A User Guide for *SLUSCHI*

Solid and Liquid in Ultra Small Coexistence with Hovering Interfaces

Qi-Jun Hong, Axel van de Walle

School of Engineering, Brown University, Providence, Rhode Island 02912, USA

Abstract

Although various approaches for melting point calculations from first principles have been proposed and employed for years, their practical implementation has hitherto remained a complex and time-consuming process. The *SLUSCHI* code (Solid and Liquid in Ultra Small Coexistence with Hovering Interfaces) drastically simplifies this procedure into an automated package, by implementing the recently-developed small-size coexistence method and putting together a series of steps that lead to final melting point evaluation. Based on density functional theory, *SLUSCHI* employs Born-Oppenheimer molecular dynamics techniques under the isobaric-isothermal (NPT) ensemble, with interface to the first-principles code *VASP*. In order to make this useful code available to a wide community of researchers who could benefit from it, this article outlines the procedure to perform melting point calculations, and presents a detailed user guide to the code.

1. Introduction

The calculation of solid-liquid phase boundary with first-principles methods has received considerable attention in computational physics and engineering. Over the last two decades, first-principles melting temperature (MT) calculations have been successfully performed in a variety of contexts for metallic, ionic, and covalent systems [1–19]. Various ingenious methods have been developed, and we list below some most commonly used approaches.

(a) The free energy method: This method calculates MT by locating the intersection of the free energy curves. While typically the solid-state free energy is obtained by the straightforward quasi-harmonic approximation [20], the liquid-state counterpart is generally more difficult to calculate and it requires more sophisticated methods [21], such as the thermodynamic integration method [22], the two-phase thermodynamics method [23], and the Widom particle insertion method [24].

(b) The traditional large-size coexistence method [25]: MT is determined by searching for stabilized solid-liquid coexistence in NPT MD simulation, so that the corresponding temperature is the MT. However, it requires a large system size to maintain stable coexistence [8].

(c) The fast-heating method (or so-called Z method) [26]: A solid is gradually heated until it melts, based on NVE MD simulations. While this approach is applicable to a small system, recent studies suggest that long simulations and corrections for finite simulation time might be needed to achieve the required precision [27–29].

Although these approaches have been known for years, their practical implementation hitherto remains tedious and demands a considerable amount of human-computer interaction. These practical issues keep the effort and cost high, and researchers have to repeatedly spend an unreasonable amount of their time writing input files for various computer codes, monitoring their execution and processing their output. These practical difficulties also hinder the spread of these methods, limiting their users solely to those who possess the necessary expertise to carry out such calculations.

By developing and distributing the *SLUSCHI* code [30], we aim to provide researchers in the community a simple and well-tested tool for MT calculations, so researchers are no longer constrained by the aforementioned limits. *SLUSCHI* is a fully automated code which calculates MT based on density functional theory (DFT) [31–33] Born-Oppenheimer molecular dynamics (MD) simulations with interface to the first-principles

*Corresponding author
Email address: qhong@alumni.caltech.edu (Qi-Jun Hong)

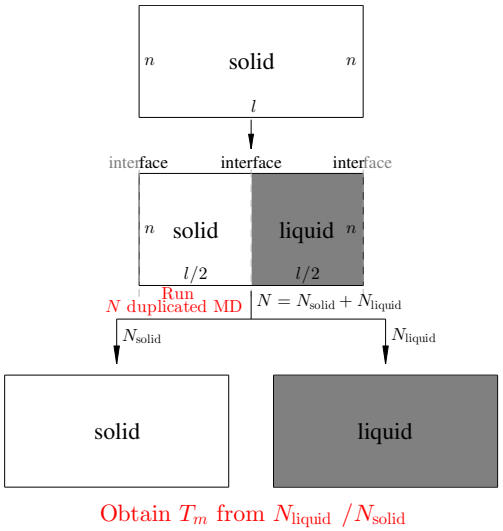

Figure 1: Schematic illustration of how small-size coexistence method is executed in practice. Starting from $n \times n \times l$ supercell with atoms on their ideal solid positions, we heat and melt the right half to obtain solid-liquid coexistence configurations. Then many parallel NPT MD simulations (here a total of $N = N_{\text{solid}} + N_{\text{liquid}}$) are performed, in order to measure the probability distribution.

code *VASP* [34–36]. Starting from a crystal structure the user specifies, *SLUSCHI* will automatically build a supercell of a proper size, prepare solid-liquid coexistence, and then employ the small-cell coexistence method [16], a highly accurate and efficient method we have developed, to calculate the MT. As this process is fully first-principles (i.e., free from the limitations of empirical potentials) and DFT is very broadly applicable, we expect that *SLUSCHI* will have wide applications in various circumstances, as we discuss in Section 4.1. We have so far utilized *SLUSCHI* to study a number of materials, including the Hf-Ta-C-N system [18], in which we computationally predict a material with the highest known MT, and lanthanum zirconate ($La_2Zr_2O_7$) [17], which is a ceramic with a complex solid structure. The details of these applications can be found in Table 3. We hope *SLUSCHI* benefits the community by providing robust and reliable MT predictions.

2. Theory

2.1. Small-cell coexistence method

The traditional coexistence method is prohibitively expensive in the framework of density functional formalism, due to the large simulation size required. Reducing the system size makes the calculation much faster, but the small size renders the coexistence unstable, because thermal fluctuations in the interface position have a magnitude similar to the simulation cell size. This problem is resolved by the small-cell coexistence method. The details of the method is available in our previous paper [16], which we briefly describe below. The small-cell coexistence method runs solid-liquid coexisting simulations on small-size systems, and the MTs are rigorously inferred based on statistical analysis of the fluctuations in the systems. The general procedure is summarized in Fig. 1. First, half of the supercell is heated and melted, in order to prepare multiple different configurations (snapshots) of half-half solid-liquid coexistence. These snapshots serve as starting points of NPT MD. For small system sizes, the two-phase coexistence quickly evolves to a single-phase equilibria, either entirely solid or entirely liquid, with a probability determined by the system's temperature relative to its MT. The solid-liquid probability distributions p_{solid} and p_{liquid} follow the relations

$$\frac{p_{\text{liquid}}}{p_{\text{solid}}} = \exp\left(-G^{l-s}(T) l_x/(2k_B T)\right), \tag{1}$$

$$H^{s/l}(T) = H^{s/l}(T_m) + C_p^{s/l}(T - T_m), \tag{2}$$

$$S^{s/l}(T) = S^{s/l}(T_m) + C_p^{s/l} \ln \frac{T}{T_m}, \tag{3}$$

$$G^{l-s}(T) = G^l(T) - G^s(T) = \frac{(T_m - T)}{T_m} H^{l-s}(T_m) - C_p^{l-s} \frac{(T - T_m)^2}{T_m}. \tag{4}$$

Detailed derivation of Eqn. (1) and its validation can be found in Ref. [16] and are omitted here. Through fitting, we obtain melting properties, e.g., MT T_m, solid and liquid enthalpies $H^{s/l}(T_m)$ at T_m and heat capacities $C_p^{s/l}$. Here G is Gibbs free energy, S is entropy, and l_x is a finite-size parameter.

