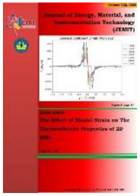
JOURNAL OF ENERGY, MATERIALS, AND INSTRUMENTATION TECHNOLOGY

Journal Webpage https://jemit.fmipa.unila.ac.id/



Hybrid DFT-ML-MD Approach for Derivation of Lennard-Jones Interatomic Potential Parameters of Al

Artoto Arkundato[1*], Widiasih[2], Anak Agung Istri Ratnadewi[3], Khalif Ardian Syah[4], Yanti Yulianti[5]

1. Physics Department, Faculty of Mathematics and Natural Sciences, Universitas Jember, Indonesia
2. Physics Education, Universitas Terbuka, Indonesia
3. Chemistry Department, Faculty of Mathematics and Natural Sciences, Universitas Jember, Indonesia
4. Master's Student, Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Indonesia
5. Physics Department, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Indonesia

Article Information

Article history:
Received June 8, 2025
Received in revised form June 11, 2025
Accepted June 12, 2025

Keywords: *Curve-Fitting, DFT, MD, Machine-Learning, Lennard-Jones Potential*

Abstract

Atomistic simulation based on computational physics of methods is used to develop accurate interatomic potentials based on DFT (density functional theory) data. The accuracy of predicting the physical properties of a material is highly dependent on the quality of the interatomic potential used. The purpose of this study is to determine the Lennard-Jones potential parameters of Al metal (epsilon and sigma) from fitting the DFT simulation output data. The use of a "robust" fitting method to reduce the influence of outliers on the potential results is very important and therefore a machine learning method is used to help find the right potential parameters. The method used is a hybrid method using DFT to generate training data, using ML (machine learning) to fit DFT data to the Lennard-Jones (LJ) potential model, and using the MD (molecular dynamics) method to validate the LJ potential parameters. Python-based programming is applied to facilitate how the three methods can be connected. The results of this study are that Al metal has an epsilon value = 0.5000 eV and sigma Al = 3.2072 Å, with a regression coefficient R2 = 0.9441 so that it can be concluded that this study can be said to be quite good and the hybrid method can be further developed to obtain the LJ potential parameter values of various other materials, especially metals.

Informasi Artikel

Proses artikel:
Diterima 8 Juni 2025
Diterima dan direvisi dari 11 Juni 2025
Accepted 12 Juni 2025

Kata kunci: *Curve-Fitting, DFT, MD, Machine-Learning, Lennard-Jones Potential*

Abstrak

Simulasi atomistik yang berbasis pada metode fisika komputasional digunakan untuk mengembangkan potensial interatomik yang akurat berdasarkan data DFT (density functional theory). Ketepatan prediksi sifat fisik suatu material sangat bergantung pada kualitas potensial interatomik yang digunakan. Tujuan dari penelitian ini adalah untuk menentukan parameter potensial Lennard-Jones logam Al (epsilon dan sigma) dari hasil pencocokan data keluaran simulasi DFT. Penggunaan metode pencocokan yang "robust" untuk mengurangi pengaruh outlier pada hasil potensial sangat penting dan oleh karena itu digunakan metode machine learning untuk membantu menemukan parameter potensial yang tepat. Metode yang digunakan adalah metode hybrid dengan menggunakan DFT untuk menghasilkan data latih, menggunakan ML (machine learning) untuk mencocokan data DFT dengan model potensial Lennard-Jones (LJ), dan menggunakan metode MD (molecular dynamics) untuk memvalidasi parameter potensial LJ. Pemrograman berbasis Python diterapkan untuk memfasilitasi bagaimana ketiga metode tersebut dapat dihubungkan. Hasil dari penelitian ini adalah logam Al memiliki nilai epsilon = 0,5000 eV dan sigma Al = 3,2072 Å, dengan koefisien regresi R2 = 0,9441 sehingga dapat disimpulkan bahwa penelitian ini dapat dikatakan cukup baik dan metode hybrid ini dapat dikembangkan lebih lanjut untuk mendapatkan nilai parameter potensial LJ berbagai material lain khususnya logam.

---

* Corresponding author.
E-mail address: a.arkundato@unej.ac.id

1. Introduction

Materials science is a multidisciplinary field that studies the structure, properties, processing, and performance of materials. The key objective is to understand how microscopic features, such as atomic arrangement and microstructure affect macroscopic properties like strength, conductivity, and thermal resistance. While not always required, beginning with a microscopic study is highly recommended, as microstructure plays a crucial role. It includes aspects like grain arrangement, dislocations, phases, and porosity on a micrometer or nanometer scale. These features strongly influence mechanical and electrical properties, material degradation, and phase transformations. For example, steel with finer grains is typically stronger (Hall-Petch effect), while ceramics with larger pores tend to be more brittle. Thus, understanding microstructure is essential in predicting and tailoring the behavior of materials. In materials design and development, microstructural analysis often serves as the first step toward engineering materials with targeted properties for specific applications.

How do technology and computational methods of materials play a role? In recent decades, technological advances, particularly in computational methods, have become essential tools in the discovery and development of new materials. The emergence of machine learning has further accelerated this progress, offering new ways to predict and optimize material properties. With increasingly powerful computers and advanced numerical algorithms, researchers can now simulate the behavior of materials without needing to perform time-consuming and costly laboratory experiments. This allows for efficient exploration of material systems, including those that are challenging or even dangerous to replicate in lab environments. For instance, observing material degradation inside an operating nuclear reactor is extremely difficult due to the extreme heat and pressure involved. Computational simulations offer a safe and effective alternative. Two major atomistic simulation methods in materials science are classical Molecular Dynamics (MD) and Density Functional Theory (DFT). MD is particularly useful for studying time-dependent (dynamic) processes, while DFT excels at calculating static quantum properties of materials. These methods have even been combined into a hybrid technique called ab-initio molecular dynamics (AIMD), which offers both accuracy and dynamic insight (Iftimie et.al, 2005). These computational tools are now central to materials design and innovation.

