On the other hand, that of the image expression for CNN with 0.1 Å spacing as in our previous study [10] is (352, 352, 1760), which is approximately 1.62 GB in the npy format. That is, data size of the image



**Fig. 2.** Flowchart of the graph convolutional network-based machine learning model performed in this study.

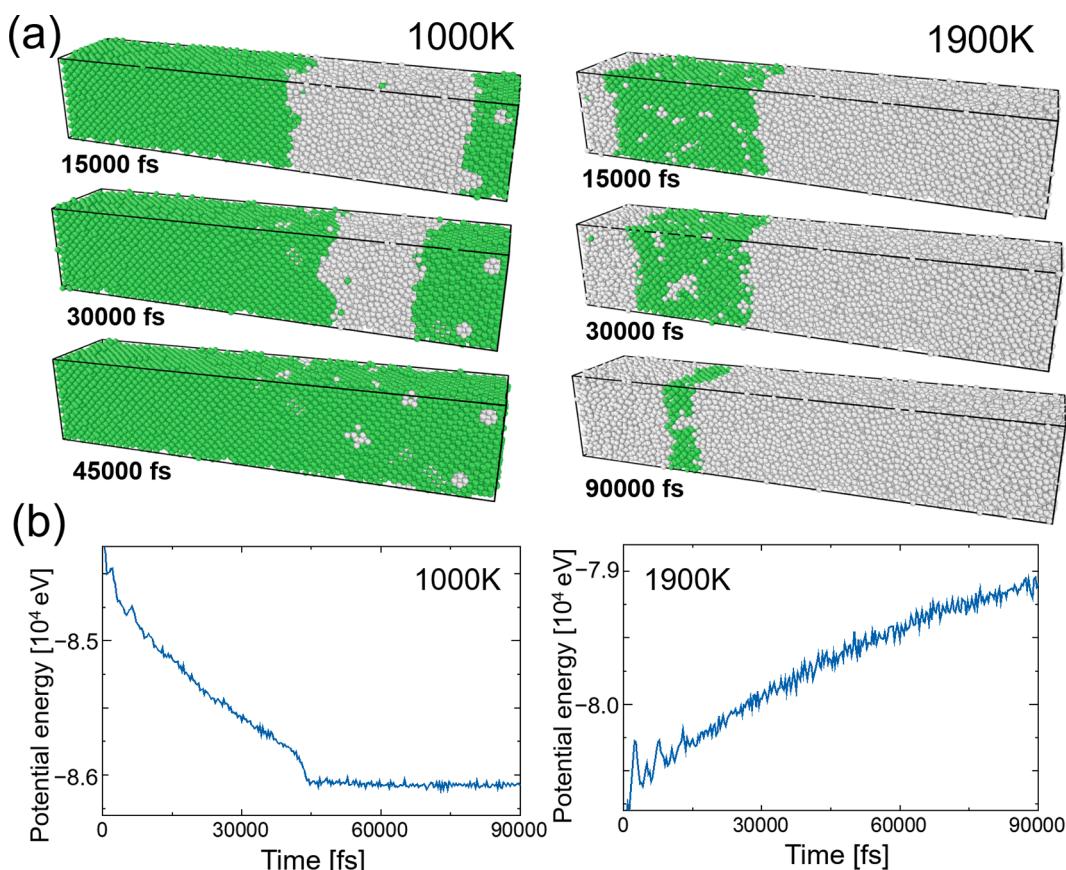
expression for CNN is 1000 times larger than that of the edge-list representation for the same atomic coordinates. Therefore, the edge-list representation is quite efficient for representing atomic configuration of large-scale MD simulations. Moreover, it is well known that the computational cost increases as the power of the number of atoms (i.e.,  $O(N^2)$ ,  $N$  is the number of atoms) when counting all pair of atoms in the system. We reduce the computational cost by dividing the space of calculation system into subdomains, and calculating the distance between pairs of atoms belong to same and neighboring subdomain only. This method is known as the domain decomposition method (DDM) for accelerating MD simulation [29]. In this study, the calculation system is equally divided into  $10 \times 10 \times 10$  subdomains. We considered all atoms belong to 6 subdomains in proximity in the vertical and horizontal directions although 26 neighboring meshes including in diagonal direction are often considered in the conventional DDM. We have confirmed

in the preliminary calculation that this difference does not affect the ML results significantly.

