Master in High Performance Computing

Machine Learning for Predicting Lattice Thermal Conductivity

Supervisor(s):
Stefano de Gironcoli,
Franco Pellegrini

Candidate:
Neeraj Kulhari

9th edition
2022–2023

Acknowledgements

First and foremost, I would like to express my deepest gratitude to the **International Centre for Theoretical Physics (ICTP)** and the entire **MHPC faculty**. Their generous financial support and dedicated mentorship have not only made this educational journey possible but have also been pivotal in shaping my career in High Performance Computing. Their backing has been deeply instrumental in developing my computational expertise.

I am deeply grateful for the constant availability and bright guidance provided by **Stefano de Gironcoli** and **Franco Pellegrini** throughout this project. Their expertise and mentorship have been truly invaluable to my work.

I am also indebted to **Ivan Girotto** for his valuable technical support, as well as to **Irina Davidenkova** for her insightful discussions, both of which greatly contributed to the success of this thesis.

I extend my thanks to the dedicated staff at **SISSA** and **ICTP** for their continuous assistance with administrative matters. Furthermore, I want to express my sincere gratitude to all my **MHPC colleagues** for fostering an international and friendly atmosphere, which significantly enriched the collaborative experience and made this journey so memorable.

Lastly, I want to express my profound appreciation to my **family** for encouraging me in this endeavor. You are very special to me, and I could not have reached this milestone without your unwavering love and support.

Abstract

This thesis presents a machine learning–accelerated approach for computing anharmonic interatomic force constants (IFCs), which are essential for predicting lattice thermal conductivity in crystalline materials. Traditionally, the calculation of these force constants requires a large number of computationally expensive single-point Density Functional Theory (DFT) evaluations. To address this computational bottleneck, **Machine Learning Interatomic Potentials (MLIPs)** are employed using the **PANNA (Properties from Artificial Neural Network Architectures)** framework. The layered chalcogenide GeS_2 is used as a representative case study.

During model development, an important limitation of standard MLIP training strategies is identified. Neural networks trained only on small structural perturbations tend to remain confined within a purely harmonic regime, preventing them from learning the highly distorted configurations that govern anharmonic lattice dynamics. This phenomenon is referred to as the ***harmonic trap***. To overcome this limitation, a **Query-by-Committee active learning workflow** is implemented and combined with *ab initio* Molecular Dynamics (AIMD) data to expand the sampled configuration space.

The resulting hybrid dataset enables the neural network potential to accurately represent both harmonic and anharmonic regions of the potential energy surface. As a result, the computational time required to evaluate 212 supercell configurations is successfully reduced from 55.80 **node-hours** of rigorous DFT calculations to just 7.03 **seconds** using the trained surrogate model. This corresponds to an extraordinary computational speedup of nearly **four orders of magnitude**. Furthermore, a systematic sensitivity analysis demonstrates that lattice thermal conductivity calculations are extremely sensitive to force prediction noise. To avoid artificial phonon scattering in the Boltzmann Transport Equation (BTE) solver, force prediction errors must remain below a critical threshold of $\sim 50\,\mu\mathbf{eV/\AA}$. The proposed active learning framework successfully achieves this level of physical accuracy while maintaining substantial computational efficiency. Overall, this work establishes a robust methodology for accelerating anharmonic lattice dynamics calculations and enabling high-throughput screening of

thermoelectric materials.

Contents

Chapter 1

Introduction

1.1 Motivation and Background

In recent decades, the transition toward renewable energy technologies has become increasingly important in order to reduce global dependence on fossil fuels. Among the various approaches being explored, thermoelectric (TE) materials offer a promising solution because they can directly convert waste heat into electrical energy.

A large fraction of the energy produced in industrial processes, power plants, and automotive engines is dissipated as waste heat. Thermoelectric systems provide a potential route for recovering a portion of this energy, thereby improving overall energy efficiency.

Despite these advantages, the widespread adoption of thermoelectric technologies remains limited. Current applications are largely restricted to specialized fields such as radioisotope thermoelectric generators (RTGs) used in space exploration and small-scale electronic cooling devices. One of the main challenges is the need for materials that combine high efficiency with low cost, environmental stability, and abundance of constituent elements. Consequently, the search for new thermoelectric materials composed of Earth-abundant elements has become an important research direction in condensed matter physics and materials science.

1.2 The Thermoelectric Figure of Merit

The performance of a thermoelectric material is characterized by the dimensionless figure of merit zT, defined as [1]

$$zT = \frac{\sigma S^2 T}{\kappa_e + \kappa_l}, \tag{1.1}$$

where σ represents the electrical conductivity, S is the Seebeck coefficient, T is the absolute temperature, and κ_e and κ_l denote the electronic and lattice contributions to the thermal conductivity, respectively.

Achieving high thermoelectric performance is challenging because these parameters are strongly interdependent. For instance, increasing the Seebeck coefficient generally leads to a reduction in electrical conductivity. Similarly, the Wiedemann–Franz law links electrical conductivity to electronic thermal conductivity, meaning that improvements in electrical transport can simultaneously increase heat conduction.

Fortunately, the lattice thermal conductivity κ_l can often be reduced independently of the electronic transport properties. Materials with intrinsically low lattice thermal conductivity are therefore highly desirable for thermoelectric applications. However, κ_l cannot be reduced indefinitely and is ultimately bounded by the amorphous limit of the material.

Several strategies have been proposed to enhance thermoelectric performance, including band engineering, valley degeneracy, resonant doping, and nanostructuring. These approaches primarily aim to improve the electronic power factor while maintaining low thermal conductivity.

Layered materials have attracted particular interest due to their unique structural and electronic properties [2]. As a representative layered van der Waals material, germanium disulfide (GeS_2) exhibits complex lattice dynamics and strong anharmonicity, making it an excellent candidate for studying computational approaches to thermal transport.

1.3 Computational Challenges and Machine Learning

A major computational challenge in predicting lattice thermal conductivity lies in accurately determining the anharmonic third-order interatomic force constants (IFCs). Within the framework of first-principles calculations, these quantities are typically obtained using Density Functional Theory combined with finite-displacement approaches and the solution of the phonon Boltzmann Transport Equation.

However, such calculations are extremely demanding computationally. For complex materials and large supercells, the required simulations can consume hundreds of thousands of CPU hours on high-performance computing clusters.

To address this limitation, machine learning techniques have recently emerged as powerful tools for accelerating atomistic simulations. By training machine learning interatomic potentials on a carefully selected set of first-principles calculations, it becomes possible to reproduce DFT-level accuracy at a fraction of the computational cost.

In this thesis, machine learning models based on the PANNA framework are trained to predict atomic forces and energies for GeS_2. These models are then used to accelerate the generation of anharmonic force constants required for thermal conductivity calculations.

1.4 Thesis Objectives

The primary objective of this thesis is to evaluate the reliability and sensitivity of machine-learning-based predictions of lattice thermal conductivity for GeS_2. Rather than relying solely on conventional static dataset splits, this work investigates the fundamental data requirements necessary for accurately capturing anharmonic lattice dynamics.

The main goals of this study are summarized as follows:

1. **Sensitivity analysis via controlled perturbations:** To compute lattice thermal conductivity from first-principles data and systematically introduce Gaussian noise into the atomic forces. This analysis quantifies the stability of the Boltzmann Transport Equation solution and establishes a strict force accuracy threshold in the μeV/Å range required for reliable ML-based predictions.

2. **Identification of the "harmonic trap":** To investigate the limitations of training machine learning potentials exclusively on small harmonic displacements. This analysis reveals the importance of sampling strongly distorted configurations in order to accurately capture anharmonic lattice dynamics.

3. **Active learning for dataset construction:** To implement a Query-by-Committee active learning workflow that iteratively selects the most informative configurations from both finite-displacement calculations and *ab initio* molecular dynamics simulations. This strategy enables efficient exploration of the potential energy surface while minimizing the number of required DFT calculations.

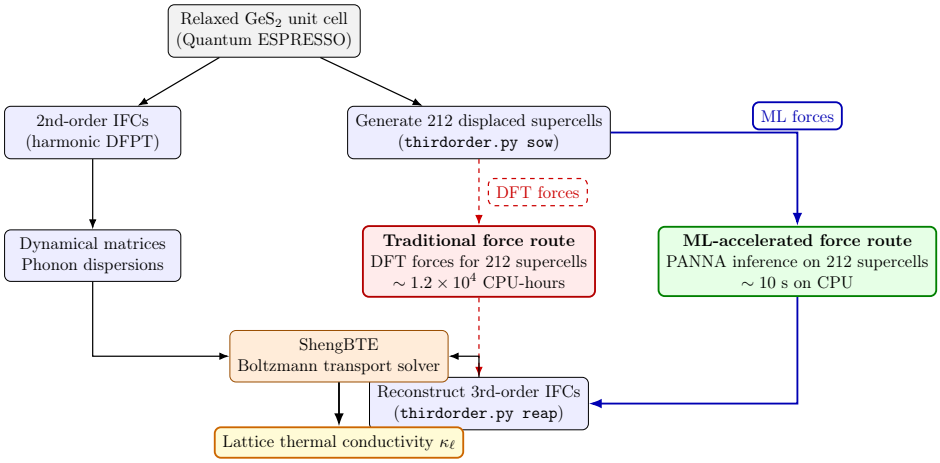

Figure 1.1: Workflow for calculating lattice thermal conductivity κ_ℓ of GeS_2. The highly expensive traditional DFT force evaluations on 212 displaced supercells (red dashed path) are replaced by fast `PANNA` machine-learning force inference (solid blue path), reducing the dominant computational bottleneck by several orders of magnitude.

Chapter 2

Theoretical Background

2.1 First-Principles Calculations

The prediction of material properties from first principles—based solely on fundamental physical laws and without empirical fitting parameters—requires solving the many-electron Schrödinger equation for a system of interacting nuclei and electrons.[1] Within this *ab initio* paradigm, all observables are, in principle, determined uniquely by the atomic numbers and positions of the constituent atoms.[1] For the present work, this means that the electronic, vibrational, and thermal transport properties of GeS_2 can be obtained starting only from its crystal structure and chemical composition, provided that suitable approximations are adopted to render the many-body problem tractable.

2.2 The Many-Body Problem

For any solid-state system, the central task is to find an approximate solution of the non-relativistic, time-independent Schrödinger equation for the full electron–nuclear Hamiltonian:[2]

$$\hat{H}\,\Psi(\vec{r},\vec{R}) = E\,\Psi(\vec{r},\vec{R})\,, \tag{2.1}$$

where $\vec{r}$ collectively denotes all electronic coordinates, $\vec{R}$ all nuclear coordinates, and Ψ is the many-body wavefunction. The complexity of this problem arises from the fact that every charged particle interacts with every other via long-range Coulomb forces, leading to a high-dimensional, strongly correlated system.

The full Hamiltonian can be written as:[3]

$$\hat{H} = T^{\text{elec}}(\vec{r}) + T^{\text{nucl}}(\vec{R}) + V^{\text{nucl-elec}}(\vec{R},\vec{r}) + V^{\text{elec}}(\vec{r}) + V^{\text{nucl}}(\vec{R})\,. \tag{2.2}$$

Each term has the following explicit form:

1. **Electronic kinetic energy**

$$T^{\text{elec}} = -\frac{\hbar^2}{2m_e}\sum_i \nabla_i^2 \tag{2.3}$$

   where m_e is the electron mass and the sum runs over all electrons.

2. **Nuclear kinetic energy**

$$T^{\text{nucl}} = -\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 \tag{2.4}$$

   where M_I is the mass of nucleus I.

3. **Electron–nuclear interaction**

$$V^{\text{nucl-elec}}(\vec{R},\vec{r}) = -\sum_i\sum_I \frac{Z_I e^2}{|\vec{R}_I - \vec{r}_i|} \tag{2.5}$$

   with Z_I the atomic number of nucleus I.

4. **Electron–electron interaction**

$$V^{\text{elec}}(\vec{r}) = \sum_i\sum_{j<i} \frac{e^2}{|\vec{r}_i - \vec{r}_j|} \tag{2.6}$$

5. **Nuclear–nuclear interaction**

$$V^{\text{nucl}}(\vec{R}) = \sum_I\sum_{J<I} \frac{Z_I Z_J e^2}{|\vec{R}_I - \vec{R}_J|}\,. \tag{2.7}$$

Directly solving this many-body problem is impossible for realistic solids. Therefore, a hierarchy of physically motivated approximations is introduced to decouple electronic and nuclear motion and to simplify the treatment of electron–electron interactions.

2.3 Approximations to the Many-Body Problem

2.3.1 Born–Oppenheimer Approximation

The Born–Oppenheimer (BO) approximation provides the first key simplification by exploiting the large mass ratio between nuclei and electrons

($M_{\text{ion}} \gg m_e$).[4] Because nuclei move much more slowly than electrons, the electronic subsystem can be assumed to adjust quasi-instantaneously to any change in nuclear positions. This allows one to treat the nuclei as fixed classical parameters in the electronic Hamiltonian and neglect the nuclear kinetic energy T^{nucl} in the electronic problem.

Within this approximation, one first solves the electronic Schrödinger equation for electrons moving in the external potential generated by a static nuclear configuration. The resulting electronic ground-state energy then acts as an effective potential energy surface for the nuclear motion. The BO approximation is remarkably accurate for most condensed-phase systems, although it can break down in situations where multiple electronic states are nearly degenerate and non-adiabatic couplings become important (e.g., conical intersections or strong electron–phonon coupling).[5]

2.3.2 Hartree and Hartree–Fock Approximations

Even after separating electronic and nuclear motion, the electronic many-body problem remains highly non-trivial due to electron–electron interactions. In the Hartree approximation, the total multi-electron wavefunction is approximated as a simple product of single-particle orbitals, neglecting the explicit antisymmetry required by the Pauli principle.[6] Each electron moves in an average effective potential generated by the nuclei and the mean field of all other electrons:

$$\left(-\frac{\nabla^2}{2} + V_{\text{ext}}(\vec{r}) + V_{\text{additional}}(\vec{r})\right) \phi_i(\vec{r}) = \epsilon_i \, \phi_i(\vec{r}) \, . \tag{2.8}$$

Here, $V_{\text{ext}}(\vec{r})$ is the external potential due to the nuclei, and $V_{\text{additional}}(\vec{r})$ is a self-consistent mean-field term arising from the electron–electron interaction.

While conceptually important, the Hartree method suffers from a critical drawback: the product wavefunction does not satisfy antisymmetry under particle exchange, and thus violates the Pauli exclusion principle.[7] This limitation is remedied in the Hartree–Fock (HF) method by introducing Slater determinants to enforce antisymmetry and an explicit exchange term. However, both Hartree and HF remain computationally demanding for periodic solids and do not fully capture correlation effects, motivating the development of Density Functional Theory.[1]

2.4 Density Functional Theory (DFT)

Density Functional Theory (DFT) reformulates the many-electron problem in terms of the ground-state electron density $\rho(\vec{r})$ rather than the many-body

wavefunction.[1] Since $\rho(\vec{r})$ depends only on three spatial coordinates, this leads to a dramatic reduction in complexity compared to working directly with the full N-electron wavefunction.

2.4.1 Hohenberg–Kohn Theorems

The foundation of DFT is provided by the two Hohenberg–Kohn (HK) theorems.[8]

1. **First HK Theorem:** For a system of interacting electrons in an external potential $V_{\text{ext}}(\vec{r})$, the ground-state electron density $\rho_0(\vec{r})$ uniquely determines $V_{\text{ext}}(\vec{r})$ up to an additive constant. Consequently, $\rho_0(\vec{r})$ uniquely determines the full many-body Hamiltonian and all ground-state properties of the system.

2. **Second HK Theorem:** There exists a universal energy functional $E[\rho]$ such that, for any trial density $\rho(\vec{r})$ that is N-representable,

$$E[\rho] \geq E[\rho_0] = E_0\,, \tag{2.9}$$

   where E_0 is the exact ground-state energy. In practice, this establishes a variational principle in terms of the density: the true ground-state density minimizes $E[\rho]$.