Molecular Dynamics (MD) is a computational technique that simulates the motion of atoms and molecules over time under specified conditions such as temperature and pressure. In MD, atomic or molecular motion is governed by Newton's laws, and interactions between atoms are described using interatomic potentials such as the Lennard-Jones potential (Lennard-Jones, 1924). This allows us to calculate the position and velocity of particles at any given time within an evolving system. With the help of statistical physics, we can extract physical quantities of interest, such as pressure, temperature, and diffusion coefficients, from these time-dependent simulations. The core idea is that different molecular shapes and material structures lead to different chemical and physical properties. Hence, MD enables us to explore a wide range of compositions and element combinations to design material systems with potentially superior properties.

What about Density Functional Theory (DFT)? While MD excels in studying the dynamic properties of large atomic systems in a way that approximates real experimental conditions, DFT is more suited for accurately predicting the electronic properties and total energy of smaller systems. DFT can model chemical bonding, reactivity, and electronic band structures, often without requiring empirical parameters. Essentially, DFT is a numerical method for solving the Schrödinger equation for many-body quantum systems. However, DFT is computationally expensive and typically limited to systems containing only a few dozen atoms on a standard PC. Simulating hundreds or thousands of atoms requires access to high-performance supercomputers. On the other hand, MD is more scalable for large systems and longtime simulations but relies heavily on the choice of empirical interatomic potentials. In modern materials science, both MD and DFT serve complementary roles and are often used together to gain a comprehensive understanding of materials.

That is the urgency of using computational methods in the development and design of new materials with potential applications in future technologies. While not all materials research must begin with computation, and not every semiconductor study starts with simulations like DFT or MD, computational approaches have become common and strategic starting points in modern research. These methods help evaluate whether a proposed semiconductor structure is stable under specific conditions, which is the main focus of this study. To estimate physical properties, we use Molecular Dynamics (MD) simulations. MD requires a potential model to describe interatomic interactions, such as the Lennard-Jones (LJ) potential used here. Building this potential model requires reference data, which we obtain from either experiments or high-accuracy simulations like Density Functional Theory (DFT). In our workflow, we extract cohesive energy and lattice constant values from DFT and use them to construct the LJ potential. To determine the two LJ parameters, epsilon (well depth) and sigma (atomic size), we apply a simple machine learning (ML) method: curve fitting. Although more complex ML tools like Artificial Neural Networks (ANN) are increasingly popular, they often require significant computational resources and specialized frameworks. In contrast, basic curve fitting in Python is sufficient for our needs. Our goal is to show that LJ potential parameters for a material can be obtained through this method. While the LJ model is less accurate than other interatomic potentials like EAM, MEAM, or ML-based approaches, it remains useful for understanding general trends in material behavior. It is like using a voltmeter that is not perfectly calibrated, still useful if used consistently. In this work, we demonstrate how to derive epsilon and sigma values for pure Aluminum (Al) using DFT data. These parameters are then optimized through ML-based fitting and tested in an MD simulation. This approach lays the groundwork for further simulations of more complex materials.

2. Research Methods

2.1 Density Functional Theory (DFT)

Basically, DFT is a method in quantum mechanics that is used to calculate the electronic structure of atoms, molecules, or solids. It can also be used to determine the total energy, electron density, energy bands, interatomic forces, and many other material properties. Basically, the DFT method is to solve the Schrodinger differential equation of a many-electron material system which, because for this system it is impossible to solve directly, is approached by the electron density functional theory method. DFT changes the many-body problem into a one-particle problem with an equation called the Kohn-Sham equation (Hohenberg, & Kohn, 1964; Kohn & Sham, 1965):

$$\frac{h^2}{8m\pi^2}\nabla^2 + V_{eff}(\boldsymbol{r})]\psi\_i(\boldsymbol{r}) = \varepsilon\_i\psi\_i(\boldsymbol{r}) \tag{1}$$

where $\Psi_i(\mathbf{r})$ is the Khon-Sham wave function (one-electron orbital), ε_i is the energy of the i-th orbital, and $V_{eff}(\mathbf{r})$ is the effective potential function of the electron density. From the DFT results, we can obtain information on the total energy of the system, see the optimal geometric structure, density of states (DOS) and band structure which are important for understanding the electrical and optical properties of the material. We will not discuss the solution of the Kohn-Sham equation in theory, but in this study, we will use a very popular code in material computing, namely Quantum Espresso (QE). The QE code applies the DFT method in studying the physical properties of material systems, and in general almost all material systems can be studied with this code such as metals (such as pure and alloys), molecules (such as TiO_2 multifunctional materials), ceramics (such as SiC), composites (such as FeNi) and so on. The limitation of using this QE DFT code is the need of providing better hardware for studying more complex structures. For small-scale research you may be able to use a PC with the latest processor with GPU support for fairly good results. If you want a medium scale (the number of atoms is increasing and the complexity is increasing) you may need an HPC (high-performance computing) cluster. And if you want to simulate a more difficult material system such as a complex nanostructure, it may be better to use a supercomputer. In this study, DFT simulations are used to calculate the total energy of the material system.