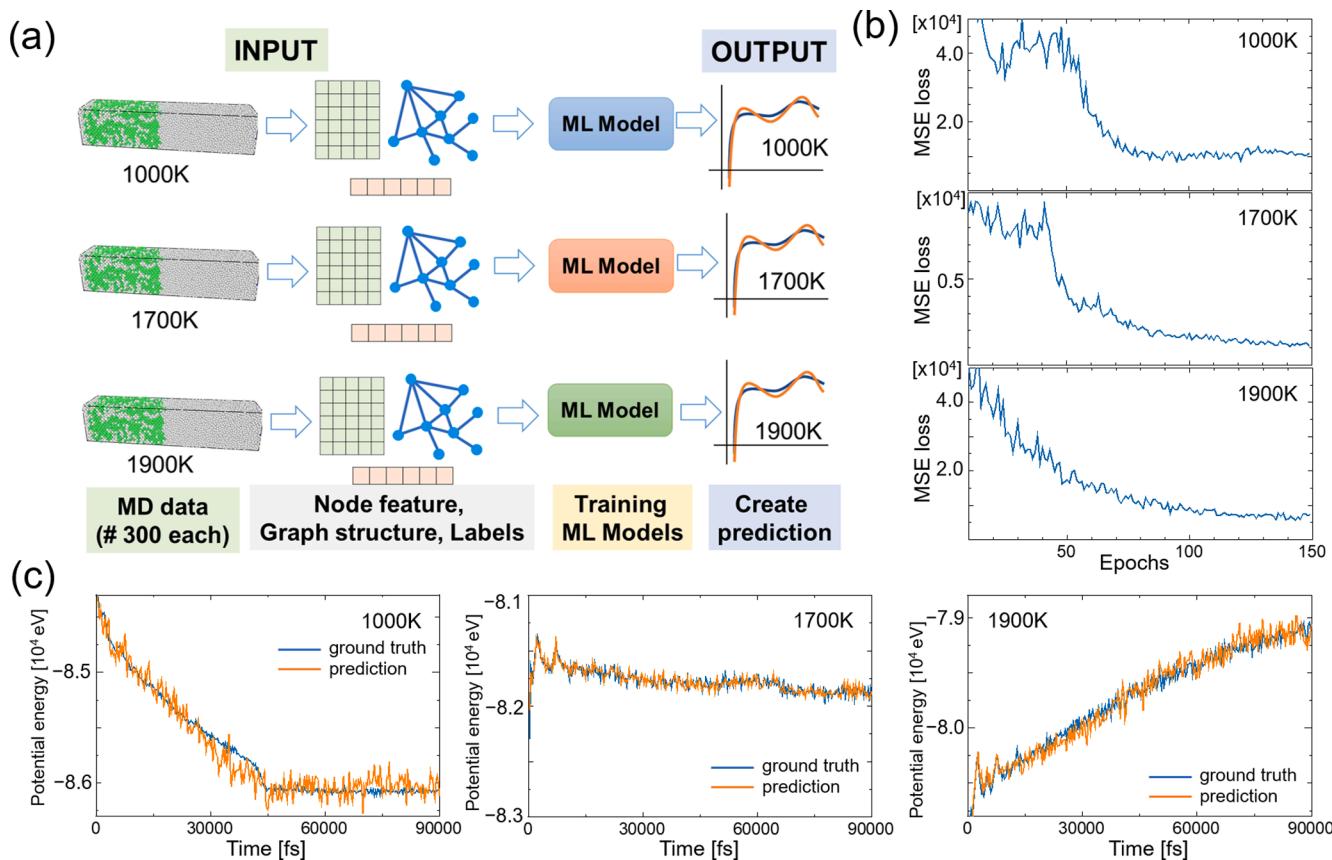
#### 2.4. Graph convolution network-based machine learning model

**Fig. 2** shows a flowchart of the GCN-based ML model performed in this study. GraphSAGE [30] is employed to convolute features of the graph structure, which is an extension of conventional GCN and can handle inductive problem setting. GraphSAGE is used two times to convolute features and simple fully connected layer is used in the last step of the model. We employ the mean squared error (MSE) as a loss function, the Cosine Annealing [31] as a scheduler function, and the adaptive moment estimation (ADAM) [32] as an optimizer. These procedures are implemented by PyTorch [33] and PyTorch Geometric [34].

We perform four examinations to validate the performance of the ML model. First, we investigate whether the ML model gives a good prediction of training data. In the first examination, separate ML models are applied to each training data of different temperatures and trained separately to predict the potential energy of each temperature independently. In the second examination, one common ML model is employed for the prediction of potential energies of three different temperatures. After that, we investigate the generalization performance of the ML model, which is the ability to predict the properties of unseen data. In the third examination, all data from MD simulations are divided into training and validation dataset. One common ML model is trained with training dataset and evaluated the MSE loss by using validation dataset. Finally, the extraction interval of data on the prediction performance is evaluated. Schematics of the examinations are shown later in each session.



**Fig. 3.** (a) Snapshots and (b) time evolution of the potential energy for solid-liquid coexisting system during the annealing at 1000 and 1900 K. Green and white atoms in the snapshots represent solid and liquid atoms, respectively.



**Fig. 4.** (a) Schematic image of training step of the first examination. (b) Variation of mean squared error (MSE) as a function of the number of epochs. (c) Time variation of predicted potential energy at the epoch of the best MSE loss (orange line) and the potential energy from molecular dynamics simulation as a value of ground truth (blue line).