2.2. MT fitting

SLUSCHI runs solid-liquid coexistence at multiple s temperatures $\mathbf{T} = \{T_1, T_2, \cdots, T_s\}$ with $\mathbf{n} = \{n_1, n_2, \cdots, n_s\}$ samples, which yield $\mathbf{k} = \{k_1, k_2, \cdots, k_s\}$ liquids and $\{n_1 - k_1, n_2 - k_2, \cdots, n_s - k_s\}$ solids, as well as their corresponding enthalpies during MD simulations. Because $C_p^{s/l}$ is the slope of enthalpy-temperature curve, $C_p^{s/l}$ is obtained through two independent linear fittings on the enthalpies of the solid and the liquid, as in Fig. 2(a). This leaves T_m and l_x the only two independent parameters yet to determine in Eqns. (1)-(4) (since $H^{l-s}(T_m)$ is a function of T_m).

The purpose of MT fitting is to infer the values of T_m and l_x from the sampled data ($\mathbf{T}$, $\mathbf{n}$, and $\mathbf{k}$). This is achieved following the principles of maximum likelihood estimation [37, 38].

2.2.1. Probability function

From a statistical point of view, the sampling of solid-liquid coexistence follows a binomial distribution with parameters n and p, i.e., the discrete probability distribution of the number of successes in a sequence of n independent yes/no experiments (Bernoulli trials), each of which yields success with probability p. The probability function f of getting k liquids (and $n - k$ solids) in n samples is

$$f\left(k, n|p\right) = C_n^k p^k \left(1-p\right)^{n-k}, \text{ where } p = \frac{p_{\text{liquid}}}{p_{\text{solid}} + p_{\text{liquid}}}. \tag{5}$$

Here p_{solid} and p_{liquid} are determined by the sampling temperature T, the MT T_m and the finite-size parameter l_x according to Eqns. (1)-(4). Hence

$$f\left(k, n|T, T_m, l_x\right) = f\left(k, n|p\left(T, T_m, l_x\right)\right) = C_n^k \left[p\left(T, T_m, l_x\right)\right]^k \left[1 - p\left(T, T_m, l_x\right)\right]^{n-k}. \tag{6}$$

The joint probability function for a set of vectors ($\mathbf{T}$, $\mathbf{n}$, and $\mathbf{k}$) is

$$f\left(\mathbf{k}, \mathbf{n}|\mathbf{T}, T_m, l_x\right) = \prod_{i=1}^{s} f_i\left(k_i, n_i|T_i, T_m, l_x\right) = \prod_{i=1}^{s} f_i\left(k_i, n_i|p_i\left(T_i, T_m, l_x\right)\right). \tag{7}$$

2.2.2. Likelihood function

Given a set of parameters $\{T_m, l_x\}$, the probability function $f\left(\mathbf{k}, \mathbf{n}|\mathbf{T}, T_m, l_x\right)$ tells how probable the data ($\mathbf{T}$, $\mathbf{n}$, and $\mathbf{k}$) are. However, what we are facing is an inverse problem: we already observe the data ($\mathbf{T}$, $\mathbf{n}$, and $\mathbf{k}$), and we need to find the "important" probability functions f with parameters $\{T_m, l_x\}$, i.e., those that are likely to have produced the data.

We define the likelihood function

$$\mathcal{L}\left(T_m, l_x|\mathbf{k}, \mathbf{n}, \mathbf{T}\right) = f\left(\mathbf{k}, \mathbf{n}|\mathbf{T}, T_m, l_x\right). \tag{8}$$

We note the difference between the probability function and the likelihood function. The probability function is a function defined on the data scale, given a particular set of parameter values. On the other hand, the likelihood function is a function defined on the parameter scale, given a particular set of observed data. In practice, we create a 2-D grid of T_m and l_x and evaluate the likelihood function $\mathcal{L}\left(T_m, l_x|\mathbf{k}, \mathbf{n}, \mathbf{T}\right)$ at each grid point $\{T_m, l_x\}$, while the observed data ($\mathbf{T}$, $\mathbf{n}$, and $\mathbf{k}$) are kept constant.

2.2.3. MT fitting

The MT fitting process in *SLUSCHI* consists of the following steps.

1. Because $C_p^{s/l}$ is the slope of enthalpy, $C_p^{s/l}$ is obtained by two independent linear fittings of the solid and the liquid, as in Fig. 2(a).
2. A 2-D grid of possible $\{T_m, l_x\}$ (e.g., a uniform grid in the x-y plane of Fig. 2(b)) is generated.

(now for each $\{T_m, l_x\}$, run step 3 and 4)

3. For each $\{T_m, l_x\}$, Eqn. (2) is determined, as well as $H^{l-s}(T_m)$. Then Eqn. (4) and thus Eqn. (1) are also determined. The probabilities $\mathbf{p} = \{p_1, p_2, \cdots, p_s\}$ are calculated for $\mathbf{T} = \{T_1, T_2, \cdots, T_s\}$ according to Eqns. (1)-(4).