These theorems guarantee that, in principle, all ground-state properties of an interacting electron system can be obtained by minimizing an energy functional of the electron density alone, without explicit reference to the many-body wavefunction.[8]

2.4.2 Kohn–Sham Formalism

To make DFT computationally practical, Kohn and Sham introduced an auxiliary system of non-interacting electrons that reproduces the exact interacting ground-state density.[9] The exact ground-state energy functional can be decomposed as

$$E[\rho] = T_s[\rho] + E_{\text{H}}[\rho] + E_{\text{XC}}[\rho] + \int V_{\text{ext}}(\vec{r})\,\rho(\vec{r})\,d\vec{r}\,, \tag{2.10}$$

where $T_s[\rho]$ is the kinetic energy of the non-interacting reference system, $E_{\text{H}}[\rho]$ is the classical Hartree electrostatic energy, and $E_{\text{XC}}[\rho]$ is the exchange–correlation (XC) energy functional containing all many-body effects beyond the simple mean field.[1]

Minimization of $E[\rho]$ with respect to ρ under the constraint of fixed particle number leads to the Kohn–Sham equations for a set of single-particle orbitals $\{\psi_i(\vec{r})\}$:[9]

$$\epsilon_i\,\psi_i(\vec{r}) = \left(-\frac{\nabla^2}{2} + V_{\mathrm{H}}[\rho](\vec{r}) + V_{\mathrm{XC}}[\rho](\vec{r}) + V_{\mathrm{ext}}(\vec{r})\right)\psi_i(\vec{r})\,, \tag{2.11}$$

with the density given by

$$\rho(\vec{r}) = \sum_i |\psi_i(\vec{r})|^2\,. \tag{2.12}$$

Here, V_{H} is the Hartree potential and $V_{\mathrm{XC}} = \delta E_{\mathrm{XC}}[\rho]/\delta\rho$ is the exchange–correlation potential, which accounts for both Fermi exchange and dynamical electron–electron correlations.[10] In practice, the accuracy of DFT is governed by the choice of approximate XC functional (e.g., LDA, GGA, hybrid), while its efficiency makes it the standard tool for first-principles calculations in materials science.[1] For layered GeS_2, semi-empirical van der Waals corrections such as Grimme-D2 are often employed on top of GGA functionals to capture interlayer interactions consistently with the rest of this work.

2.5 Lattice Dynamics and Phonons

In a crystalline solid such as GeS_2, atoms occupy periodic lattice sites and vibrate around their equilibrium positions. Small displacements give rise to collective vibrational modes that propagate through the crystal as quantized lattice waves known as phonons.[11] At finite temperature, phonons play a central role in determining thermodynamic properties, heat capacity, and thermal transport.[12]

Within the harmonic approximation, the potential energy is expanded to second order in the atomic displacements, and the corresponding dynamical matrix yields phonon frequencies and eigenvectors throughout the Brillouin zone.[13] The absence of imaginary phonon frequencies is a necessary condition for dynamical stability of the crystal.[11]

2.5.1 Linear Response and DFPT

Density Functional Perturbation Theory (DFPT) provides a variational linear-response framework to compute phonon frequencies and eigenvectors directly in reciprocal space for insulators and metals alike.[13, 14] In DFPT,

the second derivatives of the total energy with respect to atomic displacements define the (harmonic) interatomic force constants (IFCs), from which one constructs the dynamical matrix:

$$D^{IJ}_{\alpha\beta}(\vec{q}) = \frac{1}{\sqrt{M_I M_J}} \sum_{\vec{R}} \Phi^{IJ}_{\alpha\beta}(0, \vec{R})\, e^{i\vec{q}\cdot\vec{R}}\,, \tag{2.13}$$

where $\Phi^{IJ}_{\alpha\beta}$ are the real-space IFCs, M_I and M_J are atomic masses, and $\vec{q}$ is a phonon wavevector. Diagonalization of $D(\vec{q})$ yields phonon frequencies $\omega(\vec{q}, p)$ and eigenvectors for each branch p.[13]

2.5.2 Anharmonicity and the Boltzmann Transport Equation

To compute the lattice thermal conductivity κ_l, it is essential to go beyond the harmonic approximation and include phonon–phonon scattering processes arising from anharmonic terms in the potential energy.[12] The leading contribution is captured by the third-order IFCs, which describe three-phonon interactions.

Within the relaxation-time approximation (RTA), the lattice thermal conductivity along a Cartesian direction α can be expressed as

$$\kappa_\alpha = \frac{1}{V} \sum_{\vec{q},p} C_V(\vec{q}, p)\, v_\alpha(\vec{q}, p)^2\, \tau(\vec{q}, p)\,, \tag{2.14}$$

where V is the crystal volume, $C_V(\vec{q}, p)$ is the mode-resolved heat capacity, $v_\alpha(\vec{q}, p)$ is the phonon group velocity component, and $\tau(\vec{q}, p)$ is the phonon lifetime.[12] The lifetimes are obtained by constructing a scattering matrix from the third-order anharmonic IFCs and solving the linearized Boltzmann Transport Equation for phonons.[15]

In this work, the anharmonic IFCs are computed using the real-space finite-displacement method and then interfaced with Boltzmann Transport Equation solvers such as *ShengBTE* and *thirdorder.py*, which allow an accurate and fully first-principles evaluation of κ_l in GeS_2.[15]

2.6 The Machine Learning Approach to IFCs

2.6.1 The Computational Bottleneck

Obtaining third-order IFCs via the real-space supercell approach requires an irreducible set of atomic displacements. For complex systems such as GeS_2,

this can necessitate hundreds of single-point DFT force calculations on large supercells, creating a significant computational bottleneck when combined with dense q-meshes in ShengBTE.[15]

Chapter 3

Computational Details and *Ab Initio* Results

3.1 First-Principles Framework

The ground-state structural and electronic properties of GeS_2 were investigated within Density Functional Theory (DFT), as implemented in the QUANTUM ESPRESSO suite of codes.[16, 17] All calculations were performed using periodic boundary conditions and a plane-wave basis set.[1] To obtain a reliable equilibrium structure suitable for subsequent phonon and anharmonic force constant calculations, we carried out variable-cell relaxations (`vc-relax`) in which both the lattice vectors and the internal atomic coordinates were optimized simultaneously.

3.1.1 Exchange–Correlation and Pseudopotentials

Electronic exchange–correlation effects were described using the Perdew–Burke–Ernzerhof (PBE) functional within the Generalized Gradient Approximation (GGA).[18, 19] The interaction between valence electrons and ionic cores was treated using norm-conserving and ultrasoft pseudopotentials from the PSLibrary,[20] specifically the `pbe-n-rrkjus_psl.1.0.0.UPF` datasets for both Ge and S. These pseudopotentials include scalar-relativistic effects where appropriate and have been extensively validated for chalcogenide systems and solid-state calculations.[20]

3.1.2 Plane-Wave Basis and Cutoff Convergence

The Kohn–Sham orbitals were expanded in a plane-wave basis set with a kinetic energy cutoff `ecutwfc` of **65 Ry**. To represent the charge density and

potentials, we adopted an auxiliary cutoff `ecutrho` of **520 Ry**, corresponding to a standard 1:8 ratio that is well suited for ultrasoft pseudopotentials.[21] These values were selected on the basis of preliminary convergence tests, which confirmed that total energies, equilibrium lattice parameters, and forces are converged within a few meV per formula unit and below the target force thresholds for structural optimization.

3.1.3 Brillouin Zone Sampling and Anisotropy

Brillouin zone integrations were performed using a Monkhorst–Pack k-point mesh of $9 \times 9 \times 3$.[22] The dense sampling in the in-plane directions (9×9 in the xy-plane) captures the strongly anisotropic electronic structure associated with the layered nature of GeS_2, while the coarser sampling of 3 points along k_z exploits the larger interlayer spacing.[23] This anisotropic grid provides a good compromise between accuracy and computational cost, and ensures well-converged total energies and forces for subsequent lattice-dynamical calculations.[24]

3.1.4 Van der Waals Corrections

Because GeS_2 is a layered van der Waals material, long-range dispersion interactions between adjacent layers play a non-negligible role in determining the out-of-plane lattice constant and phonon spectrum.[25] Standard semi-local GGA functionals such as PBE systematically underestimate these dispersion forces.[26] To remedy this, we employed the semi-empirical Grimme DFT-D2 correction,[27] as implemented in Quantum ESPRESSO. The DFT-D2 method has been widely validated for layered chalcogenides and significantly improves the agreement of the optimized c lattice parameter with experimental data.[28]

3.1.5 Relaxation Protocol and Convergence Criteria

Structural optimizations were carried out using the Broyden–Fletcher–Goldfarb–Shanno (BFGS) quasi-Newton algorithm for both ionic and cell degrees of freedom.[29] The self-consistent field (SCF) convergence threshold on the total energy was set to 10^{-10} **Ry**, ensuring highly converged charge densities before each ionic update.[17] Geometry optimization was deemed converged when the change in total energy between successive ionic steps was smaller than 10^{-5} Ry and the maximum residual force on any atom was less than 10^{-4} Ry/bohr. These stringent criteria guarantee that the final structure

is sufficiently relaxed for accurate extraction of harmonic and anharmonic interatomic force constants.[13]

The input files used for these calculations (`GeS2.scf.in` for the unit cell and `GeS2.sc.scf.in` for supercell forces) are provided in the Annexure. These settings were used for all 212 symmetry-distinct displaced supercells.

Chapter 4

Calculation of Anharmonic Force Constants

4.1 Introduction to the Finite Displacement Method

The accurate determination of lattice thermal conductivity (κ_l) requires knowledge of the anharmonic interatomic force constants (IFCs), which govern phonon–phonon scattering processes in crystalline solids.[15, 13] While harmonic IFCs (second-order derivatives of the total energy) can be efficiently computed using Density Functional Perturbation Theory (DFPT),[13] the calculation of third-order IFCs remains computationally demanding and is typically performed using the finite-displacement method in real space.[15]

In this work, we employ the `thirdorder.py` package,[30] an open-source Python-based tool designed to interface seamlessly with *ShengBTE*[15] and *almaBTE*[30] for the efficient generation of third-order force constants. The finite-displacement approach involves systematically displacing atoms from their equilibrium positions and computing the resulting forces on all other atoms via DFT. By fitting these forces to a Taylor expansion of the potential energy surface, one can extract the anharmonic IFCs.[15]

The workflow consists of two primary stages:

1. **Configuration generation ("sow"):** The `thirdorder.py` script identifies a symmetry-reduced set of atomic displacements by exploiting the space-group symmetry of the GeS_2 lattice using the `spglib` library.[31] This symmetry reduction dramatically decreases the number of required supercell calculations—often by more than an order of magnitude—making the approach computationally tractable. The

displaced structures are then exported as input files compatible with the QUANTUM ESPRESSO `pw.x` engine.[16]

2. **Force reconstruction ("reap"):** Once all DFT single-point calculations are complete, the script parses the output files to extract the force vectors acting on each atom in every displaced configuration. The third-order IFCs are then reconstructed by solving the overdetermined system of linear equations that relate atomic displacements $\mathbf{u}$ to the induced forces $\mathbf{F}$:[15]

$$\Phi^{ijk}_{\alpha\beta\gamma} = -\frac{\partial^3 E}{\partial u_{i\alpha}\,\partial u_{j\beta}\,\partial u_{k\gamma}} \approx \frac{\Delta F_{i\alpha}}{\Delta u_{j\beta}\,\Delta u_{k\gamma}}\,, \tag{4.1}$$

where i, j, k label atoms, α, β, γ denote Cartesian components, and E is the total energy. The output is formatted into the `FORCE_CONSTANTS_3RD` file required by BTE solvers.

4.2 Software Architecture and Compilation

The `thirdorder.py` suite is implemented as a hybrid Python/Cython framework.[30] The high-level workflow logic and I/O routines are written in Python for flexibility and ease of use, while the computationally intensive symmetry-reduction algorithms are implemented in a Cython-compiled core module (`thirdorder_core`). This design achieves near-C performance for the bottleneck operations while retaining the convenience of a scripted interface.

To deploy the code on the Leonardo HPC cluster, the package was compiled from source against the system C compiler (`gcc`) and the Python 3.x development headers. The build process generates a shared-object library (`thirdorder_core.so`), which must be placed in the Python module search path alongside the main interface script `thirdorder_espresso.py`. The `spglib` library[31] is a critical dependency, as it handles the identification of crystal symmetry operations and equivalent atoms, enabling efficient reduction of the displacement set.

4.3 Implementation for the GeS_2 System

The anharmonic force constant calculation for GeS_2 requires two input files (provided in full in Appendix 7.6.3):

1. **Optimized unit cell (`GeS2.scf.in`):** A QUANTUM ESPRESSO input file containing the fully relaxed atomic positions and lattice

vectors obtained from the variable-cell relaxation (`vc-relax`) described in Chapter 3. This file defines the reference equilibrium geometry from which all displacements are measured.

2. **Supercell template (`GeS2.sc.scf.in`):** A modified input file specifying the DFT parameters—pseudopotentials, plane-wave cutoffs, k-point sampling, and convergence thresholds—to be used for the force evaluations on the displaced supercells.

4.3.1 Computational Strategy and Sampling

Computing third-order IFCs requires constructing a supercell large enough to capture the spatial range of anharmonic interactions while avoiding spurious interactions across periodic images. For the layered GeS_2 system, a $3 \times 3 \times 1$ supercell expansion was adopted. This corresponds to a multiplication of the lattice vectors in the basal plane (**a** and **b**), while the large van der Waals spacing along the **c**-axis allows for sufficient isolation without further expansion.

To maintain consistency with the unit cell calculations, the sampling of the Brillouin zone was scaled inversely to the supercell size. The unit cell was sampled with a $9 \times 9 \times 3$ mesh (as shown in `GeS2.scf.in`); consequently, the $3 \times 3 \times 1$ supercell was sampled using a $3 \times 3 \times 1$ Monkhorst-Pack grid. This ensures that the effective density of k-points in reciprocal space remains constant between the unit cell and supercell calculations, preventing numerical artifacts in the force constants.[22]

To ensure high accuracy and numerical stability of the extracted forces, the following DFT parameters were adopted for all supercell calculations (consistent with Annexure 7.6.3):

- **Plane-wave cutoffs:** Kinetic energy cutoffs of `ecutwfc` = 65 Ry and `ecutrho` = 520 Ry were utilized. These values provide a robust balance between accuracy and computational efficiency for the specific pseudopotentials employed.

- **SCF convergence threshold:** A stringent threshold of 10^{-10} Ry was imposed on the total energy convergence (`conv_thr = 1d-10`). This high precision ensures that the residual forces are converged to better than 10^{-5} Ry/bohr, which is essential for accurate phonon lifetimes in *ShengBTE*[15] and for the training of the PANNA neural network potential.

- **Starting wavefunctions:** A random initialization (`startingwfc = 'random'`) was used for the electronic wavefunctions, coupled with a mixing beta of 0.5 to ensure stable convergence in the larger supercell.
- **Van der Waals correction:** The Grimme DFT-D2 dispersion correction[27] was consistently applied to all supercell calculations to maintain consistency with the relaxed reference structure.

4.3.2 The "Sow" Operation: Displacement Generation

The first step in the finite-displacement workflow is to generate the set of symmetry-distinct atomic displacements. For the GeS_2 system, this is achieved by invoking the `thirdorder.py` script in "sow" mode:

```
python3 thirdorder_espresso.py GeS2.scf.in sow 3 3 1 -3 GeS2.sc.scf.in
```

Here, the arguments have the following meanings:

- `3 3 1`: Supercell dimensions along the three lattice vectors (**a**, **b**, **c**). This choice respects the layered nature of the material, expanding only within the covalently bonded planes.
- `-3`: Cutoff parameter specifying the range of third-order interactions to the third nearest-neighbor shell. The negative sign indicates that the cutoff is interpreted in units of neighbor shells rather than an absolute distance.[30]
- `GeS2.sc.scf.in`: Template file defining the DFT calculation parameters for each displaced configuration.

The script uses `spglib`[31] to identify all symmetry operations of the GeS_2 structure, then constructs a minimal set of independent displacements such that all third-order IFCs within the specified cutoff can be uniquely determined. For the present system, this procedure generated 212 unique displacement configurations, significantly fewer than the several thousand that would be required without symmetry reduction. Each configuration is written as a separate QUANTUM ESPRESSO input file (`3RD.DISP_XXX.in`).