### 3. Results and discussion

#### 3.1. Molecular dynamic simulation for data creation

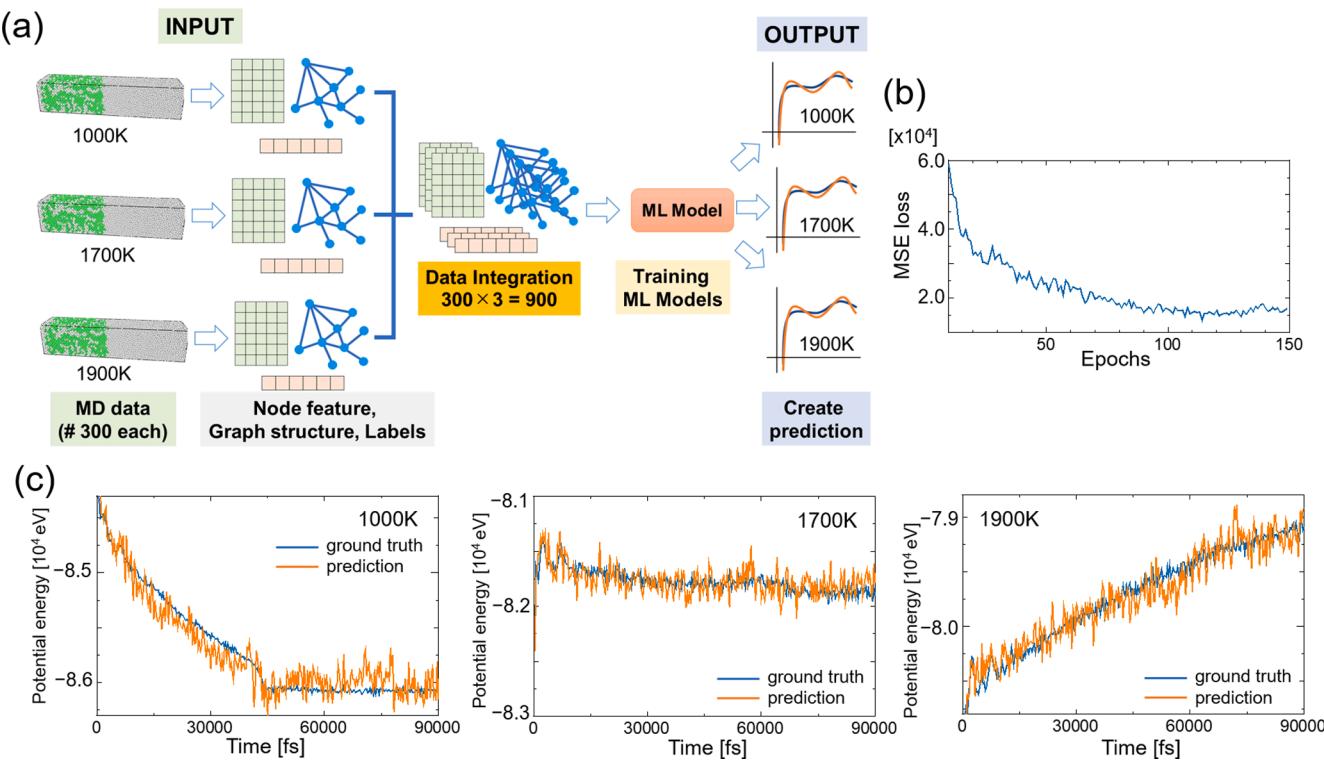
Fig. 3 shows snapshots and time evolution of the potential energy for solid–liquid coexisting system during the annealing at 1000 and 1900 K. CNA is used to define the attribute of each atom (i.e., solid or liquid). In the case of 1000 K, the solid–liquid interface moved toward the liquid phase due to solidification. The potential energy of the system decreases as the area of solid phase increases. There was no change in the potential energy once the solidification completes after approximately 40000 fs. On the other hand, the potential energy increases as the melting proceeds at 1900 K and some of solid phase remains after 90000 fs calculation. Since the melting point of nickel calculated by this EAM potential was estimated to be approximately 1700 K [35,36], solidification or melting occurs below or above the melting point, respectively. Atomic position, velocity and attribute (solid or liquid) of all atoms for every 300 fs of 90000 fs MD simulations are collected as input data of the ML model upon the conversion into the graph structure. That is, the number of data is 300 for each temperature. Moreover, the effect of extraction interval of data is examined using data extracted every 600 and 900 fs in Section 3.5 where the number of data is 150 (600 fs interval) and 100 (900 fs interval) for each temperature, respectively. The potential energy at the corresponding time is used as the label data for the ML model. We employ the relative value of the potential energy with respect to the value of the initial configuration when calculating the MSE loss to increase the signal-to-noise ratio of the MSE loss. Similar calculations are performed for other temperature conditions to prepare data for the ML model.