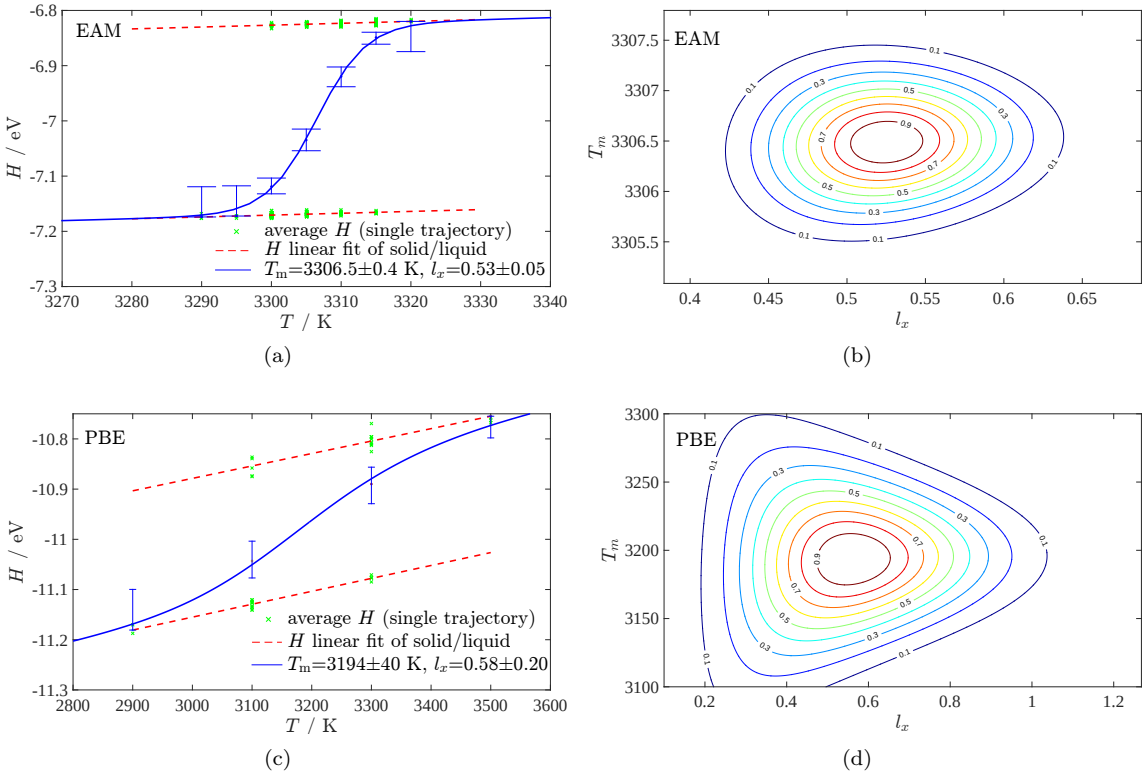

Figure 2: MT fitting in small-cell coexistence method. (a) and (b) show an ideal case with a tantalum embedded atom method (EAM) [39] potential (Li, 2003) [40], while (c) and (d) present a practical implementation based on the projector augmented wave (PAW) technique [41] and the Perdew-Burke-Ernzerhof (PBE) functional [42] with the $5p$ core electrons relaxed. The former empirical study is carried out on a $18\times9\times9$ body-centered cubic supercell with 412 trajectories sampled, which contrasts drastically with the latter DFT example with a small $6\times3\times3$ supercell and only 38 trajectories. The DFT example represents a challenging scenario where both the system size and sample size are very limited. Nevertheless, the small-cell coexistence method still works effectively, which suggests that the method is robust. (a) and (c): each green dot corresponds to one MD trajectory. The red lines are linear fits of the solid and liquid phases, according to Eqn. (2). For temperatures near the MT, there is a solid-liquid binomial distribution of the final states. Hence the MT can be obtained based on Eqns. (1)-(4). The blue curve corresponds to the set of parameters $\{T_m, l_x\}$ with the highest likelihood (the peak in (b) and (d)). (b) and (d): a 2-D contour of the likelihood function $\mathcal{L}(\{T_m, l_x\})$, which provides the mean and the standard deviation of the MT.

4. The likelihood function $\mathcal{L}\left(T_m, l_x|\mathbf{k},\mathbf{n},\mathbf{T}\right)$ for each set of parameters $\{T_m, l_x\}$ is evaluated according to Eqns. (5)-(8).

(after computing $\mathcal{L}\left(T_m, l_x|\mathbf{k},\mathbf{n},\mathbf{T}\right)$ for each $\{T_m, l_x\}$)

5. The discrete matrix of $\mathcal{L}\left(T_m, l_x|\mathbf{k},\mathbf{n},\mathbf{T}\right)$ (as a contour of it shown in Fig. 2(b)) is obtained. We generate a large number (e.g., 100,000) of data points $\{T_m, l_x\}$ following the likelihood function, and we compute the mean and the standard deviation of the MT.

Figure 2 shows the fitting of $C_p^{s/l}$, the binomial distribution at the sampled temperatures, and the calculated likelihood function of $\{T_m, l_x\}$.

2.3. SLUSCHI

SLUSCHI calculates MT following a list of steps, as summarized in Fig. 3. The details of input tags are described in Section 3.1.2.

1. The code `optunitcell` first runs a quick structure optimization on the unit-cell provided, under the pressure `$press` the user specifies. (i.e., volume and position relaxation (`ISIF=3` and `IBRION=2`) in *VASP*)
2. Based on the optimized unit-cell, the code `solid.x` builds a supercell, such that periodic images are separated by a distance `$radius` approximately (see `radius` tag in Section 3.1).
3. By running isobaric-isothermal (NPT) MD, the code `volsearch` determines the thermal expansion of the solid at the estimated MT `$temp` the user provides.
4. In `meltcoex`, two supercells are put together into a new supercell twice larger. Half of the supercell is frozen, i.e., the atoms are not allowed to move, while the other half is gradually heated in MD until the solid

Table 1: An example of job.in. Details of the tags are explained in Section 3.1.

```
intpol = 0
temp = 2500.0
press = 0.0
navg = 3
factor = 10000.0
confident = 0
error = 50.0
vaspcmd = sbatch
kmesh = 20
gamma_possible = 0
radius = 10.0
adj_potim = 1
tgt_nelm = 7
adj_nbands = 1
add_nbands = 100
adj_bmix = 1
diff_solid = 0.4
diff_liquid = 1.2
```

melts. Then snapshots of half-solid-half-liquid coexistence are collected from the MD trajectory.
5. The code `coexrun` finally runs MD simulations on solid-liquid coexistence, collect the results of solid-liquid distribution and calculate the MT based on the small-cell coexistence method.

3. Running *SLUSCHI*

3.1. Input files

Running *SLUSCHI* requires heavy interactions with the ab initio computation code *VASP*. Before launching a *SLUSCHI* run, all commands are preset in the input files. *SLUSCHI* handles the interactions in an automated manner during the run. The input files include the standard *VASP* input files and a job.in file with *SLUSCHI* input parameters.

3.1.1. Standard VASP Input files

- `INCAR` is used in various types of *VASP* calculations, ranging from structure optimization to MD simulations. Be aware that *SLUSCHI* will automatically adjust the following tags.
  - `ISMEAR` is set to -1 to enforce Fermi-Dirac distribution of the electrons, with `SIGMA` being the value k_BT (in eV) to impose the electronic temperature. This is a standard method of finite-temperature DFT developed by Mermin [43, 44].
  - `NSW` is set to 80. In *SLUSCHI*, the isobaric-isothermal ensemble (NPT) is realized through two steps consisting (1) canonical ensemble (NVT) MD for 80 ionic steps and (2) cell shape and size adjustment once every 80 ionic steps, in response to the average pressure. Our study suggests

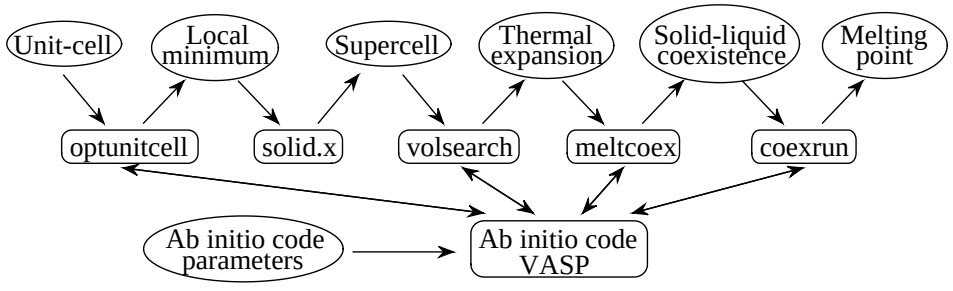

Figure 3: A diagram of *SLUSCHI*. Individual steps are carried out in sequence to approach MT. Interaction with *VASP* is heavily employed.

that this strategy provides a stable barostat, compared to a direct barostat. The latter may lead to considerable volume oscillation, due to the small system size and thus pressure instability.

  - `SMASS` is set to 0. `SMASS` is a tag which determines the thermomass. With this setting, the system and the thermal bath are coupled so that the energy exchange oscillates at a period of 40 ionic steps.
  - `TEBEG` and `TEEND` are set to the ionic temperature.
  - `NBANDS` and `POTIM` are adjusted based on previous electronic structure calculations (user may turn it off by setting `adj_nbands=0` and `adj_potim=0`. See Section 3.1.2).

- `POSCAR` provides the *unit-cell* atomic structure of the solid. *SLUSCHI* first optimizes this structure, and then builds a supercell with a proper size, specified by the `radius` tag. In order to maximize performance, we recommend using the primitive unit-cell, so that the smallest possible supercell that meets the `radius` requirement is generated.

- `POTCAR` needs to be in accordance with the `POSCAR`, just like standard *VASP* input.

- `KPOINTS` is not required. The `kmesh` tag in the `job.in` file specifies the k-mesh and generates the `KPOINTS` file.