2.2 Lennard-Jones Potential

Every object has a shape and according to atomic theory objects are composed of molecules, and each molecule is composed of atoms. Differences in shape (geometrically) of an object can be caused by the arrangement/structure and type of bonds between atoms that form it. One model of bonds between atoms is the pair-wise bond which is a simple model. Pair-wise means based on the pair-wise bond of two objects. In the context of potential or atomic interaction, pair-wise potential means that the total energy of the system will be calculated as the sum of all interactions between pairs of atoms i and j:

$$V_{\text{total}} = \Sigma V(\boldsymbol{r}_{ij}) \tag{2}$$

The Lennard-Jones potential is pair-wise because it only considers the interactions of two atoms modeled by explicit formulas (Stephan & Hasse, 2024):

$$V(\boldsymbol{r}) = 4\varepsilon[(\sigma/r)^{12} - (\sigma/r)^{6}] \tag{3}$$

The characteristic of the pair-wise potential is that it is computationally simple and fast, and does not take into account the effects of three or more directly bonded atoms (such as angles or torques). In molecular dynamics simulations, forces, potentials, and Newton's laws are directly related through the relation:

$$F = -\, \mathrm{d}V/\mathrm{d}r \tag{4}$$

with V is the potential energy of the system, for example Lennard-Jones, Morse, EAM and others. After the force is obtained, then with Newton's law we get the relation:

$$a = F/m \tag{5}$$

Once the acceleration is known, a numerical integration algorithm (such as Verlet, Velocity-Verlet, or Leapfrog) can be used to calculate the position and velocity at subsequent times:

$$r\_i(t + \Delta t) = r\_i(t) + v\_i(t) \times \Delta t + (1/2) \times a\_i(t) \times \Delta t^2 \tag{6}$$

Since potential energy is energy associated with position, it will be calculated from the structure of matter. Then the force is calculated from the potential. Then the acceleration from Newton's law. So that the new position of a moving object will be updated through integration or the formula above. And so on, the process will repeat step by step integration step until the position of the object is known entirely within the time span being evaluated. The solution to Newton's equation of motion is therefore none other than the trajectory of the object's motion, namely the path formed from the positions of atoms at any time. How is this trajectory information used by MD simulations to calculate physical quantities? MD code designers will try to find a relationship between position and the physical quantities sought, for example one of which is the RDF radial distribution function related to the function $g(r)$:

$$g(r) = \langle n(r)\rangle \;/\; (4 \times \pi \times r^2 \times \rho \times \Delta r) \tag{7}$$

The Radial Distribution Function $g(r)$ describes the relative probability of finding a pair of atoms at a distance r, compared to a random system (ideal gas). In this case $n(r)$ is the number of atoms in the shell between r and $r + \Delta r$ from the central atom, ρ is the average atomic density and Δr is the bin width. Its value is calculated from the position data of all atoms during the simulation time. Each snapshot of the atomic position produces a pair of distances

$r_{ij}=|\vec{r}_i-\vec{r}_j|$ and all these distances are the basis for the construction of the histogram $g(r)$. The function $g(r)$ reflects the structure of the material where for a solid crystal it will have sharp peaks indicating that the atomic positions are arranged periodically.

2.3 Cohesive Energy

Cohesive energy is the energy required to separate the bonded atoms that form a crystalline solid into their individual neutral, unbonded atoms, into a free gaseous state. Therefore, it is the total bond energy per atom in a solid. This is similar to ionization energy but for atomic systems not material systems. Mathematically the cohesive energy of a material system is formulated:

$$E_{\text{Cohesive}} = E_{\text{Free\_atom}} - E_{\text{solid\_atom}} \tag{8}$$

$E_{\text{Free\_atom}}$ is the energy of one atom in the free state, $E_{\text{solid\_atom}}$ is the total energy per atom in the solid structure (bulk). So, if the system is a supercell, then $E_{\text{solid\_atom}}$ is the total energy of the supercell divided by the number of atoms in the supercell. For example, if your DFT simulation supercell is a 2x2x2 fcc crystal then the number of atoms is $N = 32$. The total energy of the system is the kinetic energy plus the potential energy. To obtain the value of cohesive energy, we do not need to consider kinetic energy because, cohesion is a ground state concept, which means the temperature is at 0 K. Therefore, all calculations of cohesive energy are done in conditions without atomic motion, so only the potential energy $V(r)$ is relevant. Cohesive energy is specifically calculated using the total potential energy in the ground state:

$$E_{\text{Cohesive}} = E_{\text{free\_atom}} - (E_{\text{total\_potential\_crystal}})/N \tag{9}$$

The concept of fitting Lennard-Jones parameters is carried out so that the potential function can reproduce the physical properties of the target such as cohesive energy and lattice constant of a material. Let's discuss it step by step conceptually and technically. The LJ potential parameter epsilon ε is related to the minimum depth of the potential and this is of course related to the cohesive energy. The parameter σ is related to the distance at which the LJ potential becomes zero, so this is certainly related to the distance between atoms, and affects the lattice constant. Then r is the distance between two atoms in a pair. The definition of a pair here should not be interpreted as the closest pair, but all possible pairs of two atoms in the material must be seen and their interactions calculated. It's just that for simplification and ignoring very small errors, the interaction of very far-flung pairs (further than r_{cut}) is usually ignored. So, the total energy calculated is the energy of the entire system up to this distance r_{cut}. This cohesive energy is usually negative, indicating that the atoms in the solid are more stable than in a free state (because they have formed bonds). **Figure 1** is the general form of LJ potential.