#### 3.2. Verification of ML model for prediction of training data

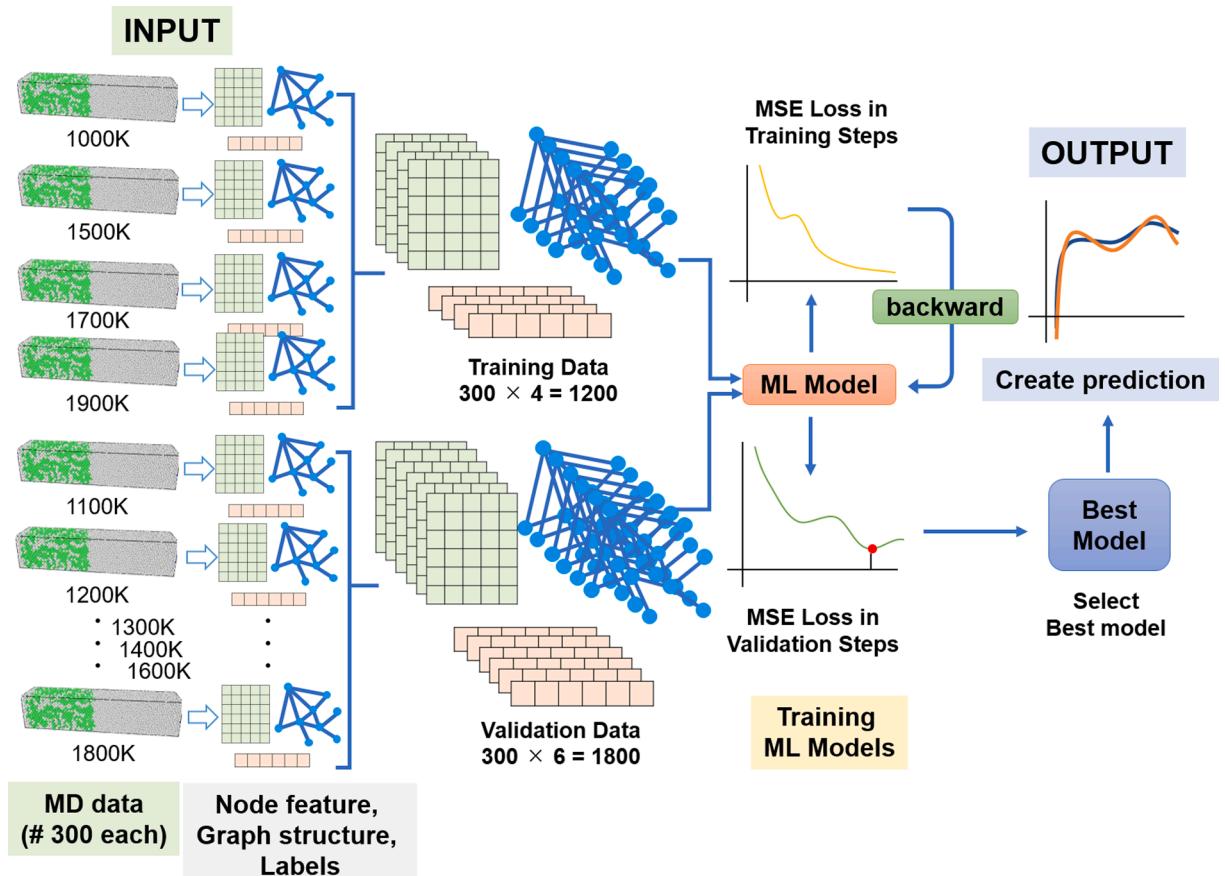
First, we verify that the developed ML model gives good predictions on training data. The data from MD simulations of the annealing at 1000, 1700 and 1900 K are employed as training data of three independent ML models as illustrated in the schematic in Fig. 4(a). The number of epochs is 150 and the learning rate is set to  $1.0 \times 10^{-3}$ . Fig. 4(b) shows the variation of MSE as a function of the number of epochs. It is confirmed that the MSE drops sharply in the very beginning and then converges slowly for all temperatures although there are some fluctuations. That indicates that the features of the training data are being learned correctly. Fig. 4(c) shows time variation of predicted potential energy at the epoch of the best MSE loss and that of ground truth (i.e., the MD result). Best MSE loss are 8522.6 at Epoch 95 for 1000 K, 1066.8 at Epoch 141 for 1700 K, and 5651.4 at Epoch 143 for 1900 K, respectively. The models for all temperatures succeed in tracking the trend of the label data. In the case of 1000 K, solidification progresses up to 45000 fs where the potential energy keeps decreasing and then the potential energy has plateaued once solidification completes. Interestingly, the ML model interprets this trend correctly and it outputs appropriate predicted value. Therefore, it was verified that our ML model gives correct predictions on training data.

#### 3.3. Prediction of potential energies at different temperatures with a common ML model

Next, we investigated whether a one ML model trained by several training data can distinguish data from different conditions correctly. That is, one common ML model is employed for the prediction of potential energies of three different temperatures uniformly as illustrated