4.3.3 DFT Force Calculations

The 212 displaced supercells were submitted to the Leonardo HPC cluster as independent single-point SCF calculations using SLURM array jobs. To maximize computational throughput, each configuration was assigned to 4

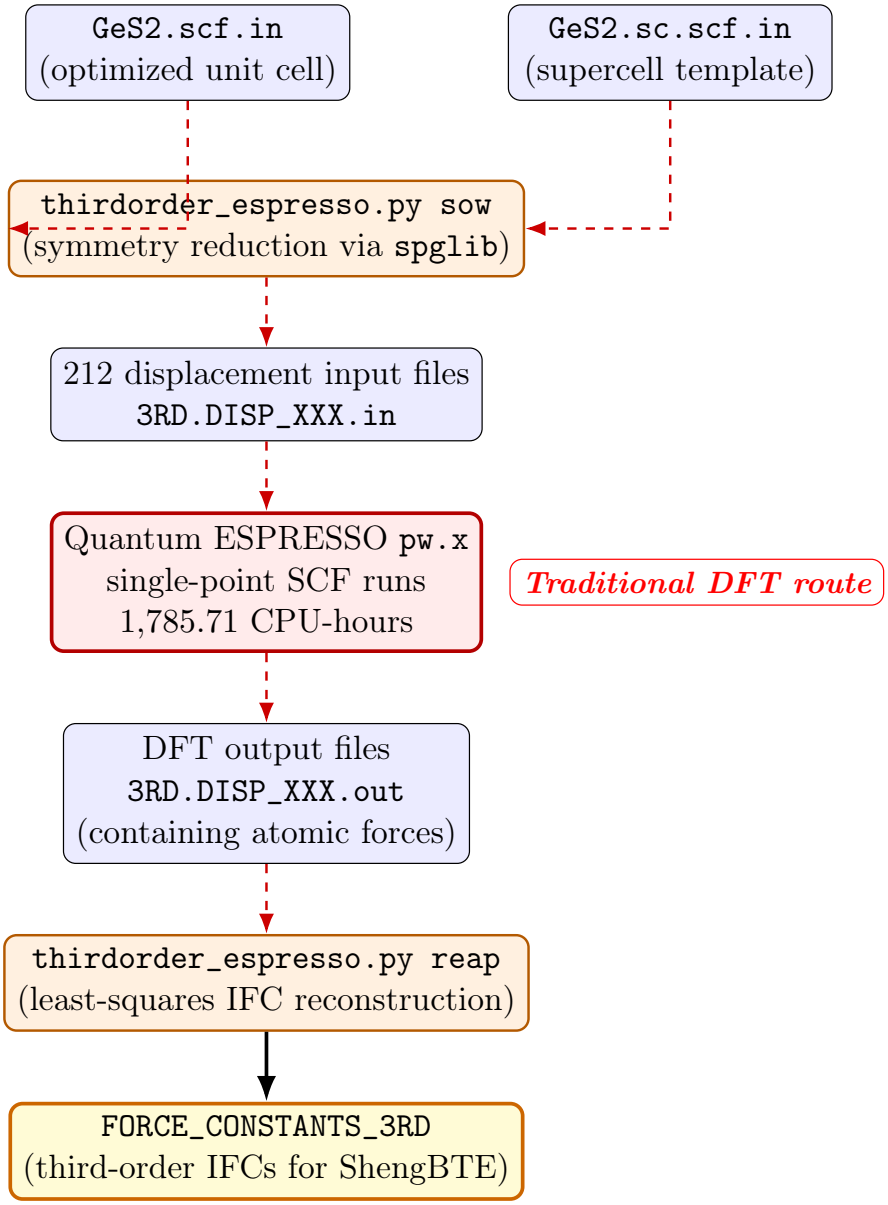

Figure 4.1: Finite-displacement workflow for extracting third-order anharmonic interatomic force constants (traditional DFT route). The `sow` step exploits crystal symmetry to generate a minimal set of displaced supercell calculations, while the `reap` step reconstructs the full third-order IFC tensor from the resulting DFT forces.

MPI tasks with 8 OpenMP threads per task (`cpus-per-task=8`), effectively allocating 32 CPU cores and 4 NVIDIA A100 GPUs per job.

The average wall time per configuration was approximately 16 minutes. By parsing the exact `WALL` times from all 212 Quantum ESPRESSO output logs, the total cumulative computational cost for this DFT stage was strictly quantified at 55.80 node-hours, which translates to $1,785.71$ CPU-core-hours. All calculations were monitored for strict electronic convergence before extracting the atomic forces.

4.3.4 The "Reap" Operation: Force Reconstruction

Once all 212 DFT calculations completed successfully, the forces were collected and processed in "reap" mode:

```
find . -name "3RD.DISP_*.out" | sort -V | \
python3 thirdorder_espresso.py GeS2.scf.in reap 3 3 1 -3
```

The `reap` operation parses each output file to extract the force vectors on all atoms, then solves the least-squares problem to reconstruct the full set of third-order IFCs $\Phi^{ijk}_{\alpha\beta\gamma}$ consistent with the calculated forces. The final result is written to the file `FORCE_CONSTANTS_3RD`, which serves as the primary input for solving the phonon Boltzmann Transport Equation in *ShengBTE*.[15]

4.3.5 HPC Performance and Scaling Analysis

To ensure optimal utilization of the Leonardo HPC cluster resources and to mathematically justify the chosen parallelization strategy, rigorous performance benchmarking was conducted for the DFT workflow using Quantum ESPRESSO. Both strong and weak scaling behaviors were evaluated.

Strong Scaling

Strong scaling measures the parallel efficiency of solving a fixed-size physical problem as the number of processing elements increases. A single $3 \times 3 \times 1$ displaced GeS_2 supercell (54 atoms) was evaluated across 1, 2, 4, and 8 NVIDIA A100 GPUs using a hybrid MPI/OpenMP parallelization scheme.

As illustrated in Figure 4.2, the calculation exhibits near-perfect linear scaling from 1 to 2 GPUs, and maintains a highly efficient speedup of $6.64\times$ when utilizing 8 GPUs across two compute nodes. While 8 GPUs yielded the fastest absolute wall time, assigning 4 GPUs (1 complete node) per configuration was identified as the optimal balance between throughput

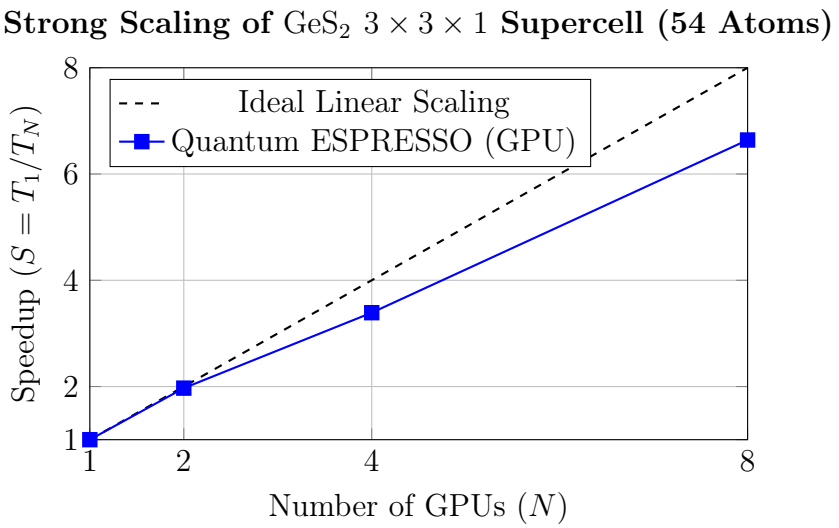

Figure 4.2: Strong scaling performance of a single point DFT force evaluation on the Leonardo cluster. Efficiency remains exceptionally high (near-linear) up to 8 GPUs. The minor deviation from the ideal line at 4 and 8 GPUs is attributed to inter-process communication overhead for the 3D FFT grids.

efficiency and total cluster allocation limits for the high-throughput SLURM array.

Weak Scaling

Weak scaling evaluates the system's ability to maintain a constant solution time when both the physical problem size and the computational resources are increased proportionally. The benchmark was constructed by scaling the GeS_2 system from a 6-atom unit cell (1 GPU), to a 24-atom supercell (4 GPUs), and to the production 54-atom supercell (8 GPUs).

The weak scaling behavior, depicted in Figure 4.3, highlights the fundamental scaling physics of standard DFT implementations, which scale as $\mathcal{O}(N_k \cdot N^3)$ where N_k is the number of irreducible k-points and N is the number of electrons. The 6-atom system required a dense $9 \times 9 \times 3$ Monkhorst-Pack mesh, while the 24-atom and 54-atom systems required proportionally reduced meshes ($5 \times 5 \times 3$ and $3 \times 3 \times 1$, respectively) to maintain constant phase-space density.

The initial wall time increase from the 6-atom to the 24-atom geometry occurs because the cubic scaling penalty of the atoms outpaces the linear hardware increase. However, the subsequent drop in wall time for the 54-atom system running on 8 GPUs is a direct consequence of the massive reduction in

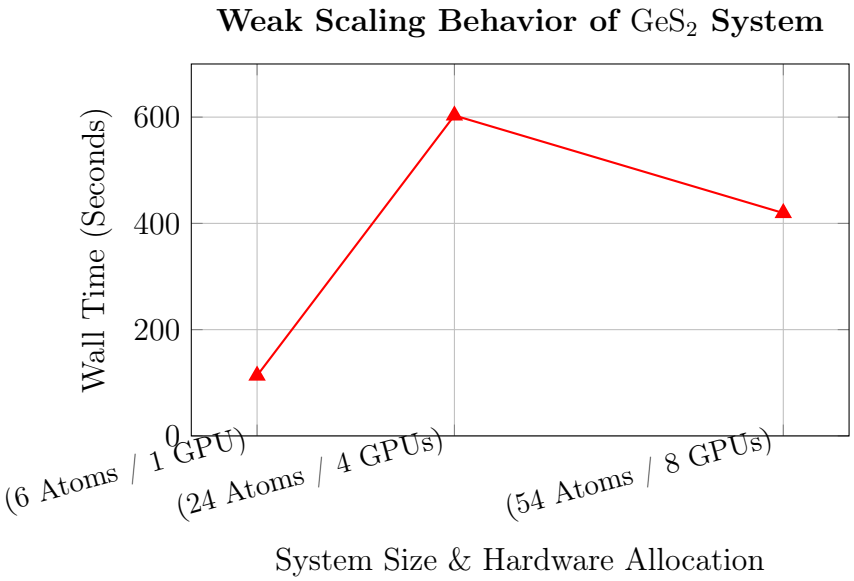

Figure 4.3: Weak scaling performance. The wall time is governed by the combined computational cost of real-space size (N^3) and reciprocal-space density (N_k). The optimal balancing of k-points in the 54-atom supercell ($3 \times 3 \times 1$) leads to a lower absolute wall time than the intermediate 24-atom geometry.

required k-points, demonstrating the high computational efficiency achieved for the final production geometry.

Chapter 5

Introduction to PANNA

To overcome the prohibitive computational cost associated with third-order interatomic force constant (IFC) calculations (e.g., 1,785.71 CPU-core-hours for GeS_2, as detailed in prior chapters), this research employs the Properties from Artificial Neural Network Architectures (`PANNA`) framework [32]. `PANNA` is an open-source software package designed to train and validate deep neural network potentials for atomistic simulations by extracting structural and force information directly from first-principles outputs.

Unlike traditional empirical potentials, `PANNA` learns the local atomic environment through a sophisticated descriptor-based approach. By acting as a high-fidelity surrogate model, it reproduces Density Functional Theory (DFT) level accuracy at a fraction of the computational cost using TensorFlow-based neural networks. This efficiency enables the rapid extraction of the anharmonic force constants strictly required for solving the phonon Boltzmann Transport Equation (BTE).

5.1 Computational Methodology and Atomic Descriptors

5.1.1 Data Generation: The Harmonic vs. Anharmonic Regimes

The foundational training dataset comprised 212 displaced $3 \times 3 \times 1$ GeS_2 supercells. These displacements were systematically generated using the `thirdorder.py` script from `ShengBTE` [15], which applies symmetry-reduced finite displacements (typically ~ 0.01 Å) to map the force constants.

However, because these systematic displacements only sample the very bottom of the potential energy well (i.e., the harmonic regime), a neural

network trained exclusively on them fails to accurately capture the steep anharmonic "walls" of the potential energy surface. To prevent the surrogate model from yielding unphysical results in high-temperature transport scenarios, *ab initio* Molecular Dynamics (AIMD) simulations were performed on a $3 \times 3 \times 1$ supercell in the canonical (NVT) ensemble at 500 K, with crystal symmetry explicitly disabled.

From the resulting thermal trajectory, 53 statistically independent structural snapshots were extracted. To obtain the exact ground-truth energies and atomic forces required for neural network training, high-precision single-point self-consistent field (SCF) calculations were performed on each of these snapshots using the QUANTUM ESPRESSO `pw.x` executable. Integrating these explicitly evaluated, high-variance configurations into the dataset provided the neural network with critical information regarding high-magnitude, anharmonic atomic interactions.

5.1.2 All-Neighbor Atomic Descriptors (Behler–Parrinello)

To convert Cartesian coordinates into a translationally and rotationally invariant format suitable for machine learning, `PANNA` employs modified Behler–Parrinello (mBP) symmetry functions [33]. The local environment of atom i is encoded into a G-vector comprising distinct radial and angular components.

The radial symmetry function captures the distance to neighboring atoms j within a specified cutoff radius R_c:

$$G_i^{\text{rad}} = \sum_{j \neq i} e^{-\eta(R_{ij} - R_s)^2} f_c(R_{ij}) \,. \tag{5.1}$$

The angular symmetry function captures the bond angles θ_{ijk} between the central atom i and neighbors j, k:

$$G_i^{\text{ang}} = 2^{1-\zeta} \sum_{j,k \neq i} (1 + \cos(\theta_{ijk} - \theta_s))^{\zeta} e^{-\eta(R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \,, \tag{5.2}$$

where $f_c(R)$ is a smooth cutoff function ensuring the descriptors decay cleanly to zero at the cutoff boundary.

For the layered GeS_2 system, these descriptors were rigorously tuned to adequately capture the complex crystalline structure and variations in Ge–S bonding:

- **Radial:** Cutoff $R_{c,\text{rad}} = 4.6$ Å, $\eta_{\text{rad}} = 16.0$ Å^{-2}, utilizing 16 Gaussian bins ($R_{sN,\text{rad}} = 16$).

- **Angular:** Cutoff $R_{c,\text{ang}} = 3.1$ Å, $\eta_{\text{ang}} = 6.0$ Å^{-2}, utilizing 4 radial bins ($R_{sN,\text{ang}} = 4$) and 8 angular bins ($\Theta_{sN} = 8$).

This precise configuration yields a highly descriptive 128-dimensional vector per atom ($g_{\text{size}} = 128$).

5.2 Implementation of the ML Data Pipeline

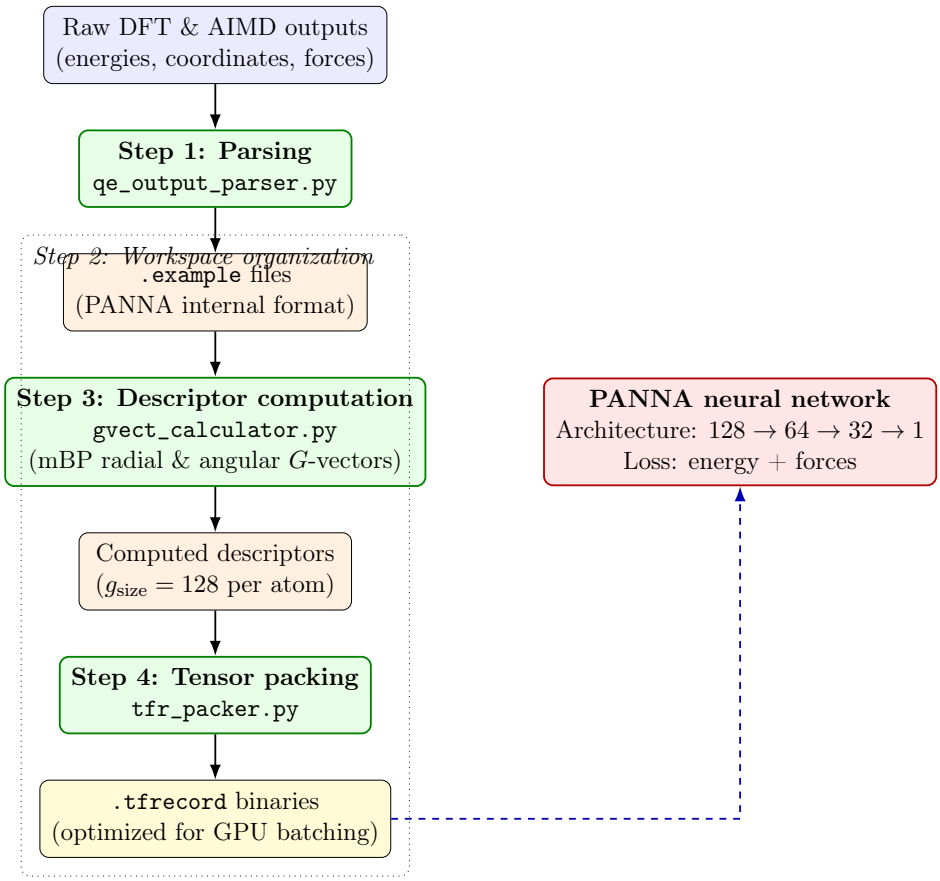

Figure 5.1: The `PANNA` data preprocessing pipeline. Raw *ab initio* data are systematically converted into high-dimensional modified Behler–Parrinello (mBP) symmetry functions and packed into `TFRecord` files for efficient neural network training.