- `jobsub` is a job script that runs *VASP*. For example, *VASP* jobs are submitted by

  `qsub jobsub` or `sbatch jobsub`

  One reminder is that the wall time limit needs to be long enough for a job to finish with 80 MD ionic steps and ∼100 atoms. (This typically requires at least 2 hours on 32 cores, but it may vary depending on the material system).

- `jobsub_gamma` and `KPOINTS_gamma` (both optional) run the Γ-point version of *VASP*. The use of Γ-point version saves the computer cost by a factor up to 50%. However, the Γ-point is usually not sufficient to represent the whole Brillouin zone. The user needs to take into account both cost and accuracy before choosing the Γ-point version. See discussions in Section 4.3 for details.

3.1.2. `job.in`

These tags in the `job.in` file specify parameters for MD simulations and MT calculations.

SLUSCHI explores the possible MT through self learning, and the calculation consists of two steps – searching and sampling. The "searching" process starts with an initial guess by the `temp` tag, and it stops when the MT "region" is found. The searching process first launches several MD trajectories at the temperature `$temp`. If this temperature is far away from the real MT, it is likely that all trajectories will end in one phase (For instance, if the temperature is too low, all coexistence will solidify. If the temperature happens to be close to the MT, then the searching process completes). Based on the outcomes of these trajectories, *SLUSCHI* chooses the next temperature to test, with a step size determined by the `confident` tag (For example, if all coexisting systems melt at T_1, the MT is likely lower than T_1 and *SLUSCHI* will sample at a lower temperature T_2). This step is repeated, until both solids and liquids "bracket" a temperature region, so that the MT is likely in the vicinity. After the searching process completes, the sampling process focuses on the temperature region found and extensive sampling is performed by running a large number of MD trajectories. MT fitting is carried out, which provides a MT estimate and its statistical error. If the error is larger than the criterion `$error`, more sampling follows until the accuracy requirement is met.

The details of the tags are described as follows.

- `intpol`: The user needs to provide an initial guess of the MT (in K), so *SLUSCHI* knows where to start the exploration. This temperature is also used to calculate thermal expansion for the solid phase, which is a prerequisite for MT calculations, as one needs to reach the desired pressure condition (i.e., the correct volume) before studying melting (see `temp` tag). The code has a feature to linearly interpolate `$temp` based on the MT value of its component elements. Set `intpol=1` to turn it on (note that the manual input of `$temp` will be overridden in this case).

- `temp`: With `intpol=0`, the code reads this tag for the MT estimate. We recommend setting this tag to the lower boundary of the MT, if some knowledge of the MT is available. Avoid putting a value higher than the actual MT, because this will undesirably melt the solid and *SLUSCHI* will fail to prepare

solid-liquid coexistence. In case there is not any knowledge about the MT, enter 500 for this tag and set the `confident` tag to 0.

- `confident`: If the user is confident about the temp tag entered, set confident to 1. Otherwise, put 0. With `confident=1`, the code will employ a smaller step (100 instead of 800 K) in the temperature search.

- `press`: MT usually depends strongly on pressure. This `$press` is the pressure condition that corresponds to the calculated MT. Enter the pressure in the unit of kbar. For ambient atmospheric pressure, enter 0 (or 0.001).

- `error`: This tag specifies the accuracy to achieve. Sampling (launching new MD samples) will continue, until the standard error is smaller than `$error`, as adding more samples reduces the statistical error of MT fitting. Reasonable values for `error` range from 20 to 100 (in K). An error of ∼50 K is relatively easy to achieve. Avoid entering a number too small (e.g., 5 K), as computer cost will skyrocket. Table 3 provides general guidance.

- `kmesh`: *SLUSCHI* adopts the automatic k-mesh generation in *VASP*. *SLUSCHI* will generate a `KPOINTS` file as

  ```
  Automatic mesh
  0
  Auto
  $kmesh.
  ```

  The k-point mesh is then defined as $(\tilde{l}_1, \tilde{l}_2, \tilde{l}_3)\times$`$kmesh`, where $\tilde{l}_i$ is the length of the reciprocal lattice vector. The lower `$kmesh`, the less dense the k-points and the lower the computer cost. Typically, set `kmesh` to 10-20 for insulators and 30-40 for metals (Some materials may require a value up to 100. See "automatic k-mesh generation" in *VASP* manual). This is generally safe to guarantee convergence with respect to k-points in energies and forces. Higher temperatures usually allow smaller `$kmesh`, because a broader Fermi-Dirac electronic distribution (due to the high electronic temperature) smooths out the states in the Brillouin zone. See Section 4.3 for discussion on Γ-point.

- `radius`: When *SLUSCHI* builds a supercell from the unit-cell input, this tag determines the distance between the periodic images and thus the supercell size. The "ideal" supercell size is $2a \times a \times a$ with angles 60°, 90° and 90° respectively, where a is specified by `radius` (in Å) and the solid-liquid interface is perpendicular to the first dimension. This cell shape theoretically minimizes the interaction among periodic images and hence the finite-size effect. The code will find the supercell that best fits the ideal size. In detail, it finds the lattice vectors $\vec{a}, \vec{b}, \vec{c}$ that minimize $\|\vec{a} - \vec{a_0}\|^2 + \left\|\vec{b} - \vec{b_0}\right\|^2 + \|\vec{c} - \vec{c_0}\|^2$, where $\vec{a_0}, \vec{b_0}, \vec{c_0}$ are the lattice vectors of the ideal supercell.

  We recommend use `radius=10` (Å), which is a good balance between cost and accuracy. A `$radius` too small may lead to a considerable finite-size error. Neither do we recommend setting `radius` too large, as it may significantly increase computer cost. A study of the finite-size error is available in Ref. [16], while Table 3 provides general guidance as well.

- `vaspcmd`: This is the command for job submission, e.g., `qsub` or `sbatch` typically.

- `navg`: This is a barostat parameter and it specifies the length of MD trajectories (80×`$navg` ionic steps) from which the average pressure is calculated. The default value is 3.

- `factor`: This is another barostat parameter and it determines the amount of cell adjustment in response to the average pressure. In detail,

$$L' = (I + P/f)L, \qquad L = \left(\vec{a}, \vec{b}, \vec{c}\right),$$

  where L and L' are the old and new lattice vector matrices ($\vec{a}, \vec{b}, \vec{c}$ are the three 3×1 lattice vectors), I is a 3×3 identity matrix, P is average stress tensor (in kbar) and f is `$factor`. The default value is 10000. In the case that stress/pressure is sensitive to volume (e.g., materials with high elastic modulus or materials under high pressure), increase `$factor`.

- `adj_potim` and `tgt_nelm` (both optional): `adj_potim` tells *SLUSCHI* whether to adjust *VASP* MD step size `$POTIM`. `adj_potim` can be set to 0 (no) or 1(yes, by default). With `adj_potim` set to 1, *SLUSCHI* will adjust the `POTIM` tag in `INCAR`, such that electronic structure calculations finish in `$tgt_nelm` (`tgt_nelm=4` by default) steps on average. This provides a general solution to the automated selection of `$POTIM`.

- `diff_solid` and `diff_liquid`: *SLUSCHI* distinguishes solids and liquids based on their diffusions. While diffusion rarely happens in the solid phase, atoms in the liquid are highly mobile. Based on this distinguishing property, these two tags helps *SLUSCHI* to determine whether the system has reached a pure phase, and whether it is a solid or a liquid.