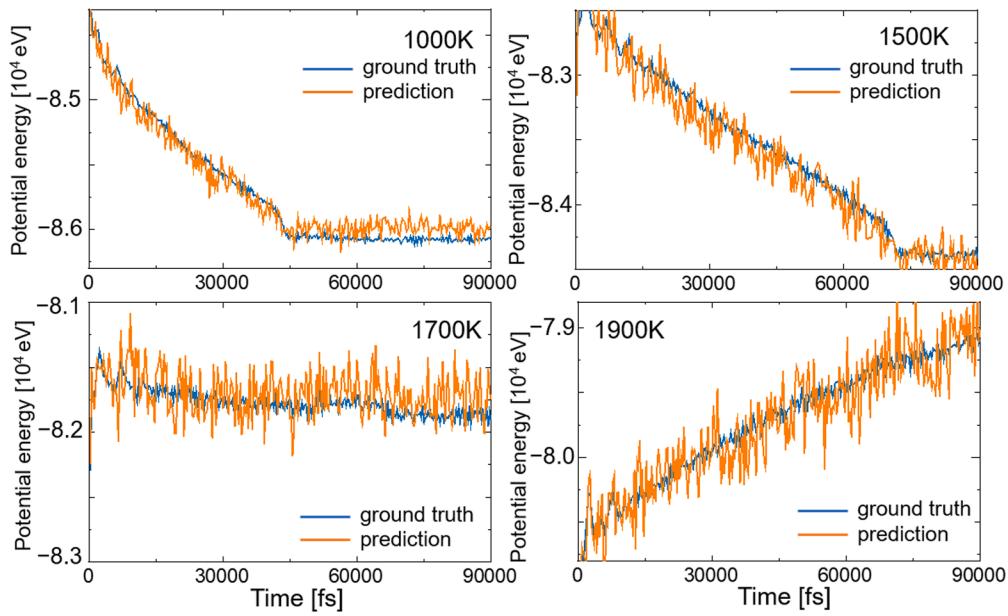


**Fig. 5.** (a) Schematic image of training step of the second examination. (b) Variation of mean squared error (MSE) as a function of the number of epochs. (c) Time variation of predicted potential energy at the epoch of the best MSE loss (orange line) and the potential energy from molecular dynamics simulation as a value of ground truth (blue line).

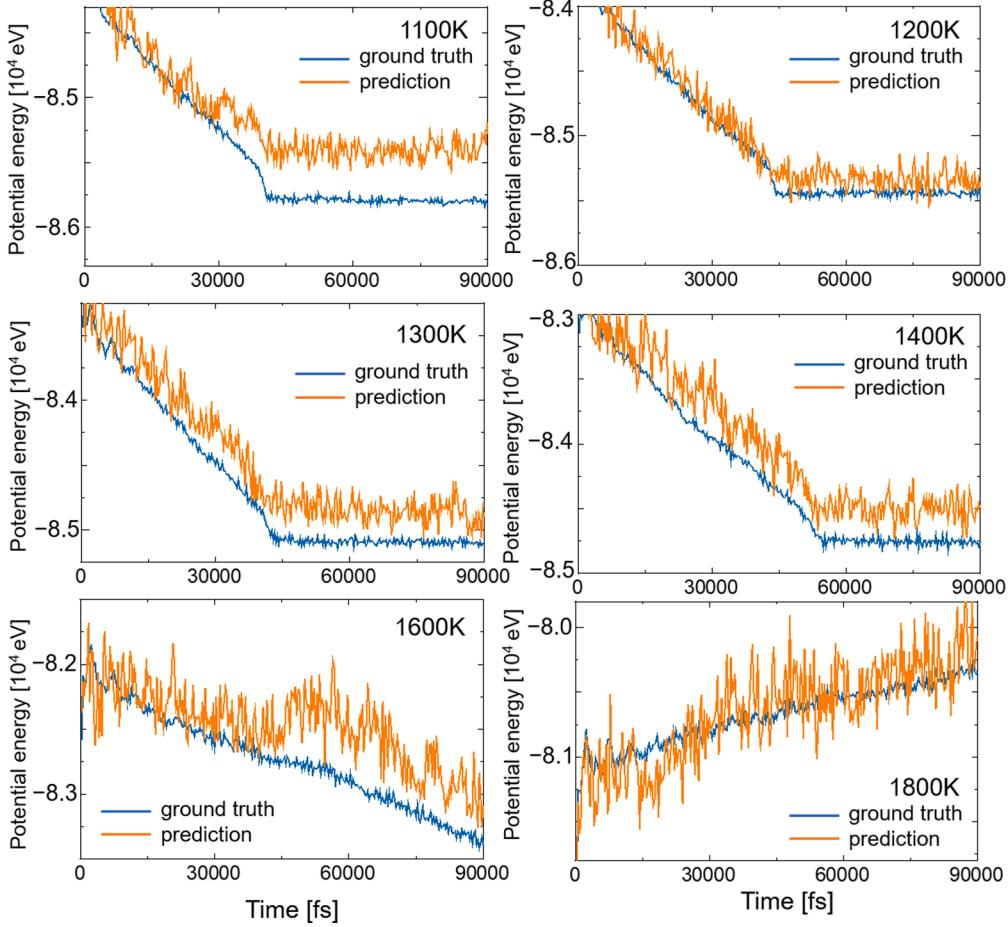


**Fig. 6.** Schematic image of training step of the third examination.

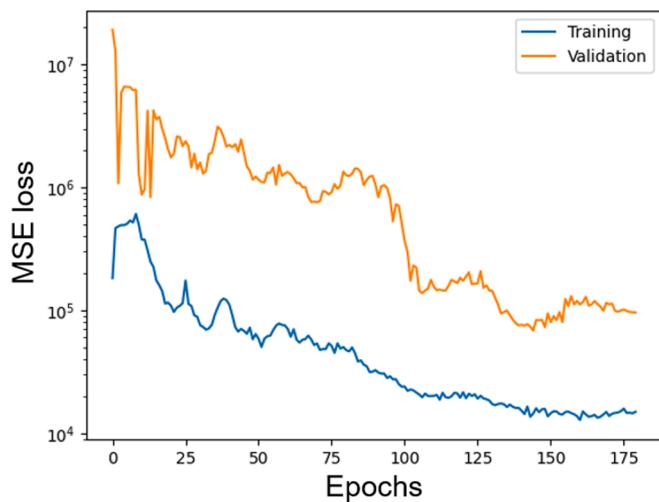
## (a) Training data



## (b) Validation data



**Fig. 7.** Time variation of predicted potential energy at the epoch of the best MSE loss (orange line) and the potential energy from molecular dynamics simulation as a value of ground truth (blue line) for temperature conditions included in the learning step ((a) training data) and for temperature conditions not used in the learning step ((b) validation data).