To process the QUANTUM ESPRESSO (QE) DFT data into a format digestible by the TensorFlow neural network, a strict, four-step preprocessing pipeline was established.

Step 1: Parse QE Outputs into PANNA Structures

The QE output files (e.g., `DISP.scf_sc.out.N`) are parsed into `PANNA`'s internal `.example` format using the `qe_output_parser.py` utility:

```
python3 /path/to/panna/tools/qe_output_parser.py -i ./ -o out_sow/
```

Step 2: Workspace Organization

The parsed structures are organized into a strict directory hierarchy to separate raw inputs, computed descriptors, and packed binary records:

```
cd out_sow/
mkdir input gvector tfr
mv *.example input/
```

Step 3: Compute G-Vectors

The physical descriptors (G-vectors) are calculated for every atom using the `gvect_calculator.py` script governed by `gvector.ini`. Crucially, the `compute_derivatives = True` flag is enabled to ensure the network can train directly on atomic force gradients.

```
python3 /path/to/panna/gvect_calculator.py -c gvector/gvector.ini
```

Step 4: Pack Descriptors into TFRecords

The final preprocessing step packs the high-dimensional G-vectors into highly efficient binary TensorFlow Records (`.tfrecord`) using `tfr_packer.py` and the `tfr.ini` configuration file, allowing for high-throughput batching during GPU training.

```
python3 /path/to/panna/tfr_packer.py --config tfr.ini
```

5.3 Neural Network Architecture and Training

Within the `PANNA` framework, the total potential energy of the GeS_2 system is decomposed into a sum of localized, atomic environmental contributions:

$$E_{\text{total}} = \sum_i E_i(G_i)\,, \tag{5.3}$$

where E_i is the predicted energy contribution of atom i, generated by a sub-network uniquely mapped to its chemical species, and G_i is its local environment descriptor. Atomic forces are obtained analytically as the exact spatial gradients of the predicted total energy:

$$\mathbf{F}_i = -\frac{\partial E_{\text{total}}}{\partial \mathbf{R}_i} \,. \tag{5.4}$$

A fully connected, feed-forward deep neural network was constructed with an input layer of 128 nodes, hidden layers of 64 and 32 nodes, and a single energy output node per species (an architecture of $128 \to 64 \to 32 \to 1$) for both the Ge and S atomic species.

The loss function $\mathcal{L}$ balances total energy and atomic force errors:

$$\mathcal{L} = \frac{1}{N} \sum \left[(E_{\text{DFT}} - E_{\text{ML}})^2 + \lambda (\mathbf{F}_{\text{DFT}} - \mathbf{F}_{\text{ML}})^2 \right] \,. \tag{5.5}$$

Because lattice thermal conductivity predictions depend entirely on the third-order spatial derivatives of the energy (i.e., the numerical gradients of the forces), a high weighting factor of $\lambda = 0.3$ was strictly applied to heavily penalize force errors over absolute energy errors. The model was trained using the Adam optimizer with an initial learning rate of 0.001 and a batch size of 20.

5.4 Structural Validation and Computational Acceleration

Before investigating phonon transport, the structural integrity of the relaxed GeS_2 unit cell must be validated. The lattice parameters obtained from our QUANTUM ESPRESSO `vc-relax` simulations—employing the PBE exchange–correlation functional with Grimme-D2 van der Waals corrections—are compared with available data in Table 5.1.

Table 5.1: Comparison of calculated lattice parameters for GeS_2 with existing literature.

Source	a **(Å)**	b **(Å)**	c **(Å)**
This Work (DFT-PBE+D2)	3.501	3.501	10.992
Experimental [34]	3.469	3.469	10.974
Other Theoretical [35]	3.500	3.500	11.200

The calculated lattice constants demonstrate a slight overestimation (within 1.2%) relative to experimental findings, which is entirely consistent with standard GGA functional performance.

A central objective of this thesis is to explicitly quantify the computational acceleration achieved using the trained machine learning potential. As outlined in Table 5.2, performing rigorous *ab initio* self-consistent field calculations for the 212 displaced supercells required 55.80 cumulative node-hours on GPU-accelerated HPC nodes. In stark contrast, the fully trained `PANNA` surrogate model evaluated the precise interatomic forces for the entire configuration set in just 7.03 seconds. This translates to an extraordinary computational speed-up of more than four orders of magnitude, unequivocally validating the use of MLIPs to bypass the primary computational bottleneck of thermal transport simulations.

Table 5.2: Computational wall-time comparison between DFT and Machine Learning inference for all 212 configurations.

Method	**Hardware Allocation**	**Avg. Time per Config.**	**Total Cumulative Time**
DFT (Quantum ESPRESSO)	4× A100 GPUs, 32 Cores	$\sim$ 16 min	55.80 h
ML Training (`PANNA`)	1 Node (Local CPU/GPU)	N/A	**15.3 min**
ML Inference (`PANNA`)	1 Node (Local CPU)	**0.025 s**	**7.03 s**

5.5 Sensitivity of Lattice Thermal Conductivity to Force Noise

Lattice thermal conductivity (κ_l) depends explicitly on third-order IFCs, which are derived from minute changes in atomic forces. To establish the target accuracy threshold required for our neural network, a systematic sensitivity analysis was performed by injecting artificial Gaussian noise into the 212 real-space DFT force evaluations prior to solving the BTE.

As demonstrated in Figures 5.2 and 5.3, standard machine learning error regimes (1–20 meV/Å) severely and artificially suppress thermal transport predictions. In the BTE formulation, phonon scattering rates are proportional to the square of the 3rd-order force constants ($\Gamma \propto |V_3|^2$). Numerical noise artificially inflates these derivatives, creating "synthetic scattering" that rapidly drives the thermal conductivity toward zero. Therefore, to ensure physical validity, surrogate force models must achieve remarkable predictive precision on the order of 10–50 μeV/Å across the evaluation dataset.

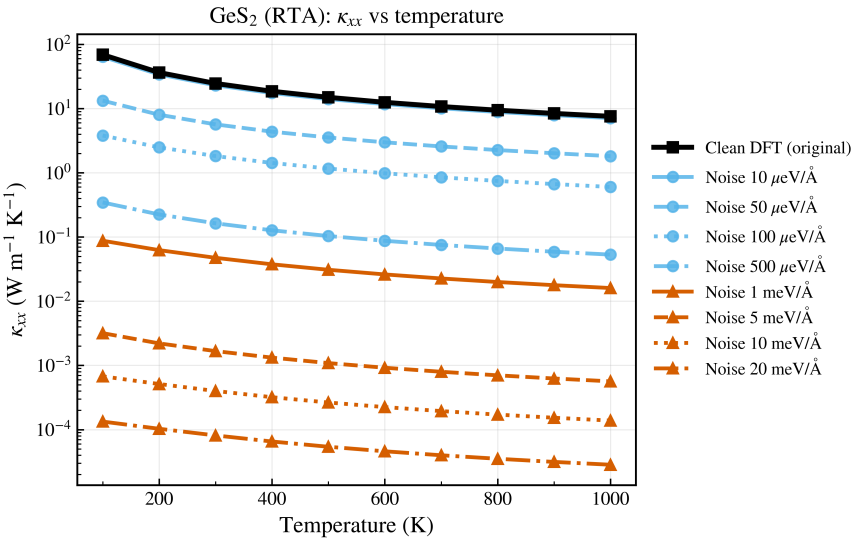

Figure 5.2: Calculated in-plane lattice thermal conductivity (κ_{xx}) of GeS_2 as a function of temperature under varying amplitudes of random Gaussian noise. Strict force accuracies of $\leq 50\,\mu$eV/Å are required to avoid artificial scattering.

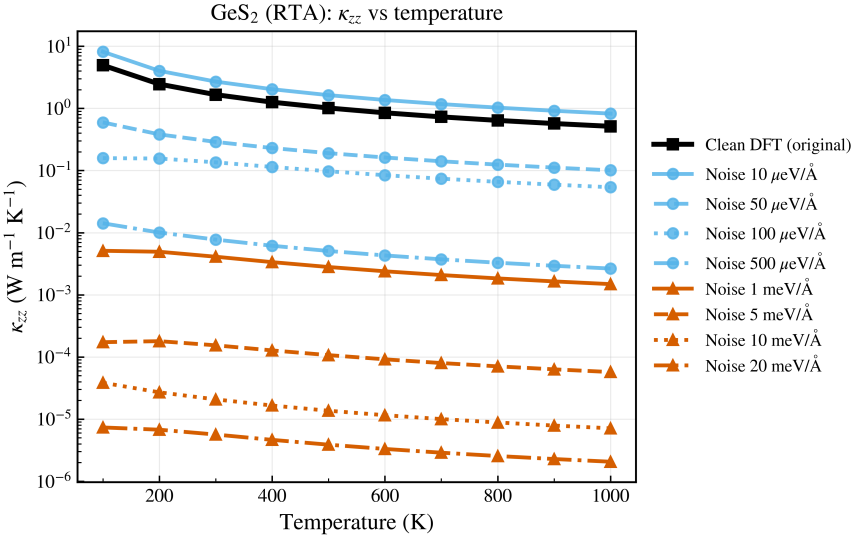

Figure 5.3: Calculated cross-plane lattice thermal conductivity (κ_{zz}) of GeS_2 under varying force noise amplitudes.

5.6 Machine Learning Model Validation

With the required accuracy bounds clearly established, multiple PANNA network iterations were trained to observe convergence and physical validity.

5.6.1 Training on the Pure DFT Dataset: The Harmonic Trap

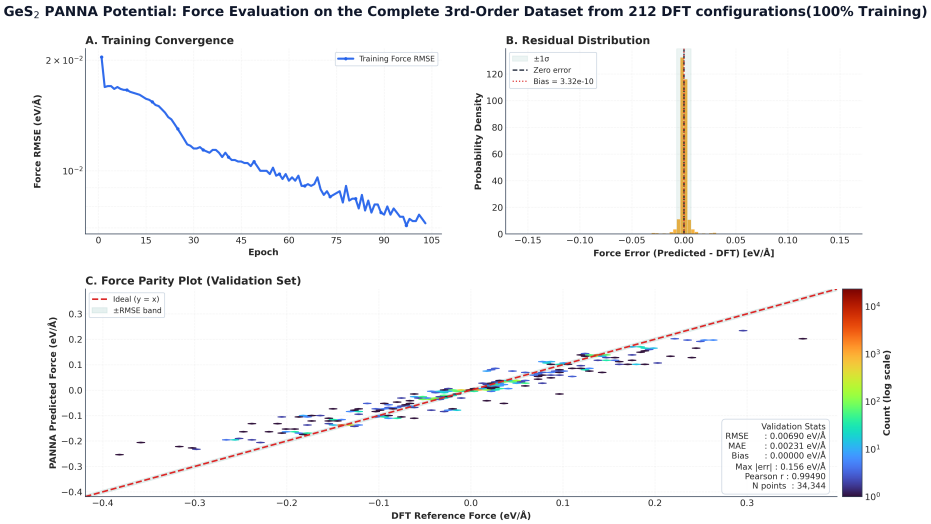

Figure 5.4: PANNA model trained on the full DFT dataset for GeS_2: (A) training convergence, (B) probability density distribution of the force residuals, and (C) log-density hexbin parity plot.

Initially, the PANNA potential was trained exclusively on the 212 systematically displaced configurations to establish a baseline. Figure 5.4 illustrates the learning dynamics when utilizing 100% of this dataset. As training progresses, the force Root Mean Square Error (RMSE) converges rapidly (Figure 5.4A). The model demonstrates excellent interpolation within this dataset, evidenced by the sharp, zero-centered probability density of the force residuals (Figure 5.4B) and the tight linear correlation in the log-density hexbin parity plot (Figure 5.4C).

To rigorously assess generalizability and rule out over-parameterization, an 80/20 train-validation split (169 training, 43 validation configurations) was employed. As shown in Figure 5.5, the network successfully maps the test set without severe degradation in the parity distribution, confirming that the model avoids gross overfitting within this specific structural regime.

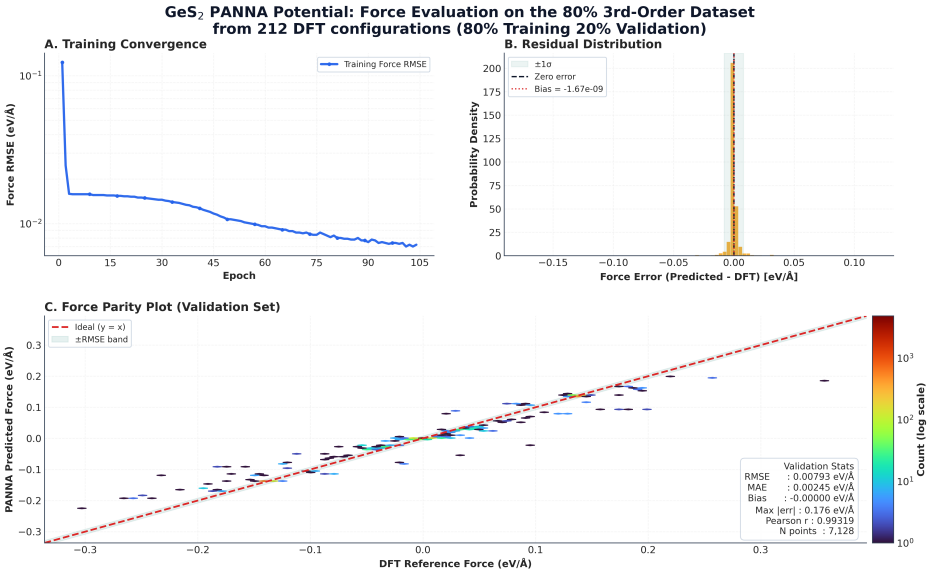

Figure 5.5: PANNA model trained on 80% of the DFT dataset and validated on the remaining 20% for GeS_2, confirming the model avoids gross overfitting within the harmonic regime.

However, despite achieving highly favorable force RMSE metrics mathematically, these pure-DFT models ultimately exhibit a severe physical limitation, effectively falling victim to the "harmonic trap." Because the 212 systematic displacements generated by `ShengBTE` represent only minor perturbations near the equilibrium crystal structure, the neural network perfectly learns the shallow bottom of the potential energy well but remains entirely blind to the anharmonic "walls." Because the network never observes high-energy, chaotic atomic structures, it cannot reliably extrapolate the steep anharmonic interactions required to accurately model temperature-dependent phonon scattering.

5.6.2 The Hybrid Approach: Augmenting with AIMD

To rectify this physical limitation and teach the network the true topology of the broader potential energy surface, the 53 high-temperature AIMD configurations were integrated into the training set. This hybrid approach firmly anchors the model with exact harmonic baseline symmetries while simultaneously supplementing it with chaotic, anharmonic snapshots.

Figures 5.6 and 5.7 present the learning behavior for this robust, combined dataset under both 100% training and an 80/20 validation split. Even with the introduction of high-variance structural noise from the AIMD trajectory,

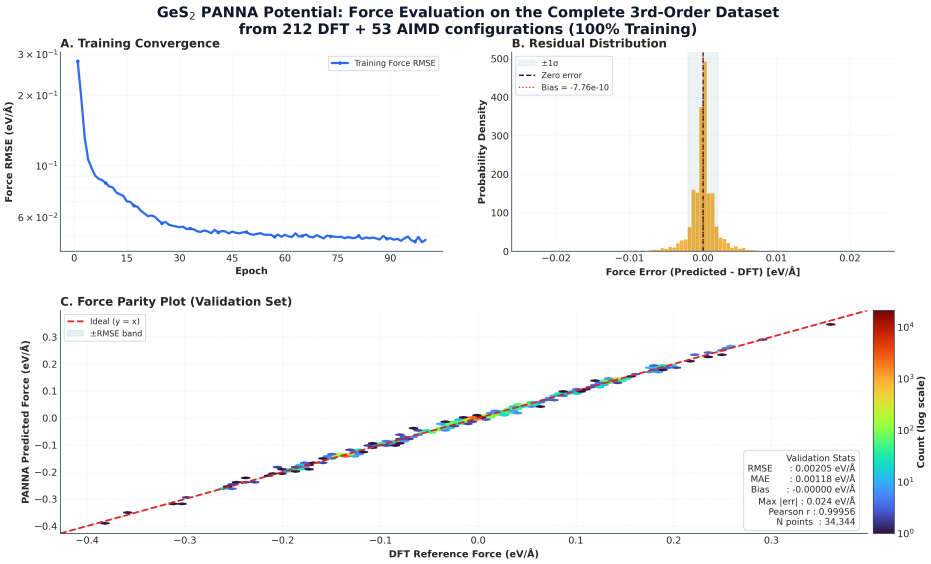

Figure 5.6: PANNA model trained on the complete combined DFT + AIMD dataset. The inclusion of AIMD snapshots allows the network to capture both harmonic baseline symmetries and steep anharmonic well-walls.

the neural network maintains steady convergence and an unbiased residual distribution.