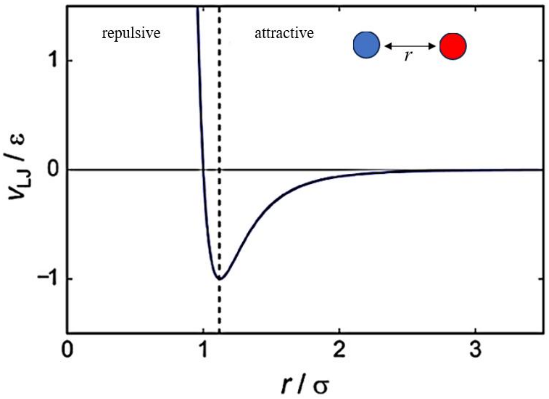


Figure 1 Lennard-Jones Potential Curve (Lenhard et.al., 2024)

2.4 Quantum Espresso dan Simulation Procedure

Quantum ESPRESSO (QE) is an open-source software package based on first-principles (ab initio) used to perform material simulations at the atomic and electronic levels using density functional theory. In this study, we used the QE code PWSCF Program v.6.7MaX (Giannozzi et al., 2009, 2017). This QE program or code is widely used in material computation research and is always updated every year (Aravindh et al., 2007). For the installation and options for running the QE code, the installation and usage instructions available in the official documentation (http://www.quantum-espresso.org), but in general we use the command "pw.x < Al.in > Al.out" to obtain the DFT training data needed in this research, especially the total energy value of the system. Then the complete QE input script used in this study is currently not publicly available because it is still in the documentation process. However, we plan to make it available through our website (https://sites.google.com/view/riset-arkundato/home) in the near future. For those who need earlier access, please contact the corresponding author via email." The contents of the QE input script template that we used in the study:

```
&CONTROL
  calculation = 'scf',                      #change to relax for relaxation
……………………
/
&SYSTEM
  ibrav = 1,
  celldm(1) = ……,                           #adjusted to the size of the supercell"
  nat = 32,                                 #number of atoms in supercell fcc 2x2x2
  ntyp = 1,
  ecutwfc = 30.0,
  ecutrho = 240.0,
  occupations = 'smearing',
  smearing = 'mp',
  degauss = 0.02,
/
&ELECTRONS
  conv_thr = 1.0d-8,
  mixing_beta = 0.7,
/
ATOMIC_SPECIES
  Al 26.98 Al.pbe-n-kjpaw_psl.1.0.0.UPF
ATOMIC_POSITIONS (crystal)                  #32 lines of atomic positions in supercell
 Al  0.000000  0.000000  0.000000
 ……………………………………
 Al  0.750000  0.750000  0.500000
K_POINTS (automatic)
```

To run this DFT QE simulation you must provide a Pseudopotential file, which in this study we use the PP file “Al.pbe-n-kjpaw_psl.1.0.0.UPF” (https://pseudopotentials.quantum-espresso.org/upf_files/Al.pbe-n-kjpaw_psl.1.0.0.UPF).

The output file (Al.out) will contain the results of the SCF simulation and the total energy data can be extracted from the Al.out file. To provide a complete picture of how the research was carried out, a flowchart for the simulation procedure of hybrid simulation is provided as in **Figure 2**.