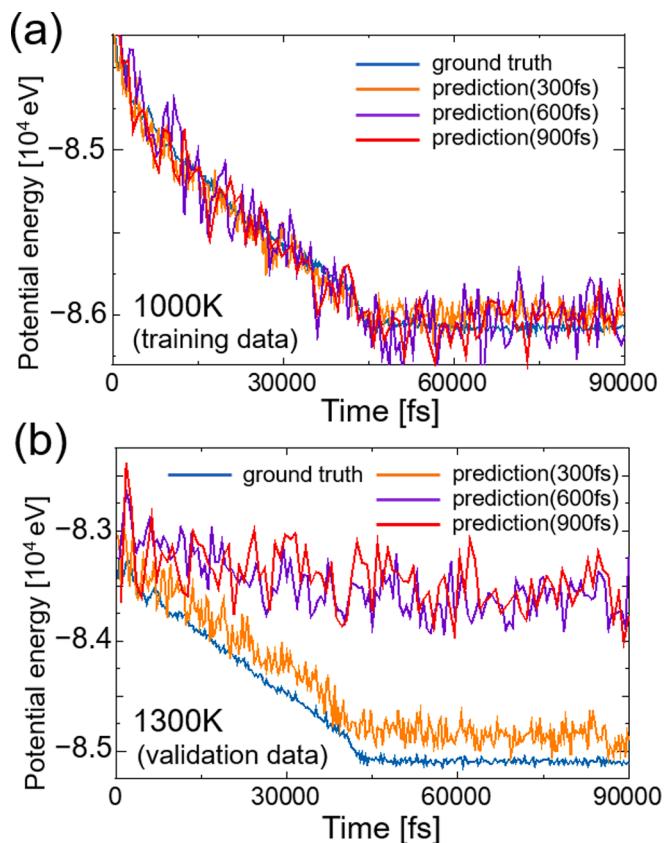
**Fig. 8.** Variation of mean squared error (MSE) loss as a function of the number of epochs for training and validation steps of Fig. 6.

in the schematic in Fig. 5(a). If one common ML model can capture characteristics of multiple temperatures from input data of the graph structure, appropriate prediction would be gained for each condition uniformly. Number of epochs is 150 and learning rate is set to  $1.0 \times 10^{-3}$ . Fig. 5(b) the MSE as a function of the number of epochs. It is again confirmed that the MSE drops sharply in the very beginning and then converges slowly. That indicates that the features of the training data are being learned correctly even if multiple input data are trained in one common ML model. Fig. 5(c) shows time variation of predicted potential energy at the epoch of the best MSE loss (MSE: 13589.0 at Epoch 114) and that of ground truth (i.e., the MD result). It is confirmed that one common ML model can predict the trend in the time variation of the potential energy for all temperatures at once although the fluctuation in the value seems larger than the case of independent ML model above. Especially, it is encouraging that the bending characteristic of the graph in 1000 K is properly predicted.

#### 3.4. Generalization performance for prediction on properties of unknown structures

In the previous section, it is confirmed that one common ML model can reproduce physical properties at multiple temperatures upon the training with the graph structure at corresponding temperatures. However, it is not clear whether the model has generalization performance sufficient to make adequate predictions on unseen graph structures. Therefore, we investigate whether one common ML model can predict the potential energy of the conditions that are not used in the learning step. Here, all data from MD simulations are divided into two groups, that is, training and validation dataset. Training dataset includes the data of 1000 K, 1500 K, 1700 K and 1900 K, and validation dataset includes the data of other temperatures. We trained one common ML model with training dataset and evaluated the MSE loss by using validation dataset. Note that validation dataset is never used for training purpose. Number of epochs is 180 and learning rate is set to  $1.0 \times 10^{-3}$ . The schematic image of this procedure is shown in Fig. 6.

Fig. 7 shows time variation of predicted potential energy at the epoch of the best MSE loss (training loss: 14461.4 and validation loss 68410.0 at 144 epoch). Note that we stopped the learning step around a minimum value of the MSE loss of the validation data before it started increasing again, which technique is known as the early stopping to avoid overlearning. At a first glance, the ML model can capture the general trend of time variation of potential energy for all temperatures including the cases without training dataset. The prediction without training dataset tends to be worse than those with training dataset



**Fig. 9.** Time variation of the potential energy from molecular dynamics simulation as a value of ground truth (blue line) and predicted potential energy at the epoch of the best MSE loss for various extraction intervals of data: 300 fs (orange line), 600 fs (purple line) and 900 fs (red line) intervals. (a) 1000 K as a representative temperature of training data and (b) 1300 K as a representative temperature of the validation data.

quantitatively for cases of 1100 K, 1300 K, 1400 K and 1600 K. However, this is a common feature of the prediction without training data set since the MSE of validation data is higher than that of training data as shown in Fig. 8. On the other hand, it is noteworthy that the model correctly predicts bending characteristics of the graphs at 1100 K, 1200 K, 1300 K and 1400 K although there are some quantitative deviations. Therefore, it is considered that the model employed in this study has a generalization performance to capture the feature of physical properties to some extent although the ML model could not give perfect quantitative prediction. It is a future task to improve the generalization performance.