To strictly prevent overfitting during the hybrid training phase, the optimal model was selected based on the epoch yielding the lowest validation force RMSE (e.g., `epoch_65_step_6500`). This specific checkpoint file, containing the frozen, mathematically converged neural network weights, acts as the final surrogate model deployed for downstream `ShengBTE` evaluations.

5.7 ML-Accelerated Extraction of Anharmonic IFCs

With the optimized model saved, the training phase is fully decoupled from the evaluation phase. The pre-trained network is now used to execute a single forward-pass inference on the exact atomic coordinates required by `ShengBTE`.

5.7.1 Evaluation Inference

The evaluation module is executed using the `validate.ini` configuration, defining `single_step = True` to ensure only the selected checkpoint is evaluated without initiating a training loop:

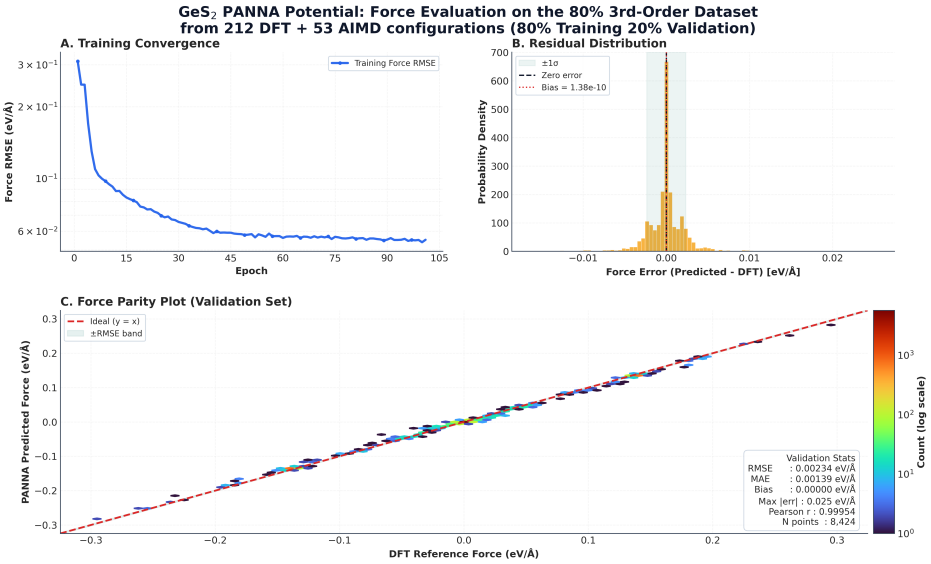

Figure 5.7: PANNA model trained on 80% of the combined DFT + AIMD dataset and validated on the remaining 20%, demonstrating robust generalization across both the harmonic and anharmonic regimes.

```
mkdir sow_eval
python3 /path/to/panna/evaluate.py --config validate.ini
```

The output is a single `.dat` file containing the exact predicted F_x, F_y, F_z force vectors for every atom across all 212 systematic displacements.

5.7.2 Force Substitution and ShengBTE Integration

These predicted forces must be formatted correctly for the BTE solver. A custom Python script, `inject_forces.py`, serves as an interface. It parses the `.dat` output, converts the forces from `PANNA` units (eV/Å) to Rydberg atomic units (Ry/bohr) using a conversion factor of 0.03889379, and injects them into standard QUANTUM ESPRESSO output templates.

Crucially, any existing dummy or equilibrium forces in the original DFT files are strictly overwritten by the ML predictions to prevent array dimension mismatches. These modified files (`*.ml.out`) are then processed by the `thirdorder_espresso.py` utility to reconstruct the 3rd-order IFC tensor via finite differences, which is subsequently fed into the `ShengBTE` iterative solver.

5.8 Thermal Conductivity Predictions: ML vs. DFT

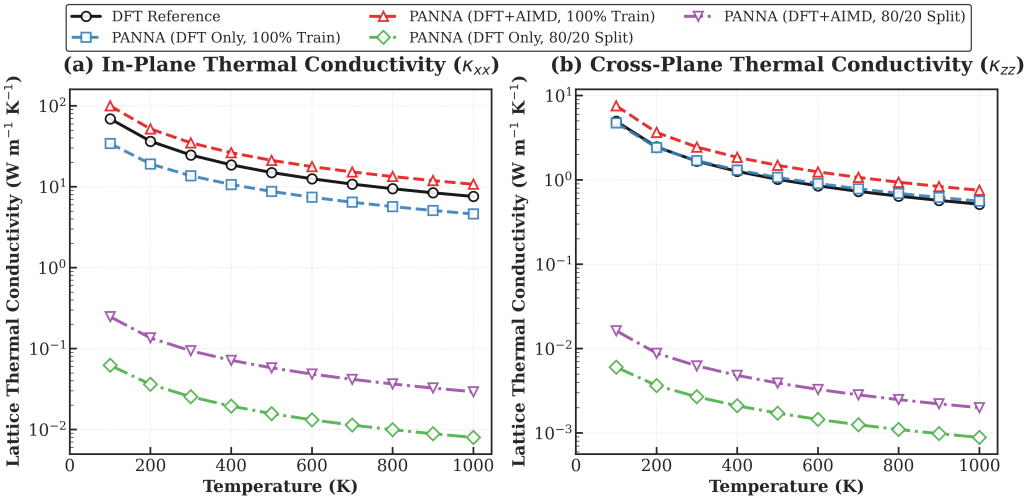

Figure 5.8: Comparison of the in-plane (κ_{xx}) and cross-plane (κ_{zz}) lattice thermal conductivity of GeS_2 predicted by different PANNA models and the DFT reference. The results are shown on a logarithmic scale to highlight the large differences between the models.

Using the interatomic force constants (IFCs) predicted by the machine learning model, we calculated the temperature-dependent lattice thermal conductivity (κ) and compared the results directly against the reference values obtained from explicit DFT calculations.

Figure 5.8 illustrates a clear divergence in the behavior of the models depending entirely on their training regimens. The models distinctly separate into two groups: the first group comprises models trained with limited or withheld data, which suffer heavily from numerical noise. The second group comprises models trained using the full hybrid dataset, which successfully reproduce the physical transport behavior of the material.

5.8.1 Failure of Data-Limited Models

Figure 5.9 displays the results for the models trained using a standard 80/20 train-validation split. In these scenarios, only 80% of the available phase-space data was utilized to train the neural network.

The results show that these data-limited models fundamentally fail to predict the correct thermal conductivity. Even when AIMD data is included,

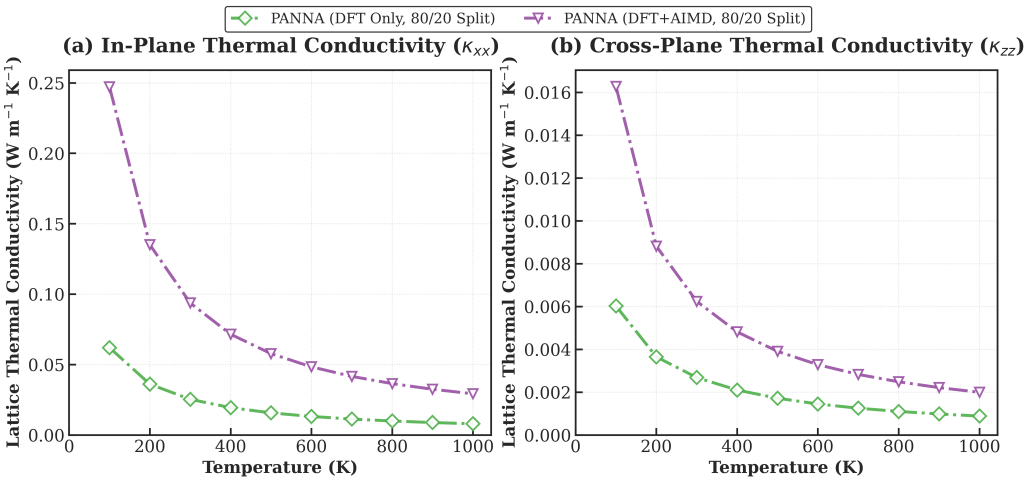

Figure 5.9: Temperature dependence of κ_{xx} and κ_{zz} predicted by the PANNA models trained using an 80/20 train-validation split. The reduced training data leads to noisy force predictions, which artificially lowers the thermal conductivity.

withholding 20% of the configurations critically impairs the ability of the neural network to map the full, continuous energy landscape of the system.

This behavior strongly corroborates the sensitivity analysis discussed earlier in Section 5.5. Because these models were trained with insufficient data coverage, the predicted forces contain localized numerical noise with errors exceeding 1 meV/Å. When these noisy forces are substituted into the phonon Boltzmann Transport Equation, the solver interprets the mathematical anomalies as strong physical anharmonic scattering.

As a direct consequence, phonons experience excessive scattering, drastically reducing their mean free paths. This artificial scattering effect causes the predicted thermal conductivity to collapse by several orders of magnitude, yielding physically unrealistic thermal transport predictions.

5.8.2 Improved Results with Hybrid Training Data

In contrast, Figure 5.10 presents the results for models trained using 100% of the available dataset. Under these conditions, the numerical noise is suppressed below the necessary critical thresholds, allowing phonons to propagate realistically through the lattice model.

The model trained strictly on DFT displacement data (`PANNA DFT Only, 100% Train`) manages to predict reasonable thermal conductivity values. However, it still systematically underestimates the DFT reference results.

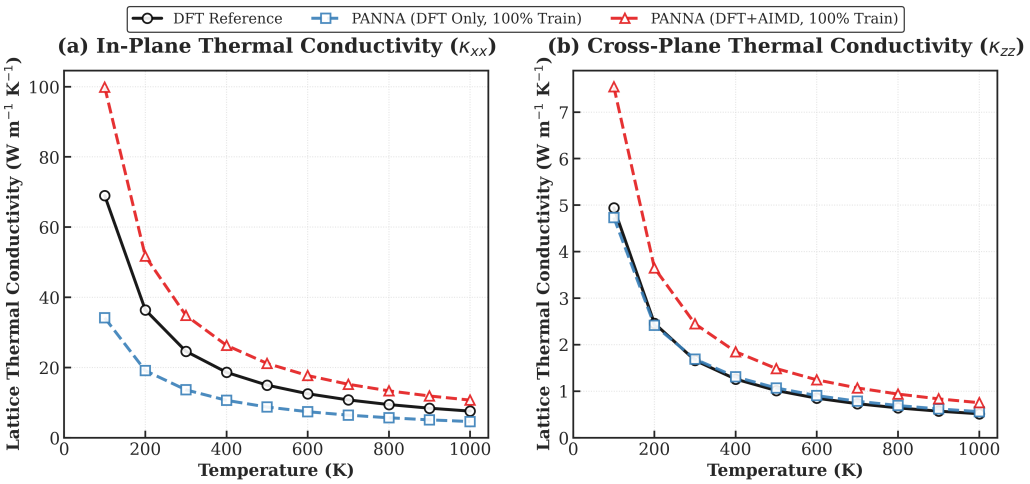

Figure 5.10: Temperature dependence of κ_{xx} and κ_{zz} predicted by PANNA models trained using the full dataset, compared with the DFT reference results.

This discrepancy arises because the model was trained exclusively on small, near-equilibrium harmonic displacements, inherently preventing it from fully capturing the complex anharmonic scattering behavior of the high-temperature system.

When the structurally chaotic AIMD data (sampled at 500 K) is combined into the training set, the model successfully learns a more realistic description of the anharmonic energy landscape. The resulting hybrid model (`PANNA DFT+AIMD, 100% Train`) produces thermal conductivity predictions that closely align with the DFT reference, particularly at higher temperatures where anharmonic phonon scattering dominates thermal resistance.

5.9 Summary

In this chapter, a comprehensive machine learning workflow was established to build and apply neural network potentials for studying thermal transport in layered GeS_2. The analysis demonstrates that the strict precision of predicted atomic forces is paramount for reliable thermal conductivity calculations.

The sensitivity study explicitly quantified that force prediction errors must be maintained below approximately 50 μeV/Å to actively avoid the artificial suppression of lattice thermal conductivity via synthetic scattering.

Furthermore, the validation results highlight that training exclusively on small finite-displacement data traps the model in a purely harmonic

description of the system. Including high-temperature AIMD configurations in the training dataset is strictly necessary to empower the model to learn the correct anharmonic behavior.

When correctly trained using this hybrid dataset methodology, the `PANNA` surrogate model successfully reproduces the first-principles thermal conductivity with high fidelity, while concurrently reducing the computational cost of the IFC evaluations by four orders of magnitude.

Chapter 6

Active Learning

6.1 Methods: Active Learning-Driven MLIP Development

6.1.1 Active Learning via Query-by-Committee

To systematically expand configurational coverage and escape the harmonic trap, we employed a Query-by-Committee (QBC) active learning framework. The committee consisted of five independently initialized neural networks:

$$\mathcal{C} = \{M_1, M_2, M_3, M_4, M_5\}\,. \tag{6.1}$$

For each unseen configuration, the force variance across the ensemble was computed to quantify epistemic uncertainty:

$$\sigma_F^2 = \frac{1}{N-1} \sum_{n=1}^{N} \left|\mathbf{F}_n - \bar{\mathbf{F}}\right|^2 . \tag{6.2}$$

Configurations exhibiting maximal σ_F^2 were selected for exact DFT labeling and subsequently appended to the training dataset. This iterative loop deliberately minimized the number of computationally expensive *ab initio*

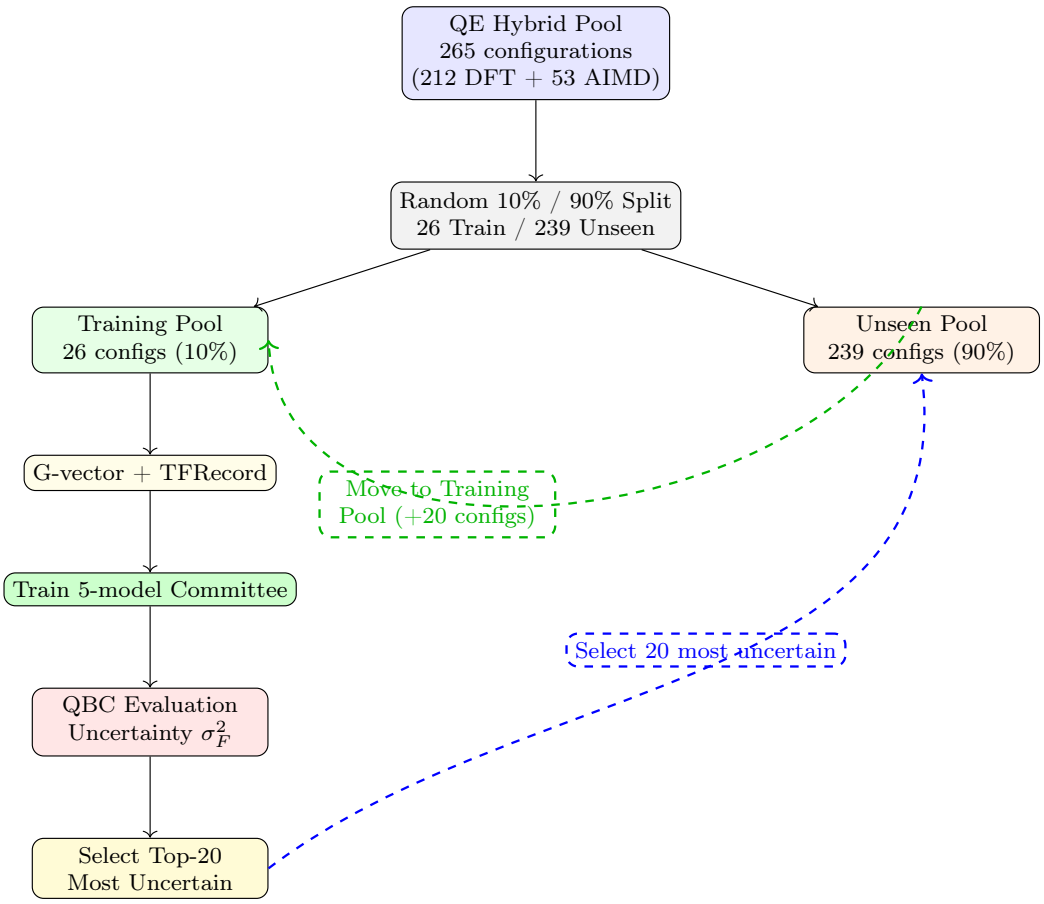

Figure 6.1: Pool-based Query-by-Committee workflow for the hybrid dataset. A random 10% seed (26/265) initializes the training pool. Each iteration promotes the 20 unseen configurations with the largest ensemble force variance (σ_F^2) to the training pool, and the 5-model committee is retrained until convergence.

evaluations while strictly maximizing information gain.