  By default, `diff_solid=0.5` and `diff_liquid=1.0`. While this setting is generally valid, exceptions exist and manual inputs may be required in such cases. In practice, if the user finds that *SLUSCHI* mistakenly labels a coexisting structure as a solid, `$diff_solid` should be decreased, in order to have a strict criterion. Similarly, if *SLUSCHI* mistakenly labels a coexisting structure as a liquid, increase `$diff_liquid`.

3.2. Running the code

Running *SLUSCHI* is simple. After the user has prepared the files described in the previous section, the `sluschi` directory should contain these files and a folder.

```
INCAR POSCAR POTCAR job.in jobsub
```

Optional files such as `KPOINTS_gamma` and `jobsub_gamma` may also present.

Then run the main executable `SLUSCHI`.

```
SLUSCHI &
```

It may take from one day up to several weeks (see Table 3) for *SLUSCHI* to obtain a reasonable MT and to reach the accuracy required, depending on system specifications. The user needs to make sure that the executable `SLUSCHI` is alive in the background.

3.3. Outputs of SLUSCHI

The main output file is `SLUSCHI.out`. A sample of `SLUSCHI.out`, along with interpretations, is presented in Table 2.

The flowchart in Fig. 3 is closely followed by *SLUSCHI*, as well as by its main output `SLUSCHI.out`. Detailed directions to subdirectories can be found in `SLUSCHI.out`. The user may look into these subdirectories and check out the detailed outputs of `SLUSCHI.out`. Follow its directions to find more detailed output files named `log.out` in each directories, e.g.,

```
Dir_OptUnitCell Dir_VolSearch Dir_Melt Dir_CoexRun
```

The final MT, along with its standard error, can be found in the file `Dir_CoexRun/MP.out`, as well as in the main output `SLUSCHI.out` file.

4. Discussion

4.1. Systems studied

SLUSCHI has been studied and tested on more than ten systems and several dozen materials, including Al, Si, Ti (bcc), Ta (various pseudopotentials, under ambient pressure and 200 GPa, bcc), Na (30-120 GPa, bcc and fcc), NaCl, $La_2Zr_2O_7$ (La_2O_3-$2ZrO_2$, pyrochlore structure), Ru and its ternary alloys, the Hf-Ta-C-N systems, etc. Here we list in Table 3 the computational results with comparison to experiments, along with computational details. *SLUSCHI* can generally achieve an accuracy of around 100 K in MT calculations with our recommended settings.

Table 2: An example of `SLUSCHI.out` in the MT calculation of Ti (Ti_pv, 07Sep2000).

```
================================================================================================
Solid and Liquid in Ultra Small Coexistence with Hovering Interfaces (SLUSCHI)
by Qi-Jun Hong and Axel van de Walle
================================================================================================
*** Unit-cell optimization is running.  Please check it is still running in Dir_OptUnitCell.  ***
*** Please read Dir_OptUnitCell/log.out for details.  ***
=== Initial Lattice Vectors ===
......   (initial lattice vectors)                                          optunitcell is running
*** I find that unit-cell optimization completed.  ***
=== Optimized Lattice Vectors ===
......   (optimized lattice vectors)
================================================================================================
*** Generate a supercell from the current unitcell ***
The supercell is:
......   (supercell lattice vectors)
|a|,|b|,|c|,theta(bc),theta(ac),theta(ab):                                  solid.x is running
9.781 10.310 10.310 53.130 90.000 90.000
In UNIT-cell, number of atoms:  1 total:  1
In SUPER-cell, number of atoms:  48 total:  48
================================================================================================
*** Volume search is running.  Please check it is still running in Dir_VolSearch ***
*** Please read Dir_VolSearch/log.out for details.  ***
*** I find that the volume search completed.  ***
=== Initial Lattice Vectors ===                                          volsearch is running
......   (initial lattice vectors)
=== Optimized Lattice Vectors ===
......   (optimized lattice vectors)
================================================================================================
*** Preparing coexistence structures...  Please check it is still running in Dir_Melt ***
*** Please read Dir_Melt/log.out for details.  ***                        meltcoex is running
*** I find that the coexistence structure preparation completed.  ***
================================================================================================
*** Now running coexistence simulations...  ***
=== Run NPT MD at 1800 K on 1 snapshot(s) ===                             coexrun is running
--- RUNNING at 1800 K for snapshot ID 4000 ---
--- READ Dir_CoexRun/1800/1800/4000/log.out FOR DETAILS ---
*** SOLID in /home/users/hqj/Sluschi_Ti/sluschi/Dir_CoexRun/1800/1800/4000 ***
=== running MPFit ===
not enough data to fit a melting temperature.
1800 1 0 1                                   output format: T, n_solid, n_liquid, n_total
=== Find next job to run ===
next job:  1 MD duplicate(s) at 1900.0 K
...
=== running MPFit ===                        Melting temperature and standard error
Melting temperature and std error:  1952.47505249923 44.6557787995850
1600 4 0 4
1700 7 0 7
1800 3 1 4
1900 4 4 8
2000 7 9 16
2100 1 3 4
2200 0 4 4
MP accuracy is achieved.  Stopping code...
```

Table 3: Systems that have been studied using *SLUSCHI*

Systems	MT (K) GGA[a]	MT (K) HSE	Expt.	Pseudopotentials	`radius` (Å)	N^b	`kmesh`	$\sum n^c$	CPU Hours	Days[d]	Notes
Al	1040 ± 13	-	933	Al 04Jan2001	12	216	(1/2,1/2,1/2)	19	5,400	7	*SLUSCHI*[e]
Al	999 ± 21	1054			10	120	auto 30[f]	58	16,000	15	*SLUSCHI*[e]
Ti_v	1750 ± 25	1971		Ti 08Apr2002	10	96	auto 20[f]	26	7,900	17	*SLUSCHI*[e]
Ti_pv	1952 ± 45	-		Ti_pv 07Sep2000	10	96		48	20,200	20	*SLUSCHI*[e]
Si	724 ± 47	-			10	80	auto 10[f]	25	1,000	2	insufficient k-points
Si	1378 ± 24	1785	1687	Si 05Jan2001	10	80	auto 20[f]	19	7,500	15	*SLUSCHI*[e]
Si	1364 ± 37	-			12	132	auto 20[f]	20	27,700	27	compare well w/ Ref. [1]
Diamond, 10 GPa	4309 ± 50		4200-4300	C 08Apr2002	10	360	gamma[h]	45	283,000	111	*SLUSCHI*[e], in progress
Ru_v	2435 ± 32		2607	Ru 06Sep2000	10	152	auto 20[f]	30	88,000	37	*SLUSCHI*[e]
Ru_pv	2550 ± 34			Ru_pv 06Sep2000	10	152	(1/4,1/4,1/4)	33	139,000	51	*SLUSCHI*[e]
Ru50%-Re25% -W25%	2564 ± 40		n/a	Ru_pv,Re_pv,W_pv 06Sep2000	10	152	(1/4,1/4,1/4)	23	247,000	80	*SLUSCHI*[e]
Ta_v	2986 ± 41		3290	Ta 17Jan2003	10	108	(0,1/4,1/4)	38	32,000	23	early runs[e]
Ta_pv	3194 ± 40			Ta_pv 07Sep2000	10	108		38	54,000	24	early runs[e]
Ta_pv, 200 GPa	7953 ± 69		n/a	Ta_pv 07Sep2000	10	166	(1/4,1/4,1/4)	80	150,000	48	*SLUSCHI*[e], bcc [19]
Hf, bcc	2562 ± 31		2506	Hf 20Jan2003	10	108	(1/2,1/2,1/2)	115	14,900	12	*SLUSCHI*[e]
Hf, hcp	2122 ± 50		n/a	Hf 20Jan2003	10	128	(1/2,1/2,1/2)	50	25,800	21	*SLUSCHI*[e], not stable at MT
Na 15 GPa	657 ± 8	-	810/698[g]		10.4	108		55	47,000	24	bcc, early runs[e]
Na 26 GPa	750 ± 16	706	970[g]		9.8	108		52	26,000	17	bcc, early runs[e]
Na 40 GPa	742 ± 17	-	950[g]		9.4	108		74	54,000	37	bcc, early runs[e]
Na 55 GPa	716 ± 12	-	810[g]	Na_pv 05Jan2001	9	108	(0,1/4,1/4)	56	64,000	31	bcc, early runs[e]
Na 80 GPa	674 ± 20	-	700[g]		10.9	180		55	170,000	67	fcc, early runs[e]
Na 100 GPa	662 ± 14	-	450[g]		10.6	180		48	57,000	28	fcc, early runs[e]
Na 120 GPa	579 ± 27	-			10.4	180		88	150,000	66	fcc, early runs[e]
NaCl	1014 ± 18		1074	Na 08Apr2002 Cl 17Jan2003	11	128	gamma[h]	59	24,000	50	early runs[e]
$La_2Zr_2O_7$	2420 ± 27	2630	2530	Zr,O 08Apr2002 La 06Sep2000	10	176	gamma[h]	64	575,000	210	early runs[e], Ref. [17]
Hf-Ta-C-N				valence	10						see Ref. [18] for details