#### 3.5. Effect of extraction interval of data on the prediction performance

Finally, we evaluate the effect of the extraction interval of data on the prediction performance by means of the third examination. In addition to the original data of 300 fs intervals, additional examinations are performed with data of 600 fs intervals and that of 900 fs intervals. The calculation procedure is the same as in the original examination except for the data interval. Fig. 9 shows the time variation of the predicted potential energy at the epoch of the best MSE loss for 1000 K as the representative temperature of the training data and for 1300 K as that of the validation data. For the case of 1000 K, the trend of ground truth is accurately reproduced even when the extraction interval is increased. On the other hand, in the case of 1300 K, the ML model fails to reproduce bending characteristics of the graph when the extraction interval is increased at 600 fs and 900 fs intervals. Therefore, it was confirmed that the extraction interval of data employed in this study (i.e., 300 fs interval) is appropriate to reproduce characteristics of the

original data.

#### 4. Conclusion

In this study, a GCN-based ML model is constructed to predict physical properties from trajectory of MD simulations. The atomic coordinates from the results of MD simulation is converted into the graph structure where atomic position, velocity, edges per node and attributes of each atom as solid or liquid are assigned for each node as features. Four examinations are performed to validate the performance of the ML model. First, the developed ML model is confirmed to give good predictions on training data by adapting each training data to an independent ML model. Next, the extensibility of the ML model is confirmed by employing one common ML model for the prediction of potential energies of three different temperatures uniformly. Then, generalization performance for the prediction on properties of unknown structures is investigated. The model correctly predicts the time variation of potential energy including the characteristics of the graph to bend in the middle of time. Finally, the extraction interval of data on the prediction performance is evaluated. When the interval of data extraction becomes larger, characteristics of the validation data are not well predicted whereas that of the training data can be reproduced well.

It is significant in this study to show that the graph representation can be a good notation for the prediction of physical properties from MD simulations despite the lack of information on interatomic distances in the graph representation. That is, the loss of information of interatomic distances is not significant for predicting properties of metallic materials from MD simulations by the GCN-based ML model. This finding encourages us since there is no established notation for atomic configuration of the large-scale MD simulation especially for metallic materials. In this study, time variation of the potential energy is employed as a typical example of a label data for the validation of the ML model. In the next step, prediction of various physical properties will be examined in parallel with the refinement of the model.

#### CRediT authorship contribution statement

**Kota Noda:** Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Yasushi Shibuta:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

#### Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Data availability

Data will be made available on the reasonable request. The codes developed in this study are available on GitHub: <https://github.com/nodematerial/MD-GNN>.

#### Acknowledgement

This work was partially supported by Grant-in-Aid for Scientific Research (B) (No. 22H01754) from Japan Society for the Promotion of Science (JSPS).