6.2 Workflow Schematic

6.3 Active Learning Efficacy: Overcoming Data Limitations and the Harmonic Trap

The stark contrast in predictive accuracy between statically split datasets and the active learning workflow highlights a fundamental challenge when applying machine learning to phonon transport calculations. A conventional random training-validation split routinely leaves critical regions of the potential energy surface unexplored, especially those corresponding to strongly distorted, non-equilibrium atomic configurations.

In contrast, the Query-by-Committee (QBC) active learning framework systematically fortifies the dataset by pinpointing configurations where the model predictions are highly uncertain. By iteratively incorporating these configurations into the training set, the neural network is forced to learn previously missing regions of the potential energy surface. This targeted strategy allows the model to progressively eliminate epistemic uncertainty, thereby significantly improving the physical fidelity of the predicted lattice dynamics.

To fully understand why active learning is strictly required for Boltzmann Transport Equation (BTE) calculations, it is necessary to examine three interconnected aspects: global force errors, the extreme sensitivity of lattice thermal conductivity to numerical force noise, and the evolution of model uncertainty during the active learning progression.

6.3.1 Limitations of Global Error Metrics

In many atomistic machine learning studies, model quality is assessed primarily using global statistical metrics, such as the Root Mean Square Error (RMSE) on a validation dataset. However, for predicting third-order interatomic force constants (IFCs), relying solely on global metrics can be highly misleading.

For example, the static hybrid model trained using an 80/20 data split (as detailed in Chapter 5) achieves an exceptionally low validation RMSE of approximately $0.00234\,\mathrm{eV/Å}$ and a Mean Absolute Error (MAE) of $0.00139\,\mathrm{eV/Å}$. The parity plot exhibits a strong correlation between predicted and reference forces, and the residual distribution appears perfectly centered around zero.

Despite this ostensibly excellent statistical performance, the thermal conductivity predicted using these forces is physically catastrophic. When the predicted forces are utilized to construct the third-order IFCs and passed to the `ShengBTE` solver, the resulting lattice thermal conductivity collapses to nearly zero in both the in-plane (κ_{xx}) and cross-plane (κ_{zz}) directions.

This observation unequivocally demonstrates that low global RMSE values alone do not guarantee physically meaningful predictions when modeling sensitive phonon transport phenomena.

6.3.2 Physical Consequences of Data Starvation

The fundamental failure of the statically split model can be understood by examining the extreme sensitivity of lattice thermal conductivity to microscopic force errors.

In the phonon Boltzmann Transport Equation, phonon scattering rates depend directly on the square of the anharmonic interaction matrix elements. As a result, even minute numerical noise in the predicted forces is immediately interpreted by the solver as synthetic anharmonic scattering.

The rigorous sensitivity analysis performed in this work (Section 5.5) established that force noise exceeding approximately 50 μeV/Å significantly suppresses the calculated thermal conductivity. If the neural network encounters atomic configurations that were poorly sampled in the training dataset, the predicted forces in those specific spatial regions will inevitably contain localized errors.

Although such errors may be numerically rare and thus minimally affect the global RMSE, they introduce highly localized noise anomalies in the third-order IFCs. When the BTE solver interprets this noise as additional phonon scattering centers, the resulting thermal conductivity is artificially and dramatically reduced. This phenomenon explains why the statically split model fails: the withheld 20% of the dataset contains atomic environments that are structurally essential for accurately describing the anharmonic potential energy surface.

6.3.3 Active Learning Progression

To systematically address these limitations, the Query-by-Committee active learning framework was deployed. The committee consists of five identical neural network architectures, differentiated only by their random weight initializations. The standard deviation of the predictions across the committee members provides a highly practical and localized estimate of model uncertainty.

During each iteration of the active learning loop, configurations yielding the largest committee disagreement are selectively queried and appended to the training dataset. This approach forces the model to focus strictly on regions of the configuration space where its predictive capabilities are currently unreliable.

Iteration 0: Initial Random Training

Figure 6.2 presents the committee evaluation for the initial training stage, where only 10% of the available dataset serves as the training seed. The per-model RMSE sits at approximately 0.037 eV/Å, indicating severely limited predictive accuracy.

More importantly, Panel D illustrates that committee disagreement scales exponentially with force magnitude. This explicitly indicates that the model struggles to predict forces for highly distorted atomic configurations, which correspond directly to the critical anharmonic regions of the potential energy surface.

Early Iterations

As seen in Figure 6.3, expanding the dataset to 20% during Iteration 1 immediately begins to compress the error bounds, though substantial disagreement remains in the high-force regimes.

Intermediate Iterations

As additional high-variance configurations are incorporated, the predictive accuracy improves drastically. Figure 6.4 displays the committee evaluation for Iteration 2, corresponding to approximately 30% of the active dataset.

The global RMSE decreases by more than an order of magnitude to approximately 0.0025 eV/Å. However, the disagreement plot still exhibits noticeable structural fluctuations at larger force magnitudes, indicating that several complex anharmonic configurations remain insufficiently represented.

Converged Model

After several AL iterations, the loop successfully incorporates approximately 80% of the most informative configurations. At this terminal stage, the committee disagreement becomes highly uniform across the entire force magnitude range.

Figure 6.5 shows the final committee evaluation. The global RMSE decreases further to approximately 0.0022 eV/Å. Crucially, the uncertainty

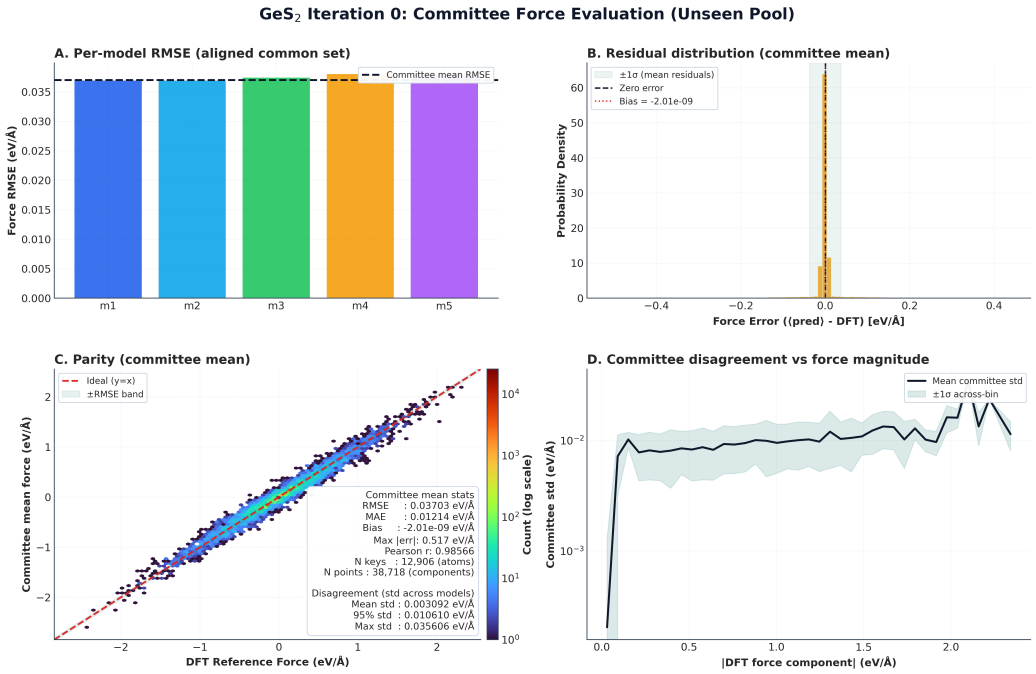

Figure 6.2: Committee evaluation for Iteration 0 using 10% of the dataset for training. (A) Force RMSE for each neural network model in the committee. (B) Distribution of force prediction residuals for the committee mean. (C) Parity plot comparing the committee mean predictions with DFT reference forces. (D) Committee disagreement as a function of force magnitude, highlighting large uncertainties for strongly distorted configurations.

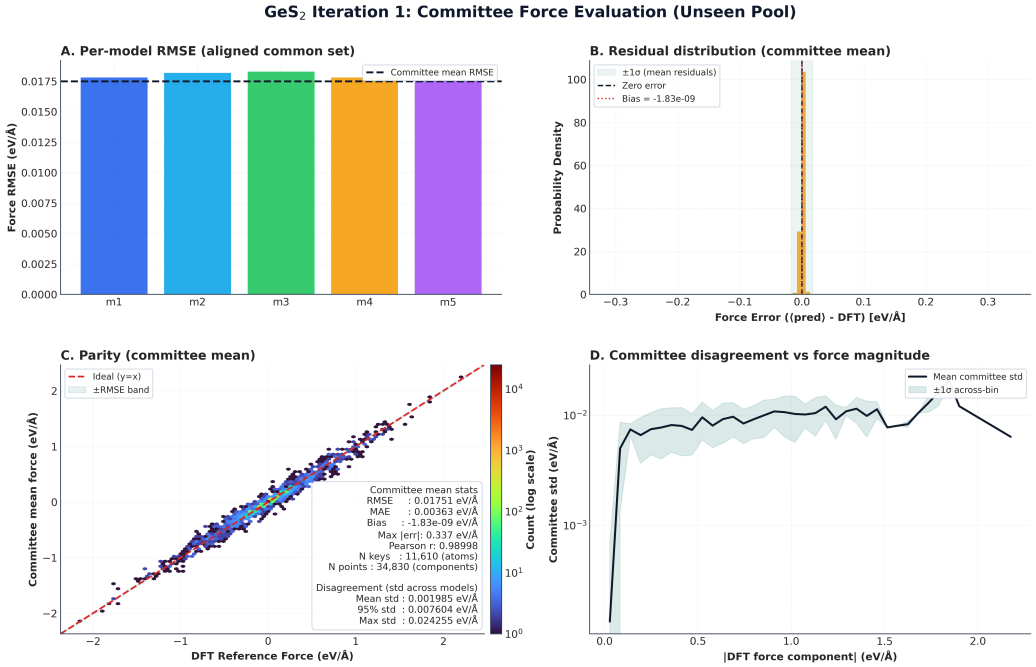

Figure 6.3: Committee evaluation for Iteration 1 using 20% of the dataset for training. (A) Force RMSE for each neural network model in the committee. (B) Distribution of force prediction residuals for the committee mean. (C) Parity plot comparing the committee mean predictions with DFT reference forces. (D) Committee disagreement as a function of force magnitude. While global error decreases, high-magnitude force uncertainty persists.

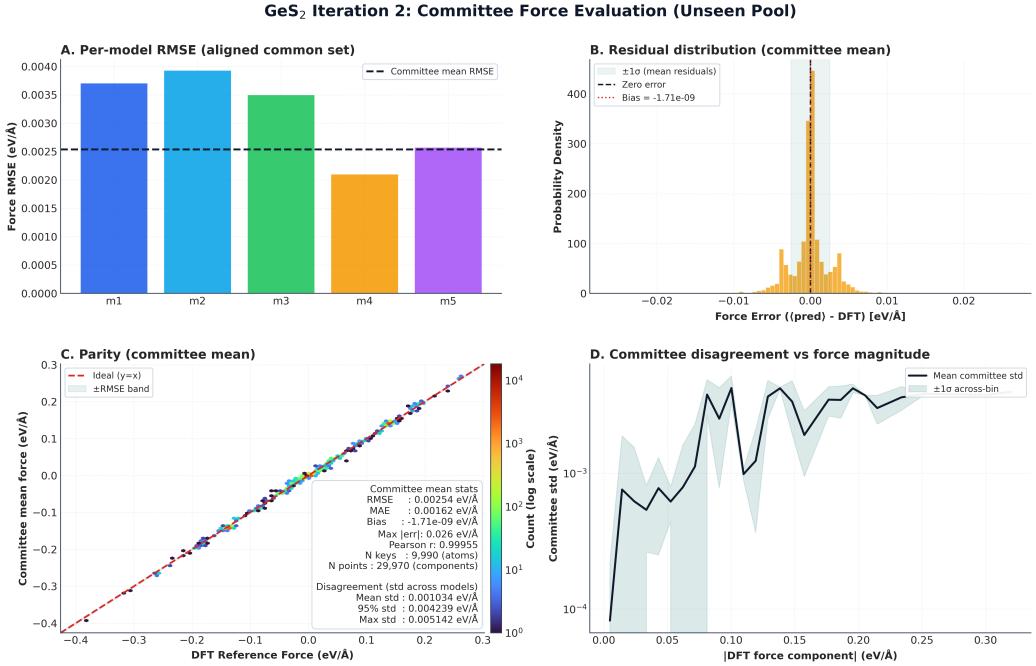

Figure 6.4: Committee evaluation after two active learning iterations (30% training data). Although the global RMSE is significantly reduced, the disagreement plot shows remaining uncertainty for large forces, indicating that some anharmonic configurations are still poorly represented in the training dataset.

distribution becomes significantly smoother and flatter, indicating that the neural network has successfully learned the previously missing, highly distorted regions of the potential energy surface.

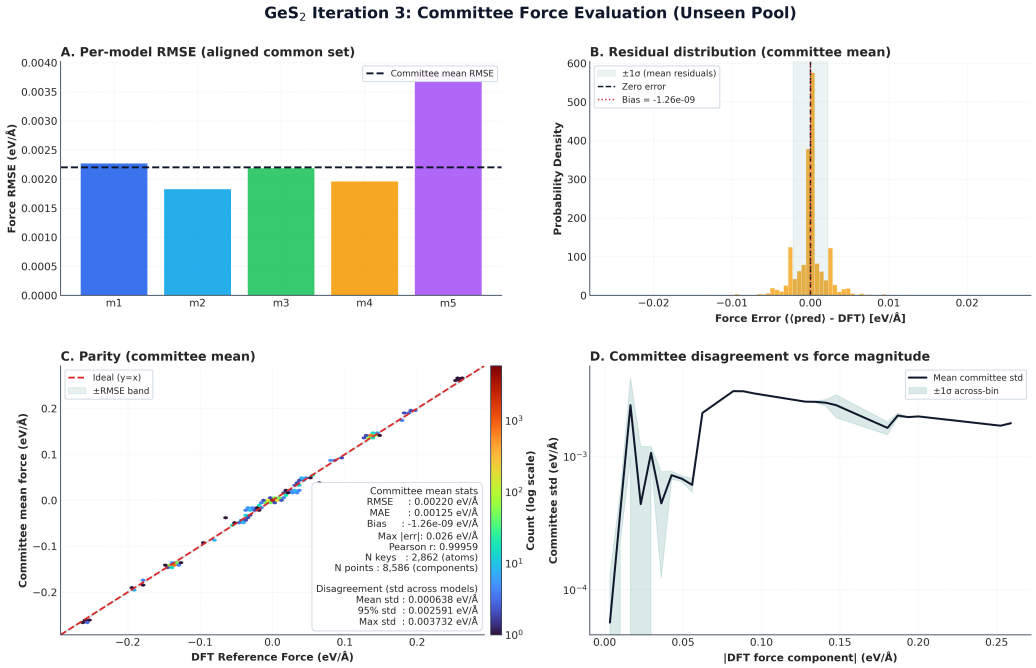

Figure 6.5: Committee evaluation for the final active learning stage using approximately 80% of the dataset. The committee disagreement is significantly reduced across the full force range, indicating that the model has successfully learned the anharmonic regions of the potential energy surface.

6.3.4 Impact on Thermal Conductivity Predictions

The targeted improvement in high-magnitude force prediction accuracy directly dictates the viability of the thermal conductivity predictions. Figure 6.6 compares the temperature-dependent lattice thermal conductivity calculated using models trained with varying fractions of the actively selected dataset.

Models trained with very small, initial datasets (10–30%) significantly underestimate the thermal conductivity due to excessive artificial phonon scattering induced by force prediction noise. As the active learning loop systematically curates the dataset and flattens model uncertainty across all configurational extremes, the predicted thermal conductivity gradually and stably converges onto the *ab initio* DFT reference values.