[a] If not specified, the PAW [41] and PBE [42] technique is employed.
[b] Number of atoms in the supercell.
[c] Total number of MD trajectories sampled.
[d] The physical time spent.
[e] "*SLUSCHI*" runs with higher efficiency compared to "early runs", which were performed at the early stage of method development.
[f] Automatic generation of k-points (see the `kmesh` tag in section 3.1.2).
[g] Experiment data are under dispute, e.g., see Ref. [47] vs. Ref. [48]. *SLUSCHI* results agree with Ref. [15] and [49] (theory).
[h] The use of Γ-point is automatically detected, as (1) `kmesh` is set to 10 for these insulators and (2) `$radius` $\geqslant 10$.

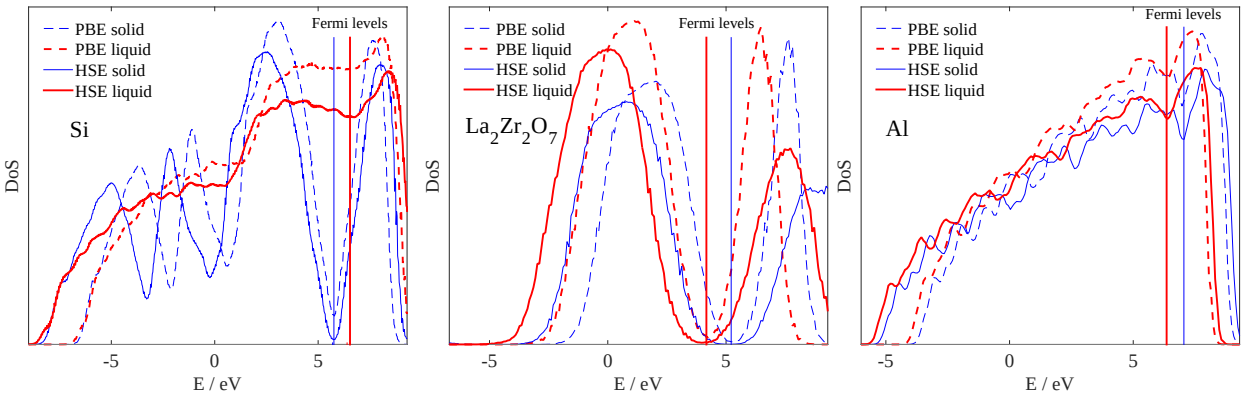

Figure 4: Electronic density of state for silicon, lanthanum zirconate ($La_2Zr_2O_7$) and aluminum, from left to right. While electronic structures mostly remain the same upon melting in the latter two, silicon undergoes drastic change upon melting: it becomes a typical conductor in the liquid state, compared to a typical semiconductor in its solid state.

4.2. Post-GGA correction

While generalized gradient approximations (GGA) are generally considered highly accurate, they have flaws especially in terms of "absolute error", due to the limitation in the density functionals. Fortunately, the absolute errors are similar in size and they tend to cancel out in solid-liquid phase transition calculations, so that the non-canceling error is small in most systems we had studied. However, we note that there are exceptions.

A typical example is silicon (see Table 3). The MT error is as large as 300 K, which amounts to almost 20% of the MT. Despite the large error, the calculated MT (1378±24 K) agrees closely with a previous calculation (1350 K) [1]. The latter adopts a different approach (which is based on free energy calculations, see Section 1) to compute the MT, but it achieves a similar DFT-level accuracy with local-density approximations and a norm-conserving pseudopotential [50]. The fact that both calculations result in similar errors strongly suggests that the drawback of DFT itself, rather than the finite-size effect, is responsible for the large error in MT calculation. Indeed, increasing the cell size does not improve the MT calculation (see Table 3).

Our study reveals a drastic change in electronic structure when silicon melts, as illustrated in the electronic density of states in Fig. 4. For example, liquid silicon is a good conductor, while solid-state silicon is a semiconductor. As the electronic structures are dramatically different, we do not expect the absolute errors in DFT to cancel with each other, and hence it becomes necessary to achieve an "absolute accuracy" in each phase. We find that a correction based on generally more accurate post-GGA methods (e.g., the Heyd-Scuseria-Ernzerhof (HSE) hybrid functional [46]) is required to help improve the description of electronic structures and the accuracy of the MT calculation.

We evaluate the HSE functional's impact on MT as an energy correction

$$\frac{T_m^{\mathrm{HSE}}}{T_m^{\mathrm{PBE}}} = \frac{\Delta H^{\mathrm{HSE}}}{\Delta H^{\mathrm{PBE}}}, \tag{9}$$

where ΔH is heat of fusion and T_m is MT. Though it is prohibitive to compute ΔH^{HSE} directly from MD, we evaluate it as an energy correction

$$H^{\mathrm{HSE}} - H^{\mathrm{PBE}} = \left\langle H^{\mathrm{HSE}} - H^{\mathrm{PBE}} \right\rangle_{H^{\mathrm{PBE}}}. \tag{10}$$

The ensemble average $\langle \cdots \rangle_{H^{\mathrm{PBE}}}$ is calculated based on randomly chosen snapshots from PBE MD trajectories. We then compute Eqn. (10) separately for the solid and the liquid phases to calculate the heat of fusion ΔH.

This feature is yet to be implemented in *SLUSCHI*. At this moment, the user needs to manually calculate the HSE correction.