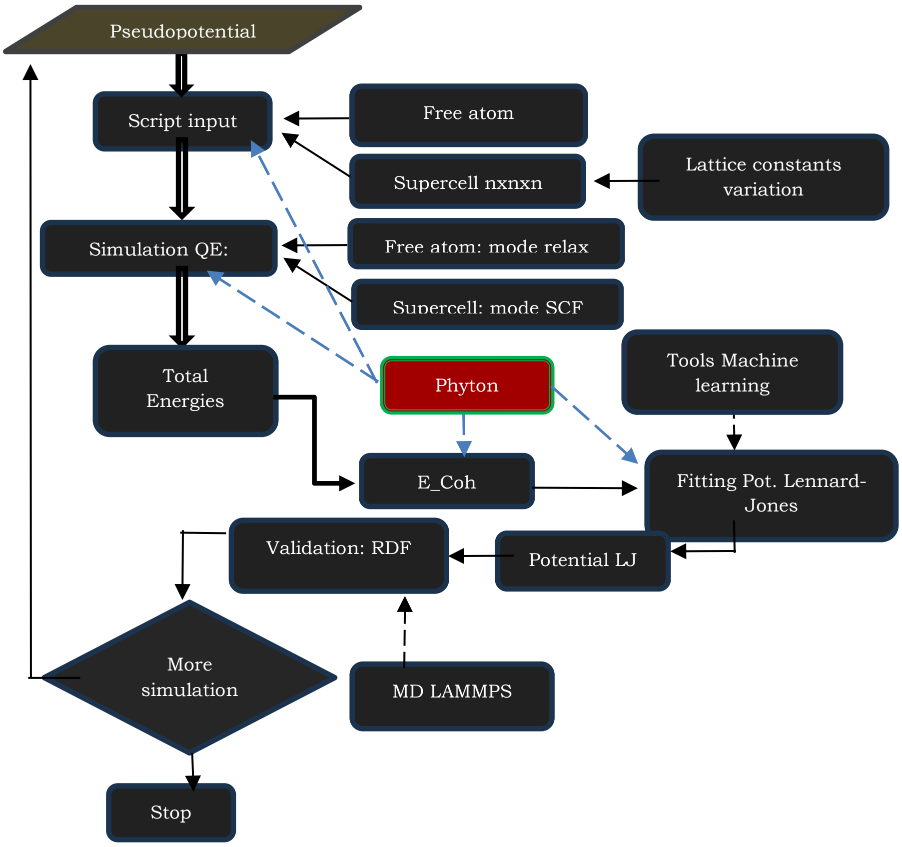


Figure 2 Flow Diagram for Hybrid Simulation

Then, for the fitting procedure to obtain the LJ potential parameters (epsilon and sigma), it can be arranged in general in the following instruction blocks:

```
#1. library/tools for fitting process
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
import datetime
```

```
#2. Potential function
def lennard_jones_potential(r, epsilon, sigma, offset=0)
offset >> added so that the saturation energy at long distances matches the
DFT data.
```

```
#3. Entry of DFT simulation result data for N constant-lattice variations
lattice_constants = np.array([...])
cohesive_energies = np.array([...])
```

```
#4. Initial parameter estimation and initial guess
Calculated from the minimum energy position (with np.argmin()).
Using relation $r_{min}=2^{1/6}\cdot\sigma$ → Based on thgeory LJ.
initial_guesses = [
    [epsilon_better, sigma_better, offset_better],
    [3.5, 3.6, -1.5],
    [4.0, 3.5, -2.0],
    [2.0, 3.7, -1.7]
]
```

```
#5. Fitting function using curve_fit from scipy.optimize
popt_temp, pcov_temp = curve_fit(
    lennard_jones_for_fit,
    lattice_constants,
    cohesive_energies,
    p0=guess,                  # Initial guess
    bounds=bounds,             # upper and lower limits of parameter
    method='trf',              # trust region reflective
    max_nfev=20000        # maximum number of function evaluations
)
```

```
#6. Select the best results
Every time the curve_fit is successful, the RMSE value is calculated:rmse_temp
= calculate_rmse(popt_temp, lattice_constants, cohesive_energies)
```

```
#7. Final Result
If fitting is successful:
Take the fitting result parameters: epsilon_fit, sigma_fit, offset_fit
Calculate:
Minimum position rmin
R-squared (coefficient of determination)
RMSE
Display the results in the terminal and graphical visualization
```

The realization of the above blocks for LJ potential fitting in the form of a complete python program script is not yet publicly available because it is still in the documentation process. However, we plan to provide it through our website (https://sites.google.com/view/riset-arkundato/home) in the near future. For those who need earlier access, please contact the corresponding author via email. "The partial contents of the fitting script that we used in the study are as follows:

```
# Initial Estimation and Fitting
from scipy.optimize import curve_fit
import numpy as np

def lennard_jones_for_fit(r, epsilon, sigma, offset):
    return lennard_jones_potential(r, epsilon, sigma, offset)

# Automatic initial estimate
idx_min = np.argmin(cohesive_energies)
```

```
r_eq = lattice_constants[idx_min]
e_min = cohesive_energies[idx_min]
sigma_init = r_eq / (2**(1/6))
epsilon_init = abs(e_min)
offset_init = cohesive_energies[-1]
p0 = [epsilon_init, sigma_init, offset_init]

bounds = ([0.5, 3.0, -5.0], [8.0, 5.0, 0.0])            #can be adjusted

popt, pcov = curve_fit(lennard_jones_for_fit, lattice_constants, cohesive_energies,
                  p0=p0, bounds=bounds)

#
import matplotlib.pyplot as plt

epsilon_fit, sigma_fit, offset_fit = popt
r_min_fit = sigma_fit * (2**(1/6))
r_fit = np.linspace(min(lattice_constants)-0.1, max(lattice_constants)+0.1, 500)
e_fit = lennard_jones_for_fit(r_fit, *popt)

plt.plot(r_fit, e_fit, label='Fitting LJ')
plt.scatter(lattice_constants, cohesive_energies, color='red', label='DFT data')
plt.axvline(x=r_min_fit, linestyle='--', color='gray', label=f'r_min = {r_min_fit:.3f}')
plt.xlabel('Lattice constant (Å)')
plt.ylabel('Cohesive energy (eV)')
plt.legend()
plt.title('Lennard-Jones Fit')
plt.grid(True)
plt.show()
```

3. Results and Discussions

3.1 DFT Simulation Results

There are 4 important stages based on the flow diagram in Figure 2, namely (1) creation of DFT quantum espresso simulation input (2) pw.x simulation produces training data for lattice constants and total energy and cohesive energy (3) Lennard-Jones potential fitting with simple machine learning curve-fit and (4) validation of epsilon and sigma potential parameters in MD lammps simulation. From the training data produced, we still use a small supercell size due to limited computing resources, namely only using a 2x2x2 supercell with a total of 32 atoms. For better results, it can be increased to a larger size. DFT usually uses periodic boundary conditions (PBC). If you only involve one small unit cell in the simulation, then the atoms or defects in it will interact with their own shadows in neighboring cells (nearest-neighbor), due to PBC. A larger supercell size can reduce this interaction, so that our simulation results will more realistically describe the real state of the material system.