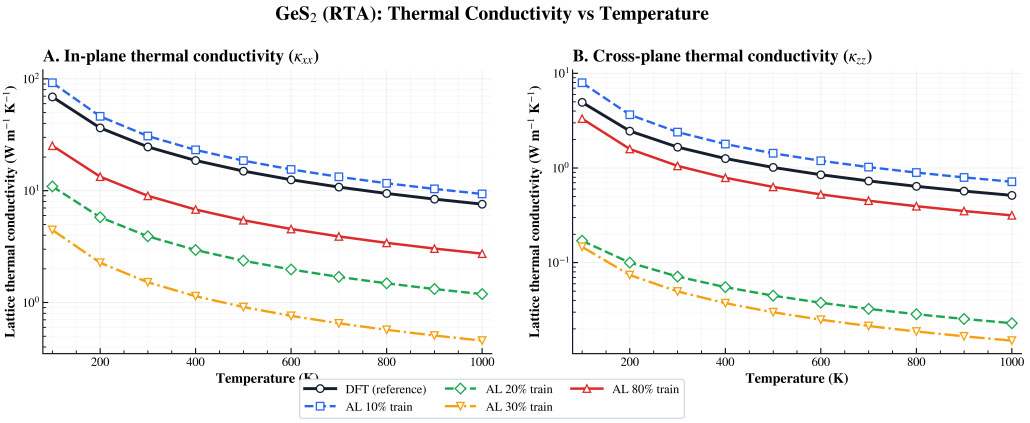

Figure 6.6: Comparison of lattice thermal conductivity predictions obtained using models trained with increasing amounts of actively selected data. The DFT reference is shown for validation. As the active learning dataset grows, the predicted conductivity systematically converges toward the explicit DFT result.

6.3.5 Summary

These results definitively demonstrate that active learning plays a mandatory, rather than optional, role in constructing reliable machine learning interatomic potentials for sensitive phonon transport calculations. While static, random dataset splits may successfully achieve low global force errors, they regularly produce physically catastrophic predictions due to localized noise in poorly sampled regions of the configuration space.

By iteratively identifying and incorporating high-uncertainty atomic configurations, the Query-by-Committee active learning framework ensures that the neural network comprehensively maps the full anharmonic potential energy surface. This targeted data curation is the key enabler for the accurate prediction of third-order force constants and the reliable, accelerated calculation of lattice thermal conductivity via the Boltzmann Transport Equation.

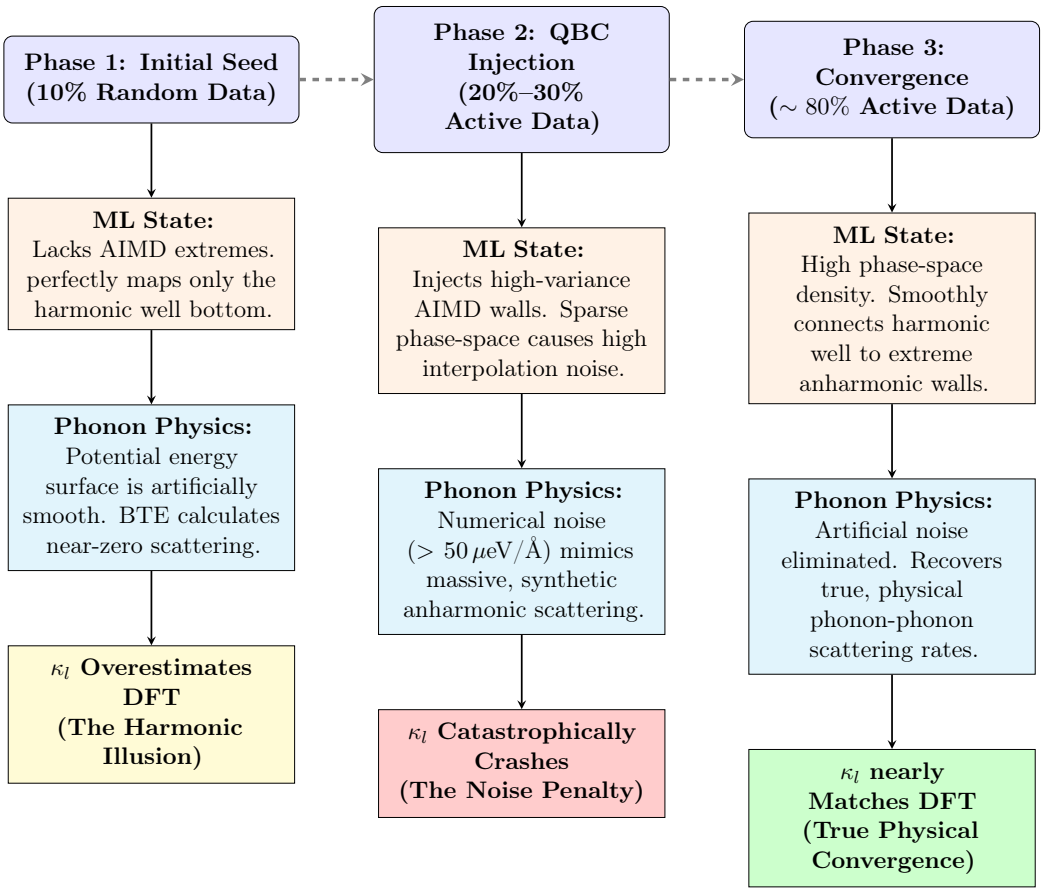

Figure 6.7: Evolution of the physical predictive capabilities during the Query-by-Committee (QBC) active learning process. The flowchart explicitly maps the transition from the artificial "harmonic illusion" caused by under-scattering (10% data), through the noise-induced synthetic scattering crash (20–30% data), to the physically accurate converged state (80% data).

Chapter 7

Conclusions and Future Perspectives

This thesis has established a Machine Learning (ML) accelerated computational framework for predicting the lattice thermal conductivity (κ_l) of the layered chalcogenide material GeS_2. By intimately coupling Density Functional Theory (DFT) calculations with the `PANNA` (Properties from Artificial Neural Network Architectures) framework, an efficient and robust workflow was developed to overcome the prohibitive computational costs normally associated with evaluating third-order anharmonic interatomic force constants.

The comprehensive results of this work demonstrate that machine learning potentials can significantly accelerate complex thermal transport calculations while strictly maintaining *ab initio* levels of accuracy, provided they are rigorously trained and validated. The primary findings and implications of this thesis are summarized below.

7.1 Computational Acceleration and Efficiency

A paramount achievement of this ML approach is the dramatic reduction in computational overhead. In this work, a deep neural network within the `PANNA` framework was effectively deployed as a high-fidelity surrogate model to predict atomic forces that would traditionally require immensely expensive DFT self-consistent field calculations.

Using the optimized neural network, the evaluation time for large supercell configurations was reduced by approximately four orders of magnitude compared to direct DFT calculations performed on GPU-accelerated High-Performance Computing (HPC) systems. This massive speed-up renders the study of vastly larger supercells and more complex, low-symmetry atomic

displacements computationally tractable.

7.2 Model Sensitivity and Noise Control

A systematic sensitivity analysis was performed to quantify exactly how microscopic errors in predicted forces propagate into the macroscopic calculation of lattice thermal conductivity. Random Gaussian noise was systematically injected into the force values to simulate the predictive variance inherent to machine learning models.

The results unequivocally demonstrated that the solution to the phonon Boltzmann Transport Equation (BTE) is hyper-sensitive to minute anomalies in atomic forces. Even marginal force perturbations act as artificial phonon scattering centers, drastically reducing the predicted thermal conductivity. The analysis established a strict accuracy threshold: force prediction errors exceeding approximately $50\,\mu\text{eV/Å}$ actively suppress thermal transport. Consequently, this dictates that an extraordinary level of precision is strictly required when developing ML potentials for phonon dynamics.

7.3 Data Scarcity, the Harmonic Trap, and Active Learning

Another foundational finding of this research concerns the topological diversity of the training data required to yield physically accurate machine learning models.

When a standard static data split (such as 80/20) was employed, the models suffered heavily from data starvation. Furthermore, training exclusively on the minor finite displacements generated by standard BTE solvers severely limited the model to a purely harmonic description of the lattice. This trapped the network in a "harmonic regime," rendering it incapable of accurately extrapolating the steep, anharmonic potential energy walls required to model phonon scattering.

To overcome this fundamental limitation, it was demonstrated that the training dataset must dynamically incorporate high-temperature configurations obtained via *ab initio* Molecular Dynamics (AIMD). By deploying a Query-by-Committee (QBC) active learning workflow, the neural network was systematically guided to sample high-variance, strongly distorted atomic environments. This hybrid, actively curated dataset empowered the model to learn the full anharmonic potential energy surface, enabling highly accurate predictions of lattice thermal conductivity.

7.4 Implications for Thermoelectric Applications

The accurate prediction of lattice thermal conductivity is a critical prerequisite for the rational design of efficient thermoelectric materials. Materials exhibiting inherently low thermal conductivity are highly sought after, as they directly enhance the dimensionless thermoelectric figure of merit (zT).

The results obtained in this thesis confirm that GeS_2 exhibits a relatively low lattice thermal conductivity. This advantageous property is primarily governed by its layered crystalline architecture and strong intrinsic anharmonicity, which collectively lead to enhanced phonon-phonon scattering.

Crucially, the ability to rapidly and accurately predict these thermal transport properties using MLIPs unlocks the possibility of performing high-throughput screening across the broader family of layered chalcogenides. This workflow could significantly accelerate the discovery and optimization of novel, Earth-abundant materials for advanced waste-heat recovery applications.

7.5 Current Limitations: The Local Descriptor Approximation

Although the ML-based workflow developed in this thesis demonstrates exceptional promise, specific physical limitations remain.

The predictive capability of the `PANNA` surrogate model relies heavily on the mathematical representation of the local atomic environment via modified Behler–Parrinello symmetry functions. Because these local descriptors decay to zero at a predefined, finite cutoff radius (R_c), they are inherently restricted to capturing short-range covalent interactions.

For layered van der Waals (vdW) materials like GeS_2, weak, long-range dispersion forces play a critical role in determining the structural and dynamical properties along the out-of-plane axis. Because the local symmetry functions cannot fully encapsulate these non-local interactions, the neural network artificially stiffens the interactions across the vdW gap. Consequently, the cross-plane thermal conductivity (κ_{zz}) is somewhat overestimated compared to explicit DFT reference calculations. Materials with strong electronic polarizability may similarly suffer unless long-range electrostatics are explicitly resolved.

7.6 Future Perspectives: Improving Long-Range Interaction Modeling

To push the boundaries of ML-driven materials modeling, several strategic avenues should be explored in future work to accurately capture long-range interactions in layered systems.

7.6.1 Inclusion of Semi-Empirical Dispersion Corrections

An immediate solution involves formulating a hybrid potential where the neural network is explicitly trained to govern only the short-range, complex covalent interactions, while a deterministic, semi-empirical dispersion correction (such as Grimme's DFT-D3 or DFT-D4) is superimposed during the force evaluation phase.

This physics-informed approach directly injects the missing long-range $1/r^6$ van der Waals interactions between adjacent layers. Decoupling these interaction regimes would help restore the true "softness" of the interlayer region, thereby correcting the overestimation of the cross-plane thermal conductivity.

7.6.2 Advanced Graph Neural Networks (GNNs)

Recent paradigm shifts in atomistic machine learning have introduced highly expressive architectures based on Message Passing Neural Networks (MPNNs), such as NequIP, Allegro, and CHGNet.

Unlike traditional Behler–Parrinello potentials that strictly truncate information at a fixed cutoff sphere, MPNNs allow localized atomic information to iteratively propagate through multiple layers of the graph network. This effectively extends the receptive field of the model without sacrificing rotational equivariance, enabling the network to organically learn complex, many-body, and moderately long-range interactions. Adopting such architectures may seamlessly resolve the current limitations surrounding the van der Waals gap.

7.6.3 Targeted Active Learning for Interlayer Dynamics

Finally, the Active Learning framework developed herein can be further specialized. While the current QBC loop successfully captured highly anharmonic in-plane distortions, future datasets must intentionally sample the soft, out-of-plane vibrational modes. The active learning candidate pool

could be explicitly seeded with configurations probing these specific degrees of freedom, including:

- Systematic interlayer sliding and registry shifts.
- Macroscopic expansion and compression along the crystallographic c-axis.
- Complex shear distortions between adjacent covalently bonded layers.

By forcing the committee to evaluate and learn these specific non-equilibrium geometries, the neural network could accurately capture the subtle physics governing interlayer dynamics, paving the way for flawless 3D thermal transport predictions in layered van der Waals heterostructures.

Bibliography

[1] Richard M. Martin. *Electronic Structure: Basic Theory and Practical Methods*. Cambridge University Press, 2004.

[2] David Sholl and Janice A. Steckel. *Density Functional Theory: A Practical Introduction*. John Wiley & Sons, 2011.

[3] Feliciano Giustino. *Materials Modelling using Density Functional Theory: Properties and Predictions*. Oxford University Press, 2014.

[4] M. Born and R. Oppenheimer. Zur quantentheorie der molekeln. *Annalen der Physik*, 389(20):457–484, 1927.

[5] David R. Yarkony. Diabolic points: a unified view of electron nuclear coupling in non-adiabatic processes. *Reviews of Modern Physics*, 68(4):985, 1996.

[6] Douglas R. Hartree. The wave mechanics of an atom with a non-coulomb central field. part i. theory and methods. *Mathematical Proceedings of the Cambridge Philosophical Society*, 24(1):89–110, 1928.

[7] V. Fock. Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems. *Zeitschrift für Physik*, 61(1):126–148, 1930.

[8] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Physical Review*, 136:B864–B871, 1964.

[9] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140:A1133–A1138, 1965.

[10] D. M. Ceperley and B. J. Alder. Ground state of the electron gas by a quantum monte carlo method. *Physical Review Letters*, 45(7):566, 1980.

[11] Charles Kittel. *Introduction to Solid State Physics*. John Wiley & Sons, 2004.

[12] John M. Ziman. *Electrons and Phonons: The Theory of Transport Phenomena in Solids*. Oxford University Press, 2001.

[13] Stefano Baroni, Stefano de Gironcoli, Andrea Dal Corso, and Paolo Giannozzi. Phonons and related crystal properties from density-functional perturbation theory. *Reviews of Modern Physics*, 73(2):515, 2001.

[14] Stefano de Gironcoli. Lattice dynamics of metals from density-functional perturbation theory. *Physical Review B*, 51(10):6773–6776, 1995.

[15] Wu Li, Jesus Carrete, Nebil A. Katcho, and Natalio Mingo. Shengbte: A solver of the boltzmann transport equation for phonons. *Computer Physics Communications*, 185(6):1747–1758, 2014.

[16] P. Giannozzi, O. Andreussi, T. Brumme, O. Bunau, M. Buongiorno Nardelli, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, M. Cococcioni, N. Colonna, I. Carnimeo, A. Dal Corso, S. de Gironcoli, P. Delugas, R. A. DiStasio Jr, A. Ferretti, A. Floris, G. Fratesi, G. Fugallo, R. Gebauer, U. Gerstmann, F. Giustino, T. Gorni, J. Jia, M. Kawamura, H.-Y. Ko, A. Kokalj, E. Küçükbenli, M. Lazzeri, M. Marsili, N. Marzari, F. Mauri, N. L. Nguyen, H.-V. Nguyen, A. Otero-de-la Roza, L. Paulatto, S. Poncé, D. Rocca, R. Sabatini, B. Santra, M. Schlipf, A. P. Seitsonen, A. Smogunov, I. Timrov, T. Thonhauser, P. Umari, N. Vast, X. Wu, and S. Baroni. Advanced capabilities for materials modelling with quantum espresso. *Journal of Physics: Condensed Matter*, 29(46):465901, 2017.

[17] Paolo Giannozzi, Stefano Baroni, Nicola Bonini, Matteo Calandra, Riccardo Car, Carlo Cavazzoni, Davide Ceresoli, Guido L. Chiarotti, Matteo Cococcioni, Ismaila Dabo, et al. Quantum espresso: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter*, 21(39):395502, 2009.

[18] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical Review Letters*, 77:3865–3868, 1996.

[19] J. P. Perdew et al. Generalized gradient approximation made simple. *Physical Review Letters*, 77:3865–3868, 1996.

[20] Andrea Dal Corso. Pseudopotentials periodic table: From h to pu. *Computational Materials Science*, 95:337–350, 2014.

[21] David Vanderbilt. Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. *Physical Review B*, 41(11):7892–7895, 1990.

[22] Hendrik J. Monkhorst and James D. Pack. Special points for brillouin-zone integrations. *Physical Review B*, 13(12):5188–5192, 1976.