4.3. Use of Γ-point

The Γ-version of *VASP* is twice faster. Hence in case that the Γ-point is sufficient to represent the Brillouin zone, the usage of Γ-version is highly recommended. Here we list several circumstances where

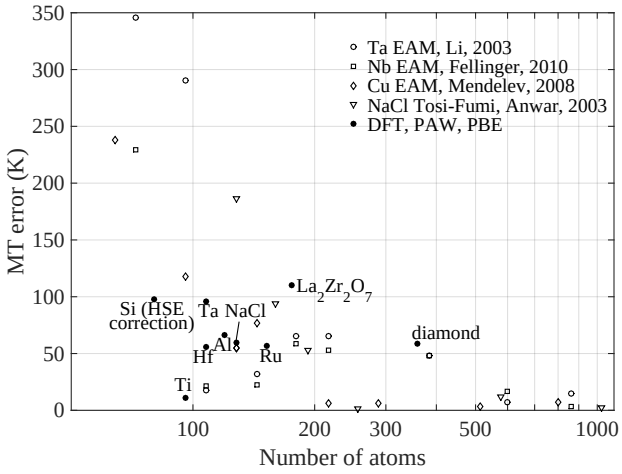

Figure 5: A summary of finite-size effect for various materials. This figure shows MT errors for all the materials studied so far, based on calculations from both empirical potentials (open markers) and DFT (solid dots). The MT benchmarks are from large-cell coexistence calculations (for empirical potentials) and experimental MT data (for DFT). 100K accuracy is generally achieved if the system contains more than 100 atoms.

the Γ-version is a good option to consider. (However, the user needs to be aware that the Γ-version may deteriorate accuracy. Therefore a careful check of the related error is required. We recommend users check convergence of energies and forces with respect to k-points.)

1. Large band-gap insulators. According to *VASP* manual, this type of materials have a relatively low demand on the number of k-points required in Brillouin zone sampling (typically 10 for `$kmesh`, which corresponds to Γ-point for a $2a \times a \times a (a = 10\text{Å})$ cell).
2. Low accuracy requirement. Depending on how accurate the MT calculation is needed, the user may consider the Γ-version after carefully checking the amount of corresponding error.
3. Exploring the *SLUSCHI* code. We recommend using the Γ-version for first-time *SLUSCHI* users. While an accurate and complete *SLUSCHI* run may take several days, the Γ-version can drastically reduce the physical time needed, so the user can focus on the execution of the code (however, the MT results may not be valid).
4. Indeed, the `meltcoex` code (which prepares half-half solid-liquid coexisting snapshots) employs the Γ-version, since high accuracy is not required.

4.4. Liquid-state free energy calculations

The calculated MT provides an avenue to liquid-state free energy (chemical potential) calculations. The user may first compute the solid-state free energy at the calculated MT by quasi-harmonic approximation. (Automated computer codes, such as the Automated Theoretic Automated Toolkit (ATAT) [51–55], are available.) For higher accuracy, the user may even use thermodynamic integration, from the quasi-harmonic potential (α) to real interactions in DFT MD (β), to account for the anharmonic effect.

$$G^{\beta} = G^{\alpha} + \int_0^1 \left\langle H_{\beta} - H_{\alpha} \right\rangle_{\lambda} \mathrm{d}\lambda$$

Then the free energy of the liquid is achieved, as the free energies are equal for the two phases at the meting temperature. After that, free energy at any point in the (T, P) space can be reached in principle by thermodynamic integration, as long as there is no phase transformation along the path.

$$H = \left(\frac{\partial (G/T)}{\partial (1/T)} \right)_P ,$$

$$V = \left(\frac{\partial G}{\partial P} \right)_T .$$

4.5. Finite-size effect

Based on our study using empirical EAM potentials in Ref. [16], we demonstrate that the small-cell coexistence method generally achieves 100K accuracy for MT calculations, if the cell contains ∼100 atoms and is larger than a $20 \times 10 \times 10 \text{Å}^3$ box. This recommendation is also supported by DFT calculations listed in Table 3. In Fig. 5 we summarize MT errors for all the materials we have studied so far, which provides a general picture of the finite-size effect. Even in the case of sodium chloride (NaCl) and the Hf-Ta-C-N system, which contain a considerable amount of long-range interaction (e.g., Coulomb interaction), following this guidance leads to satisfactory accuracy in MT calculations.

A small periodic cell has three major impacts. (1) It imposes a periodic constraint on the system. For the solid, the periodic condition limits the sampling of phonon frequencies. In particular, long-wavelength phonons are not well represented. For the liquid, the supercell enforces artificial periodicity. (2) Both the solid and the liquid are under the impact of solid-liquid interfaces. The solid is confined in between two liquid walls, and vice versa. (3) The densities of the solid and the liquid phases are different. Putting two phases in one small cell results in a mismatch of cell size.

We note that theoretically a sufficiently large supercell will in principle reduce the finite-size error and resolve this problem, while empirically the finite-size error is manageable following the `radius=10` and 100-atom rule, as we have discussed. Indeed, the finite-size effect declines quickly as the system size increases. The solid and the liquid are free to adjust volume along the direction perpendicular to the interface, though they are forced to share the same lattice vectors in parallel to the interface. This feature greatly relieves the stress caused by the different densities of the two phases. If we assume a bond length of 3 Å, there are at least three layers of neighbors between one atom and its periodic image in a $20 \times 10 \times 10$ Å^3 supercell (`radius=10`, which is the minimal cell size we recommend). Hence the structure has plenty of room to respond to the periodic constraint. When departing from an optimal state (which minimizes the Gibbs free energy of the single phase) to a coexistence state (which minimizes the Gibbs free energy of the coexistence), the Gibbs free energy of an individual phase increases (though the total free energy of the coexistence decreases). This holds true for both the solid and the liquid. Hence we expect a certain amount of error cancellation between the two phases.

Nevertheless, the finite-size effect is worth keeping in mind, and an increase in system size may be required under specific circumstances. We recommend follow the `radius=10` and 100-atom rule.

4.6. Quantum effect of vibrational free energy

The common version of molecular dynamics, which is based on the Newton's equations of motion, lacks a quantum-mechanical interpretation of lattice vibration. As this MD technique is heavily employed in *SLUSCHI*, MT calculations are hence subject to the drawback of this classical treatment of atomic motions. To include the quantum vibrational effect would require sophisticated path integral molecular dynamics methods [56, 57], which renders large-scale MD simulations infeasible.

While quantum effect is dominant at low temperatures, a quantum system starts to behave as if it is classical, with the increase of temperature. Since MT is usually much higher than Debye temperature, the classical treatment of atomic motion in MD is a rather good approximation for the study of melting properties, except for materials with very light elements (e.g., hydrogen) and strong quantum vibrational effect (e.g., water [58]). According to our study on copper [4], the difference between quantum and classical models is less than 1 meV in vibrational free energy near the MT.

4.7. Effect of thermostat

As *SLUSCHI* extensively runs NPT (or NVT) MD simulations, it is inevitably under the impact of thermostat, and thermostat artificially affects atomic motion. However, we note that the quantities that *SLUSCHI* samples (e.g., free energy difference as in Eq. 1) are thermodynamic properties rather than kinetic ones. Hence the impact from thermostat is generally considered small [59]. Furthermore, the Nose-Hoover thermostat [60–62] employed in *SLUSCHI* is capable of providing smooth and more realistic atomic movement, compared to the stochastic and sudden collision from the Anderson approach.