Then we still use a small K-POINTS size in this study, namely 2 2 2 0 0 0 to avoid long DFT simulation times. This can be increased even more if using a more sophisticated computer. For DFT simulations, setting a larger or denser K-POINTS size (points in the reciprocal space for Brillouin zone integration) is very important to obtain more accurate and convergent results. In general, the more K-POINTS, the more accurate the integration results in Brillouin space and the more precise the total energy of the system. Conversely, if K-POINTS is too small, rough or fluctuating results can occur, resulting in misjudgment of structural stability or defects. If K-POINTS is too small, the total energy may not converge and the energy/force gradient may not be smooth, affecting relaxation or training data. In this study, we have not had time to explore the convergence of K-POINTS, but the procedure can be done as follows:

```
for k in 2 3 4 5 6; do
 edit input file: set K_POINTS automatic $k $k $k 0 0 0
run pw.x
 grep ! output_file  # get total energy
done
```

In the python script snippet above we try to vary K-POINTS with different sizes 2×2×2, 3×3×3, 4×4×4, 5×5×5, 6×6×6. Then take the total energy value for example = -14.32847923 Ry. Then calculate E_per_atom = E_total/32, if the number of atoms in the supercell is 32. Then plot the results: k-point vs total energy per atom. Then choose the k-point that provides stable energy within the limit <1 meV (0.001 eV). If it is seen in the K-POINTS size 5×5×5 to 6×6×6 there is a change <1 meV, then 5×5×5 is sufficiently convergent. But once again in this study we have not conducted a K-POINTS convergence test.

3.2 Parameter of Lennard-Jones Potential

From the results of this study, the results of the Lennard-Jones potential fitting of the DFT quantum espresso cohesive energy data displayed on the Linux display can be obtained as follows:

epsilon = 0.5000 eV
sigma = 3.2072 Å
offset = -2.0439 eV
*r*_min = 3.5999 Å
R-squared = 0.9441
RMSE value of 1.2920 eV

R-squared (R^2 = 0.9441) measures how well our model explains the variation in the data. This value means that the model explains ~94% of the variation in the cohesive energy. This is quite good, but not perfect as there is still ~6% of the energy variation not explained by the Lennard-Jones model. From the results of this study, we have a variation table:

Lattice constant = [2.6000, 2.9778, 3.3556, 3.7333, 4.1111, 4.4889, 4.8667, 5.2444, 5.6222, 6.0000]
Cohesive energy = 15.16645, 3.14487, -1.69774, -3.35810, -3.61452, -3.33264, -2.88641, -2.42931, -2.02114, -1.66890]

From the data above, we have a range of Cohesive Energy from the largest to the smallest of 15.16645 – (-3.61452) = 18.78097 eV. The *RMSE* of 1.2920 eV means that our model has an average prediction error of 1.2920 eV from the calculated cohesive energy value at a certain lattice constant. In this case, the *RMSE* value of 1.2920 eV is not too large when compared to the cohesive energy range of 18.78097 eV so it is quite good at predicting the cohesive energy, although there is still a prediction error of around 1.2920 eV at each data point. The data fitting from the DFT data to the Lennard-Jones potential form can be seen in **Figure 3.**

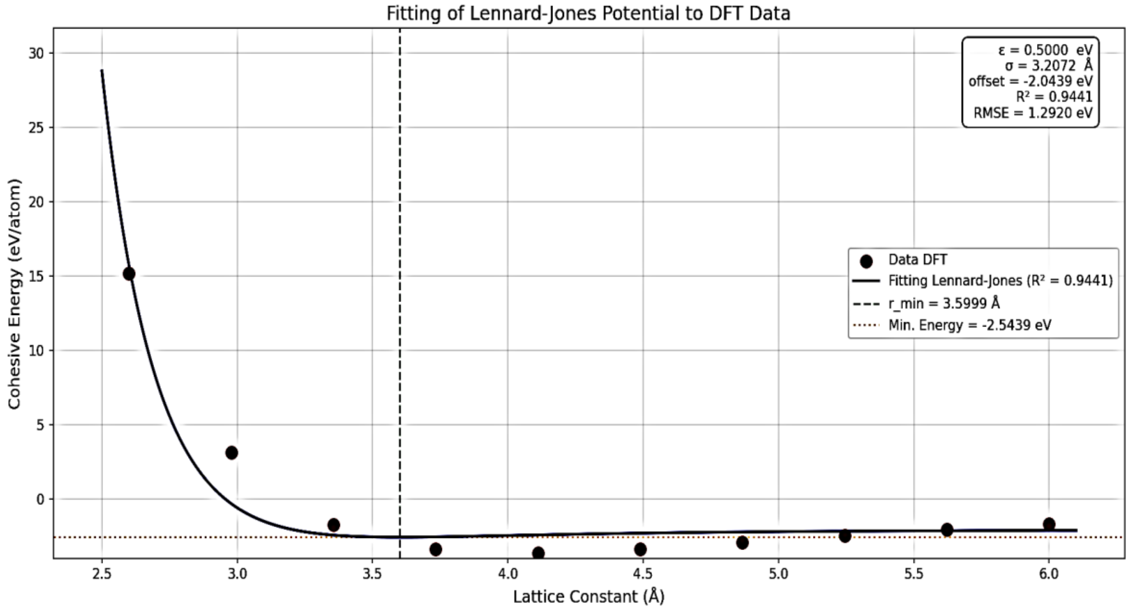


Figure 3 Plot of the fitting results of DFT data against the LJ potential model.