[23] Yuanyue Liu, Nathan O. Weiss, Xidong Duan, Hung-Chih Cheng, Yu Huang, and Xiangfeng Duan. Van der waals heterostructures and devices. *Nature Reviews Materials*, 1(9):16042, 2016.

[24] J. D. Pack and H. J. Monkhorst. Special points for brillouin-zone integrations. *Physical Review B*, 13:5188–5192, 1983.

[25] A. K. Geim and I. V. Grigorieva. Van der waals heterostructures. *Nature*, 499(7459):419–425, 2013.

[26] Jiří Klimeš, David R. Bowler, and Angelos Michaelides. Chemical accuracy for the van der waals density functional. *Journal of Physics: Condensed Matter*, 22(2):022201, 2010.

[27] Stefan Grimme. Semiempirical gga-type density functional constructed with a long-range dispersion correction. *Journal of Computational Chemistry*, 27(15):1787–1799, 2006.

[28] Stefan Grimme, Jens Antony, Stephan Ehrlich, and Stephan Krieg. A consistent and accurate *ab initio* parametrization of density functional dispersion correction (dft-d) for the 94 elements h-pu. *The Journal of Chemical Physics*, 132(15):154104, 2010.

[29] Bernd G. Pfrommer, Amal I. Martínez, Adam R. Tackett, and Steven G. Louie. Relaxation algorithms for *ab initio* molecular dynamics. *Journal of Computational Physics*, 131(1):233–240, 1997.

[30] J. Carrete, B. Vermeersch, N. Mingo, et al. almabte : A software package for thermal transport in device-level simulations of materials. *Computer Physics Communications*, 220:351–362, 2017.

[31] A. Togo and I. Tanaka. Spglib: a software library for crystal symmetry search. *arXiv preprint arXiv:1808.01590*, 2018.

[32] R. Lot, Y. Pellegrini, Y. Shaidu, et al. Panna: Properties from artificial neural network architectures. *Computer Physics Communications*, 256:107402, 2019.

[33] J. Behler and M. Parrinello. Generalized neural-network potentials for multicomponent systems. *Physical Review Letters*, 98:146401, 2007.

[34] Andrey S. Tverjanovich, Oleg B. Tsiok, Vadim V. Brazhkin, Maria Bokova, Arnaud Cuisset, and Eugene Bychkov. Remarkably stable glassy GeS_2 densified at 8.3 GPa: Hidden polyamorphism, contrasting optical properties, raman and DFT studies, and advanced applications. *The Journal of Physical Chemistry B*, 127(45):9850–9860, Nov 2023. PMID: 37910778.

[35] Xudong Wang, Jieling Tan, Chengqian Han, Jiang-Jing Wang, Lu Lu, Hongchu Du, Chun-Lin Jia, Volker L. Deringer, Jian Zhou, and Wei Zhang. Sub-angstrom characterization of the structural origin for high in-plane anisotropy in 2D GeS_2. *ACS Nano*, 14(4):4456–4462, Apr 2020.

Annexure

GeS2.scf.in

```
&CONTROL
   prefix = 'pwscf'
   calculation = 'scf'
   restart_mode = 'from_scratch'
!  verbosity = 'high'
!  wf_collect = .true.
   tstress = .true.
   tprnfor = .true.
   outdir = './tmp'
   pseudo_dir = '/leonardo/home/userexternal/nkulhari/PPs'
!  etot_conv_thr = 0.0000001
!  forc_conv_thr = 0.000001
!  nstep = 400
/

&SYSTEM
   ibrav = 0,
   celldm(1) = 6.67592990
   nat = 6,
   ntyp = 2,
   ecutwfc = 65,
   ecutrho = 520,
   vdw_corr = 'grimme-d2'
!  occupations = 'smearing',
!  smearing = 'mv',
!  degauss = 0.005d0,
!  nbnd = 52
/

&ELECTRONS
   conv_thr = 1d-010,
   mixing_beta = 0.5d0,
   mixing_ndim = 8
   diagonalization = 'david'
   diago_david_ndim = 4
   diago_full_acc = .true.
   startingwfc = 'random'
```

```
/

ATOMIC_SPECIES
   Ge 72.640000d0 Ge.pbe-n-rrkjus_psl.1.0.0.UPF
   S 32.065000d0 S.pbe-n-rrkjus_psl.1.0.0.UPF

CELL_PARAMETERS alat
   0.991634235 0.000000000 0.000000000
   0.000000000 0.991634235 0.000000000
   0.000000000 0.000000000 3.114821979

ATOMIC_POSITIONS (crystal)
   Ge 0.000000000 0.000000000 0.500000000
   Ge 0.500000000 0.500000000 0.000000000
   S 0.000000000 0.500000000 0.370911775
   S 0.000000000 0.500000000 0.870911775
   S 0.500000000 0.000000000 0.129088225
   S 0.500000000 0.000000000 0.629088225

K_POINTS {automatic}
   9 9 3 0 0 0
```

GeS2.sc.scf.in

```
&CONTROL
   prefix = 'pwscf'
   calculation = 'scf'
   restart_mode = 'from_scratch'
   tstress = .true.
   tprnfor = .true.
   outdir = './tmp'
   pseudo_dir = '/leonardo/home/userexternal/nkulhari/PPs'
/

&SYSTEM
   ibrav = 0,
   nat = ##NATOMS##,
   ntyp = 2,
   ecutwfc = 65,
   ecutrho = 520,
   vdw_corr = 'grimme-d2'
```

```
/

&ELECTRONS
   conv_thr = 1d-010,
   mixing_beta = 0.5d0,
   mixing_ndim = 8
   diagonalization = 'david'
   diago_david_ndim = 4
   diago_full_acc = .true.
   startingwfc = 'random'
/

ATOMIC_SPECIES
   Ge 72.640000d0 Ge.pbe-n-rrkjus_psl.1.0.0.UPF
   S 32.065000d0 S.pbe-n-rrkjus_psl.1.0.0.UPF

##COORDINATES##

##CELL##

K_POINTS {automatic}
   3 3 1 0 0 0
```

```
#!/usr/bin/env python3
"""
inject_forces.py  (FINAL for AL workflow)

Hybrid injection for ShengBTE thirdorder.py (QE interface):
- If ML forces available for disp_XXX -> inject ML forces into QE
↪  output file.
- If ML forces missing for disp_XXX   -> copy original QE output
↪  (DFT) unchanged.
- Rattle/AIMD configs are ignored for thirdorder.py and are
↪  training-only.

Expected PANNA forces format (8 columns):
#filename atom_id fx_nn fy_nn fz_nn fx_ref fy_ref fz_ref
disp_101 0  ... ...

Run:
```

```
  cd input/
  python3 inject_forces.py
"""

import os
import shutil
from collections import defaultdict
from pathlib import Path

# -------------------- USER SETTINGS --------------------
PANNA_FORCES_DAT = "../committee_mean_forces.dat"
# or e.g.
↪  "../model_1/eval_unseen/results/epoch_240_step_2400_forces.dat"

SOURCE_DFT_DIR = "."                # contains DISP.scf_sc.out.001
↪  ... 212
TARGET_DIR     = "out_ml_hybrid"    # output folder to create

NCONF  = 212                        # disp configs: 001..212
NATOMS = 54                         # atoms in displaced supercell
CONV_EVA_TO_RYAU = 0.03889379       # (eV/Å) -> (Ry/Bohr)

QE_FORCE_MARKER = "Forces acting on atoms (cartesian axes, Ry/au):"
QE_TOTAL_FORCE  = "Total force ="

QE_ATOM_TYPE = 1                    # keep as 1 (matches your
↪  previous script)
# -------------------------------------------------------

def parse_panna_forces(path):
    """
    Returns: forces_dict[cid] = list of (atom_id, fx, fy, fz) in
    ↪  Ry/au
    where cid is '001'..'212'
    Only disp_* are kept.
    """
    if not os.path.exists(path):
        raise FileNotFoundError(f"PANNA forces file not found:
        ↪  {path}")

    forces_dict = defaultdict(list)

    with open(path, "r") as f:
        for line in f:
```

```
            if not line.strip() or line.lstrip().startswith("#"):
                continue
            parts = line.split()

            # Accept both 8-col (with ref) and 5-col (no ref)
            if len(parts) not in (5, 8):
                continue

            fname = parts[0]  # disp_101
            if not fname.startswith("disp_"):
                continue

            try:
                num = int(fname.split("_")[1])
                cid = f"{num:03d}"
                aid = int(parts[1])
                fx = float(parts[2]) * CONV_EVA_TO_RYAU
                fy = float(parts[3]) * CONV_EVA_TO_RYAU
                fz = float(parts[4]) * CONV_EVA_TO_RYAU
            except Exception:
                continue

            forces_dict[cid].append((aid, fx, fy, fz))

    # sort by atom_id
    for cid in list(forces_dict.keys()):
        forces_dict[cid].sort(key=lambda x: x[0])

    return forces_dict


def build_qe_forces_block(forces_ryau):
    """
    forces_ryau: list of (fx,fy,fz) length NATOMS
    """
    out = []
    out.append("     Forces acting on atoms (cartesian axes,
    ↪  Ry/au):\n\n")
    for i, (fx, fy, fz) in enumerate(forces_ryau, start=1):
        out.append(
            f"     atom {i:4d} type {QE_ATOM_TYPE:2d}   force = "
            f"{fx:15.8f} {fy:15.8f} {fz:15.8f}\n"
        )
    out.append("\n")
```

```
    out.append("     Total force =    0.00000000     Total SCF
    ↪  correction =    0.00000000\n\n")
    return "".join(out)


def replace_forces_block(qe_text, new_block):
    """
    Replace QE forces block beginning at QE_FORCE_MARKER and ending
    ↪  after the blank line
    following 'Total force = ...'. Returns modified text or None if
    ↪  markers not found.
    """
    start = qe_text.find(QE_FORCE_MARKER)
    if start == -1:
        return None

    total_idx = qe_text.find(QE_TOTAL_FORCE, start)
    if total_idx == -1:
        return None

    end = qe_text.find("\n\n", total_idx)
    if end == -1:
        # fallback: replace up to total_idx line end
        end = qe_text.find("\n", total_idx)
        if end == -1:
            return None
        end = end + 1
    else:
        end = end + 2  # include the blank line

    return qe_text[:start] + new_block + qe_text[end:]


def main():
    print(f"Reading PANNA forces from: {PANNA_FORCES_DAT}")
    forces_dict = parse_panna_forces(PANNA_FORCES_DAT)
    print(f"Found ML forces for {len(forces_dict)} disp
    ↪  configurations (out of {NCONF}).")

    Path(TARGET_DIR).mkdir(exist_ok=True)

    ml_count = 0
    dft_count = 0
    missing_src = 0
```

```
for i in range(1, NCONF + 1):
    cid = f"{i:03d}"
    src = os.path.join(SOURCE_DFT_DIR, f"DISP.scf_sc.out.{cid}")
    dst = os.path.join(TARGET_DIR,
    ↪  f"DISP.scf_sc.out.{cid}")

    if not os.path.exists(src):
        print(f"  MISSING: {src}")
        missing_src += 1
        continue

    has_ml = cid in forces_dict and len(forces_dict[cid]) >=
    ↪  NATOMS

    if not has_ml:
        shutil.copy2(src, dst)
        dft_count += 1
        continue

    rows = forces_dict[cid][:NATOMS]
    atom_ids = [r[0] for r in rows]

    # require complete ordered 0..NATOMS-1
    if atom_ids != list(range(NATOMS)):
        shutil.copy2(src, dst)
        dft_count += 1
        continue

    forces_ryau = [(r[1], r[2], r[3]) for r in rows]

    with open(src, "r") as f:
        text = f.read()

    new_block = build_qe_forces_block(forces_ryau)
    new_text = replace_forces_block(text, new_block)

    if new_text is None:
        # force block not found -> keep DFT
        shutil.copy2(src, dst)
        dft_count += 1
        continue

    with open(dst, "w") as f:
```

```
            f.write(new_text)

        ml_count += 1

    print("\n" + "=" * 60)
    print("Injection finished.")
    print(f"ML injected : {ml_count:3d}")
    print(f"DFT copied  : {dft_count:3d}")
    print(f"Missing src : {missing_src:3d}")
    print(f"Output dir  : {TARGET_DIR}/")

    # sanity check
    n_out = len(list(Path(TARGET_DIR).glob("DISP.scf_sc.out.*")))
    print(f"Files written: {n_out} (expected up to {NCONF})")
    print("=" * 60)
    print("Next step (example):")
    print(f"  ls {TARGET_DIR}/DISP.scf_sc.out.* | sort -V | "
          f"python3 ../../thirdorder/thirdorder_espresso.py
          ↪ GeS2.scf.in reap 3 3 1 -3")
    print("=" * 60)

if __name__ == "__main__":
    main()


     cat select_qbc.py
import os
import glob
import numpy as np

NUM_MODELS = 5
SELECT_FRAC = 0.10   # 10% of unseen each iteration
MIN_SELECT = 10
MAX_SELECT = 80

def newest_forces_file(model_i: int) -> str:
    cands =
    ↪ glob.glob(f"model_{model_i}/eval_unseen/results/*_forces.dat")
    if not cands:
        return ""
    cands.sort(key=os.path.getmtime, reverse=True)
    return cands[0]

def read_forces_dat(path):
```

```
    """
    Returns dict: forces[sid] = (N,3) numpy array of NN forces in
    ↪  atom_id order.
    Skips corrupted/wrapped lines.
    """
    forces = {}
    atom_ids = {}

    with open(path, "r") as f:
        for line in f:
            if not line.strip() or line.lstrip().startswith("#"):
                continue
            parts = line.split()
            if len(parts) != 8:
                continue  # skip wrapped garbage lines
            sid = parts[0]
            try:
                aid = int(parts[1])
                fx, fy, fz = float(parts[2]), float(parts[3]),
                ↪  float(parts[4])
            except Exception:
                continue

            forces.setdefault(sid, []).append((aid, fx, fy, fz))

    # sort by atom_id and convert to arrays
    out = {}
    for sid, rows in forces.items():
        rows.sort(key=lambda x: x[0])
        arr = np.array([[r[1], r[2], r[3]] for r in rows],
        ↪  dtype=float)
        out[sid] = arr
        atom_ids[sid] = [r[0] for r in rows]

    return out

def main():
    # unseen pool structure ids (authoritative)
    unseen_examples = sorted(glob.glob("unseen_out/*.example"))
    unseen_sids = [os.path.basename(x).replace(".example","") for x
    ↪  in unseen_examples]
    unseen_set = set(unseen_sids)
    n_unseen = len(unseen_sids)
    if n_unseen == 0:
```

```
    raise RuntimeError("No unseen_out/*.example found")

top_n = int(round(n_unseen * SELECT_FRAC))
top_n = max(MIN_SELECT, top_n)
top_n = min(MAX_SELECT, top_n)
print(f"Unseen pool size: {n_unseen}, selecting: {top_n}")

# Read forces from all models
model_forces = []
for i in range(1, NUM_MODELS+1):
    fp = newest_forces_file(i)
    if not fp:
        raise RuntimeError(f"No forces file for model_{i}")
    print(f"model_{i}: {fp}")
    model_forces.append(read_forces_dat(fp))

# Compute disagreement
ranking = []
for sid in unseen_sids:
    # must exist in all models
    if any(sid not in mf for mf in model_forces):
        continue

    Fs = [mf[sid] for mf in model_forces]  # list of (N,3)
    n_atoms = Fs[0].shape[0]
    if any(F.shape != (n_atoms,3) for F in Fs):
        continue

    arr = np.stack(Fs, axis=0)           # (M,N,3)
    std_vec = np.std(arr, axis=0)        # (N,3)
    std_mag = np.linalg.norm(std_vec, axis=1)

    mean_sigma = float(np.mean(std_mag))
    max_sigma  = float(np.max(std_mag))
    ranking.append((sid, mean_sigma, max_sigma, n_atoms))

ranking.sort(key=lambda x: x[1], reverse=True)

print(f"\n{'Rank':<5} {'Structure':<12} {'mean_sigma(eV/A)':>18}
↪  {'max_sigma(eV/A)':>18} {'N_atoms':>8}")
print("-"*80)
for k,(sid,ms,mx,nA) in enumerate(ranking[:top_n],1):
    print(f"{k:<5} {sid:<12} {ms:>18.6f} {mx:>18.6f} {nA:>8}")
```

```
    with open("selected_for_iteration_1.txt","w") as f:
        for sid,_,_,_ in ranking[:top_n]:
            f.write(sid+"\n")

    print(f"\nSaved TOP {top_n} to selected_for_iteration_1.txt")

if __name__ == "__main__":
    main()
```