#### References

- [1] Y. Shibuta, S. Sakane, E. Miyoshi, S. Okita, T. Takaki, M. Ohno, Heterogeneity in homogeneous nucleation from billion-atom molecular dynamics simulation of solidification of pure metal, *Nature Comm.* 8 (2017) 10.
- [2] Y. Shibuta, M. Ohno, T. Takaki, Advent of cross-scale modeling: High-performance computing of solidification and grain growth, *Adv. Theory Simul.* 1 (2018) 1800065.
- [3] F. Musil, A. Grisafi, A.P. Bartók, C. Ortner, G. Csányi, M. Ceriotti, Physics-inspired structural representations for molecules and materials, *Chem. Rev.* 121 (2021) 9759.
- [4] J. Behler, M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces, *Phys. Rev. Lett.* 98 (2007) 146401.
- [5] E. Kocer, T.W. Ko, J. Behler, Neural network potentials: A concise overview of methods, *Annu. Rev. Phys. Chem.* 73 (2022) 163.
- [6] S. Takamoto, C. Shinagawa, D. Motoki, K. Nakago, W. Li, I. Kurata, T. Watanabe, Y. Yamaya, H. Iriuchi, Y. Asano, T. Onodera, T. Ishii, T. Kudo, H. Ono, R. Sawada, R. Ishitani, M. Ong, T. Yamaguchi, T. Kataoka, A. Hayashi, N. Charoenphakdee, T. Ibuka, Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements, *Nat Comm.* 13 (2022) 2991.
- [7] Y. Shibuta, Estimation of thermodynamic and interfacial parameters of metallic materials by molecular dynamics simulations, *Mater. Trans.* 60 (2019) 180.
- [8] S. Kaptan, I. Vattulainen, Machine learning in the analysis of biomolecular simulations, *Adv. Phys.: X* 7 (2022) 2006080.
- [9] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE* 86 (1998) 2278.
- [10] T. Fukuya, Y. Shibuta, Machine learning approach to automated analysis of atomic configuration of molecular dynamics simulation, *Comp. Mater. Sci.* 184 (2020) 109880.
- [11] T.-S. Liang, P.-P. Shi, S.-Q. Su, Z. Zeng, Learning physical states of bulk crystalline materials from atomic trajectories in molecular dynamics simulation, *Chin. Phys. B* 31 (2022) 126402.
- [12] C. Ramírez-Palacios, S.J. Marrink, Computational prediction of  $\alpha$ -transaminase selectivity by deep learning analysis of molecular dynamics trajectories, *QRB Discovery* 4 (2023) e1.
- [13] A. Plante, D.M. Shore, G. Morra, G. Khelashvili, H. Weinstein, A machine learning approach for the discovery of ligand-specific functional mechanisms of GPCRs, *Molecules* 24 (2019) 2097.
- [14] D. Geisel, P. Lenz, Machine learning classification of trajectories from molecular dynamics simulations of chromosome segregation, *PLoS One* 17 (2022) e0262177.
- [15] M. Tsunawaki, S. Fukuhara, Y. Shibuta, Hierarchical clustering of structural and electronic characteristics obtained from molecular dynamics simulation of catalytic reaction on metal nanoparticle, *Mater. Trans.* 62 (2021) 829.
- [16] L. Zhang, K. Qian, B.W. Schuller, Y. Shibuta, Prediction on mechanical properties of non-equiaxomeric high-entropy alloy by atomistic simulation and machine learning, *Metals* 11 (2021) 922.
- [17] L. Zhang, K. Qian, J. Huang, M. Liu, Y. Shibuta, Molecular dynamics simulation and machine learning of mechanical response in non-equiaxomeric FeCrNiCoMn high entropy alloy, *J. Mater. Res. Tech.* 13 (2021) 2043.
- [18] M. Grant, M.R. Kunz, K. Iyer, L.I. Held, T. Tasdizen, J.A. Aguiar, P.P. Dholabhai, Integrating atomistic simulations and machine learning to design multi-principal element alloys with superior elastic modulus, *J. Mater. Res.* 37 (2022) 1497.
- [19] N. Yadav, N. Chakraborty, A. Tewari, Interval prediction machine learning models for predicting experimental thermal conductivity of high entropy alloys, *Comp. Mater. Sci.* 214 (2022) 111754.
- [20] Q. Kong, Y. Shibuta, High-precision prediction of thermal conductivity of metals by molecular dynamics simulation in combination with machine learning approach, *Mater. Trans.* 64 (2023) 1241.
- [21] D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules, *J. Chem. Inf. Comput. Sci.* 28 (1988) 31.
- [22] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, *IEEE Trans. Neural Netw.* 5 (2016) 61.
- [23] S. Nosé, A unified formulation of the constant temperature molecular dynamics methods, *J. Chem. Phys.* 81 (1984) 511.
- [24] W.G. Hoover, Canonical dynamics: Equilibrium phase-space distributions, *Phys. Rev. A* 31 (1985) 1695.
- [25] G.P. Purja Pun, Y. Mishin, Development of an interatomic potential for the Ni-Al system, *Phil. Mag.* 89 (2009) 3245.
- [26] S.J. Plimpton, Fast parallel algorithms for short-range molecular dynamics, *J. Comput. Phys.* 117 (1995) 1.
- [27] A. Stukowski, Visualization and analysis of atomistic simulation data with OVITO—the open visualization tool, *Model. Simul. Mater. Sci. Eng.* 18 (2010) 015012.
- [28] A. Stukowski, Structure identification methods for atomistic simulations of crystalline materials, *Model. Simul. Mater. Sci. Eng.* 20 (2012) 045021.
- [29] K. Oguchi, Y. Shibuta, T. Suzuki, Accelerating molecular Dynamics Simulation performed on GPU, *J. Jpn. Inst. Met.* 76 (2012) 462.
- [30] W. Hamilton, R. Ying, J. Leskovec, Inductive representation learning on large graphs, *Proc. 31st Int. Conf. Neural Info. Process. Sys. (NIPS17)* (2017) 1025.
- [31] I. Loschilov, F. Hutter, SGDR: Stochastic Gradient Descent with Warm Restarts, *Proc. 5th Int. Conf. Learning Representations (ICLR)* (2017).
- [32] D.P. Kingma, J. Ba, Adam: a Method for Stochastic Optimization. *Proc. 3rd Int. Conf. Learning Representations (ICLR)* (2015).
- [33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison,

- A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, *Adv. Neural Inf. Proces. Syst.* 32 (2019) 8024.
- [34] M. Fey, J.E. Lenssen, Fast graph representation learning with PyTorch Geometric, arXiv:1903.02428 (2019).
- [35] S. Orihara, Y. Shibuta, T. Mohri, Molecular dynamics simulation of nucleation from undercooled melt of nickel-aluminum alloy and discussion on polymorphism in nucleation, *Mater. Trans.* 61 (2020) 750.
- [36] L. Chalamet, D. Rodney, Y. Shibuta, Coarse-grained molecular dynamic model for metallic materials, *Comp. Mater. Sci.* 228 (2023) 112306.