3.3 MD Simulation and Validation of LJ Potential Parameters

We have obtained the results of the LJ potential parameters from the fitting procedure above. Now we will validate it in a real case of molecular dynamics simulation with the LAMMPS program (Plimpton, 1995; Thompson et al., 2022). We want to see if our potential parameters are physically able to describe the physical phenomena that we observe. In this case, we want to focus on seeing whether the structure of the representative system with the LJ interaction potential with epsilon and sigma that we obtain. We can test it with a more complex simulation, but in this study, we only want to see how the structure of the material is seen from the RDF curve of the LAMMPS simulation results for validation (Maulana et al., 2019; Maghfiroh et al., 2020; Mardiyah et al., 2020). The LAMMPS input script snippet for validation is more or less as follows:

```
# -----------------------------
# Lattice constant Al FCC
# -----------------------------
lattice fcc 4.04
region box block 0 6 0 6 0 6
……………………………
# -----------------------------
# LJ Potential definition
# -----------------------------
pair_style lj/cut 8.5
pair_coeff 1 1 0.5000 3.2072   # epsilon (eV), sigma (Angstrom)
```

```
................................
dump 1 all custom 100 traj_al_lj.lammpstrj id type x y z vx vy vz
fix 2 all npt temp 10.0 10.0 0.1 iso 0.0 0.0 1.0
run 10000
write_data  final_structure.lmp
```

This LAMMPS input script will directly generate the FCC crystal structure for Al metal which will be simulated at a low temperature of 10 K. We simulate at this temperature because our training data from DFT is also taken at this low temperature approaching absolute zero. The LAMMPS input script above if run will generate a coordinate file “final_structure.lmp” which if opened with the Ovito program we can save the RDF data file from the “manipulation >> coordination number” menu facility (Stukowski, 2010). From the g(r) data file obtained from OVITO we can plot this function with the python script creation, as in **Figure 4**.

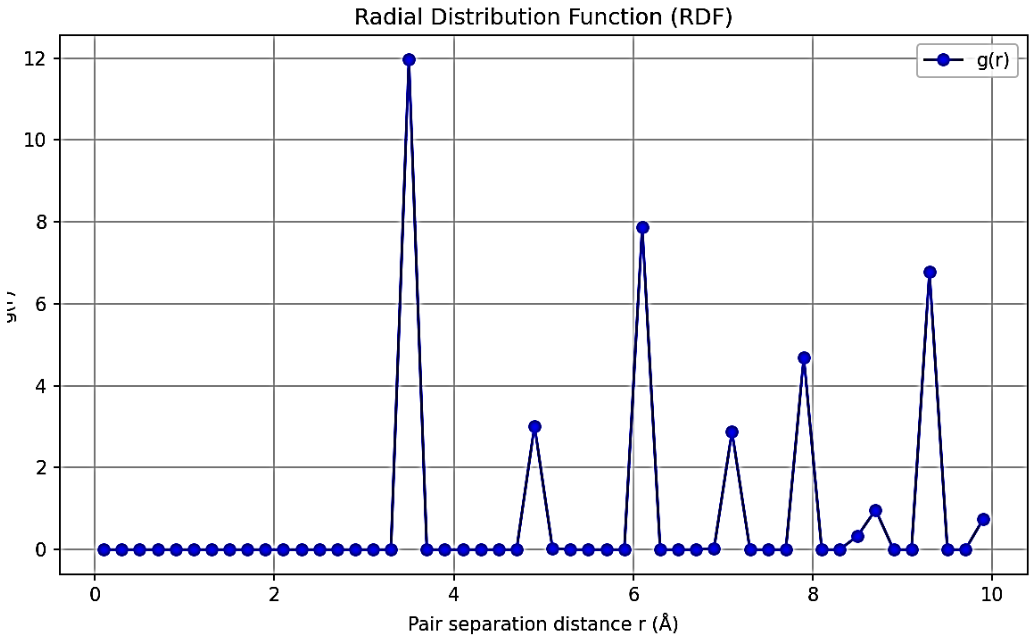


Figure 4 RDF curve for Al metal at a low temperature of 10K

From the RDF curve above we can read that the first peak is very high around 3.8 Å, with the value of the radial distribution function *g*(*r*) > 11. Then followed by the next peaks around: 5.4 Å, 6.5 Å, 7.6 Å, 8.8 Å, 10 Å which are still quite sharp and separated at a certain distance. This clearly shows the pattern of a crystalline solid that has a sharp, high, and regular RDF curve pattern and indicates a solid crystalline structure, not a liquid. Then because there is no significant broadening of the peaks, this system is likely at a low temperature (approaching 0–300 K). This indicates minimal atomic vibrations and very small disorder. Then the RDF does not approach 1 at *r* > 10 Å where in a fairly large system and liquid phase, the RDF will approach 1 at a long distance, while in a solid crystal system or small simulation, the RDF still has an oscillatory structure. Let’s discuss the relationship of the RDF curve you uploaded with the parameters *ε* (epsilon) and *σ* (sigma) of the Lennard-Jones potential. The RDF interpretation of the image can be seen from the RDF curve: The first peak is very sharp and high, located around *r* ≈ 3.6–3.8 Å. The subsequent peaks decrease but are still quite well defined. The curve shows a crystalline structure (probably fcc), not a liquid. The σ parameter is the distance at which the Lennard-Jones potential = 0. The RDF of the first peak usually occurs around:

$$r_{\min} = 2^{1/6} \times \sigma = 1.122 \times \sigma$$

If we go back from the first peak, r ~ 3.6 Å then

$$\sigma = 3.6/1.122 = 3.21 \text{ Å}$$

This matches the sigma value we used, *σ* = 3.272 Å. What about epsilon? The parameter ε describes the depth of the LJ potential well, that is, how strong the interactions between atoms are. From the RDF curve it shows that if the peaks are high then the system is highly structured, and if the order is long-range then the atoms tend to stay in their crystal positions. This means that ε is large enough that the system is thermally stable and does not melt at your simulation temperature (probably 0.1–0.3 eV is the minimum, in this simulation from the DFT fitting results we used 0.5 eV, so it is more or less appropriate).

4. Conclusion

This study aims to determine the Lennard-Jones potential parameters of Aluminum (Al) metal from fitting quantum DFT simulation data using the Quantum Espresso code. DFT data fitting was performed on the Lennard-Jones potential function form using one of the Curv-fit machine learning tools. The results of this computational (physics) research indicate that the application of the quantum hybrid method (DFT Quantum Espresso) and classical (MD Lamps) with the help of machine-learning (ML) is able to provide fairly representative results for the epsilon and sigma parameters of the Lennard-Jones potential. The findings of this study are that epsilon Al = 0.5000 eV and sigma Al = 3.2072 Å with $R^2 = 0.9441$ are good enough to describe the binding force of Al metal atoms with R^2 quite well although not equal to 1. However, this is good enough to be used as an initial prediction of the simulation of the physical properties of Aluminum through MD simulations as shown by the study of the RDF of Al metal. From the study of the RDF curve, it shows that the epsilon and sigma parameters of Al metal that were tested in MD simulations with the LAMMPS code can fairly consistently describe the structural properties of Al metal at low temperatures to room temperature, so that it can be used for rough predictions of the physical properties of Al metal at low temperatures to room temperature. However, this study still has limitations in terms of the small amount of training data used in data fitting, the number of supercell atoms is only 32 which can be increased again to a 3x3x3 supercell with 108 atoms. Then the pseudopotential file in the DFT simulation has only been tested with one type, namely Al.pbe-n-kjpaw_psl.1.0.0.UPF. Also, K-POINTS in the DFT simulation to generate training data only uses the value 2 2 2 0 0 0 which can be increased to 6 6 6 0 0 0 to be more accurate. However, due to time and computing device limitations, this study only uses limited data. For this reason, further studies are recommended to be conducted to eliminate the shortcomings of this research.

Acknowledgments

The authors gratefully acknowledge the support from LP2M, University of Jember, through the 2024 PGB grant scheme.

Bibliography

Lennard-Jones, J. E. (1924). On the determination of molecular fields. I. From the variation of the viscosity of a gas with temperature. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 106(738), 441–462. https://doi.org/10.1098/rspa.1924.0086

Hohenberg, P., & Kohn, W. (1964). Inhomogeneous Electron Gas. Physical Review B, 136, 864-871. https://doi.org/10.1103/PhysRev.136.B864

Kohn, W., & Sham, L. J. (1965). Self-Consistent Equations Including Exchange and Correlation Effects. Physical Review, 140, A1133-A1138. https://doi.org/10.1103/PhysRev.140.A1133

Plimpton, S. (1995). Fast Parallel Algorithms for Short-Range Molecular Dynamics. J. Comp Phys, 117, 1-19.

Iftimie, R., Minary, P., & Tuckerman, M. E. (2005). Ab initio molecular dynamics: Concepts, recent developments, and future trends. Proceedings of the National Academy of Sciences, 102(19), 6654–6659. https://doi.org/10.1073/pnas.0500193102

Giannozzi, P., et al. (2009). Quantum ESPRESSO: a modular and open-source software project for quantum simulations of materials. J. Phys.: Condens. Matter, 21, 395502.

Aravindh, et al. (2007). SixC1−xO2 alloys: A possible route to stabilize carbon-based silica-like solids. Solid State Communications, 144, 273-276. https://doi.org/10.1016/j.ssc.2007.09.011

Giannozzi, P., et al. (2017). Advanced capabilities for materials modelling with Quantum ESPRESSO. J. Phys.: Condens. Matter, 29, 46590.

Stukowski, A. (2010). Visualization and analysis of atomistic simulation data with OVITO - the Open Visualization Tool. Modelling Simul. Mater. Sci. Eng., 18, 015012.

Maulana, A., Arkundato, A., Sutisna, & Trilaksana, H. (2020). Mechanical properties of Fe, Ni, and Fe-Ni alloy: Strength and stiffness of materials using LAMMPS molecular dynamics simulation. In H. Trilaksana, S. W. Harun, C. Shearer, & M. Yasin (Eds.), 2nd International Conference on Physical Instrumentation and Advanced Materials 2019, Article 0034046 (AIP Conference Proceedings; Vol. 2314). American Institute of Physics Inc. https://doi.org/10.1063/5.0034046

Maghfiroh, C. Y., Arkundato, A., Misto, & Maulina, W. (2020). Parameters (epsilon, sigma) of Lennard-Jones for Fe, Ni, Pb for Potential and Cr based on Melting Point Values Using the Molecular Dynamics Method of the LAMMPS

Program. Journal of Physics: Conf. Series, 1491, 012022. https://doi.org/10.1088/1742-6596/1491/1/012022

Mardiyah, R. U., Arkundato, A., Misto, & Purwandari, E. (2020). Energy Cohesive Calculation for Some Pure Metals Using the Lennard-Jones Potential in LAMMPS Molecular Dynamics. Journal of Physics: Conf. Series, 1491, 012020. https://doi.org/10.1088/1742-6596/1491/1/012020

Thompson, A. P., Aktulga, H. M., Berger, R., Bolintineanu, D. S., Brown, W. M., Crozier, P. S., 't Veld, P. J., Kohlmeyer, A., Moore, S. G., Nguyen, T. D., Shan, R., Stevens, M. J., Tranchida, J., Trott, C., Plimpton, S. J. (2022). LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. Comp Phys Comm, 271, 10817.

Lenhard, J., Stephan, S., & Hasse, H. (2024). On the History of the Lennard-Jones Potential. Ann. Phys. (Berlin), 536, 2400115. https://doi.org/10.1002/andp.202400115

Stephan, S., & Hasse, H. (2024). On the History of the Lennard-Jones Potential. Annalen der Physik, 536(5), 2400115. https://doi.org/10.1002/andp.202400115