5. Conclusions

We have developed and distributed the *SLUSCHI* code, aiming to provide the community of researchers a robust and efficient tool for first-principles MT calculations. This manual provides a detailed instruction on how to use the code, how to interpret the outputs, along with several examples on the systems tested so far. A potential failure mode of the method is discussed and a solution provided. We hope researchers in the community find this code useful.

Acknowledgments

This research was supported by ONR under grants N00014-12-1-0196 and N00014-14-1-0055 and by Brown University through the use of the facilities at its Center for Computation and Visualization. This work uses the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575.

References

[1] O. Sugino and R. Car, Ab initio molecular dynamics study of first-order phase transitions: melting of silicon, Phys. Rev. Lett. **74**, 1823-1826 (1995).

[2] G. de Wijs, G. Kresse, and M. Gillan, First-order phase transitions by first-principles free-energy calculations: The melting of Al, Phys. Rev. B **57**, 8223-8234 (1998).

[3] D. Alfè, M. Gillan, and G. Price, The melting curve of iron at the pressures of the Earth's core from ab initio calculations, Nature (London) **401**, 462-464 (1999).

[4] Q.-J. Hong and A. van de Walle, Direct first-principles chemical potential calculations of liquids, J. Chem. Phys. **137**, 094114 (2012).

[5] D. Alfè, First-principles simulations of direct coexistence of solid and liquid aluminum, Phys. Rev. B **68**, 064423 (2003).

[6] T. Ogitsu, E. Schwegler, F. Gygi, and G. Galli, Melting of lithium hydride under pressure, Phys. Rev. Lett. **91**, 175502 (2003).

[7] D. Alfè, Melting curve of MgO from first-principles simulations, Phys. Rev. Lett. **94**, 235701 (2005).

[8] D. Alfè. Temperature of the inner-core boundary of the Earth: Melting of iron at high pressure from first-principles coexistence simulations, Phys. Rev. B **79**, 060101 (2009).

[9] A. B. Belonoshko, Molecular dynamics of MgSiO3 perovskite at high pressures: Equation of state, structure and melting transition. Geochim. et Cosmochim. Acta **58**, 4039-4047 (1994).

[10] A. B. Belonoshko, S. Davis, A. Rosengren, R. Ahuja, B. Johansson, S. I. Simak, L. Burakovsky, and D. L. Preston, Xenon melting: Density functional theory versus diamond anvil cell experiments, Phys. Rev. B **74**, 054114 (2006).

[11] J.-Y. Raty, E. Schwegler, and S. A. Bonev, Electronic and structural transitions in dense liquid sodium, Nature (London) **449**, 448-451 (2007).

[12] A. B. Belonoshko, S. Arapan, R. Martonak, and A. Rosengren, MgO phase diagram from first principles in a wide pressure-temperature range, Phys. Rev. B **81**, 054110 (2010).

[13] A. B. Belonoshko and A. Rosengren, High-pressure melting curve of platinum from ab initio Z method, Phys. Rev. B **85**, 174104 (2012).

[14] C. Zhang, L. Spanu, and G. Galli, Entropy of liquid water from ab initio molecular dynamics, J. Phys. Chem. B **115**, 14190-14195 (2011).

[15] M. P. Desjarlais, First-principles calculation of entropy for liquid metals, Phys. Rev. E **88**, 062145 (2013).

[16] Q.-J. Hong and A. van de Walle, Solid-liquid coexistence in small systems: A statistical method to calculate melting temperatures, J. Chem. Phys. **139**, 094114 (2013).

[17] Q.-J. Hong, S. V. Ushakov, A. Navrotsky, and A. van de Walle, Combined computational and experimental investigation of the refractory properties of $La_2Zr_2O_7$, Acta Mater. **84**, 275-282 (2015).

[18] Q.-J. Hong and A. van de Walle, Prediction of the material with highest known melting point from ab initio molecular dynamics calculations, Phys. Rev. B **92**, 020104 (2015).

[19] L. Miljacic, S. Demers, Q.-J. Hong and A. van de Walle, Equation of state of solid, liquid and gaseous tantalum from first principles, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **51**, 133-143 (2015).

[20] A. van de Walle and G. Ceder, The effect of lattice vibrations on substitutional alloy thermodynamics, Rev. Mod. Phys. **74**, 11-45 (2002).

[21] D. Kofke and P. Cummings, Quantitative comparison and optimization of methods for evaluating the chemical potential by molecular simulation, Mol. Phys. **92**, 973-996 (1997).

[22] J. Kirkwood, Statistical mechanics of fluid mixtures, J. Chem. Phys. **3**, 300-313 (1935).

[23] S. T. Lin, M. Blanco, and W. A. Goddard, The two-phase model for calculating thermodynamic properties of liquids from molecular dynamics: Validation for the phase diagram of Lennard-Jones fluids, J. Chem. Phys. **119**, 11792-11805 (2003).

[24] B. Widom, Some Topics in theory of fluids, J. Chem. Phys. **39**, 2808-2812 (1963).

[25] J. Mei and J. W. Davenport, Free-energy calculations and the melting point of Al, Phys. Rev. B **46**, 21-25 (1992).

[26] A. B. Belonoshko, N. V. Skorodumova, A. Rosengren, and B. Johansson, Melting and critical superheating, Phys. Rev. B **73**, 012201 (2006).

[27] D. Alfè, C. Cazorla, and M. J. Gillan, The kinetics of homogeneous melting beyond the limit of superheating, J. Chem. Phys. **135**, 024102 (2011).

[28] J. B. Haskins, J. A. Moriarty, and R. Q. Hood, Polymorphism and melt in high-pressure tantalum, Phys. Rev. B **86**, 224104 (2012).

[29] A. B. Belonoshko, T. Lukinov, L. Burakovsky, D. L. Preston, and A. Rosengren, Melting of a polycrystalline material, Eur. Phys. J. Spec. Top. **216**, 199-204 (2013).

[30] *SLUSCHI* is available at http://blogs.brown.edu/qhong/.

[31] P. Hohenberg and W. Kohn, Inhomogeneous electron gas, Phys. Rev. **136**, B864-871(1964).

[32] W. Kohn and L. J. Sham, Self-consistent equations including exchange and correlation effects, Phys. Rev. **140**, A1133-1138 (1965).

[33] R. Jones and O. Gunnarsson, The density functional formalism, its applications and prospects, Rev. Mod. Phys. **61**, 689-746 (1989).

[34] G. Kresse and J. Furthmüller, Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set, Comp. Mater. Sci. **6**, 15-50 (1996).

[35] G. Kresse and J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, Phys. Rev. B **54**, 11169-11186 (1996).

[36] G. Kresse and D. Joubert, From ultrasoft pseudopotentials to the projector augmented-wave method, Phys. Rev. B **59**, 1758-1775 (1999).

[37] R. A. Fisher, On the mathematical foundations of theoretical statistics, Phil. Trans. R. Soc. A **222**, 309-368 (1922).

[38] S. R. Eliason, *Maximum Likelihood Estimation: Logic and Practice*, (Sage, Newbury Park, CA, 1993).

[39] M. S. Daw and M. I. Baskes, Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals, Phys. Rev. B **29**, 6443-6453 (1984).

[40] Y. H. Li, D. J. Siegel, J. B. Adams, and X. Y. Liu, Embedded-atom-method tantalum potential developed by the force-matching method, Phys. Rev. B **67**, 125101 (2003).

[41] P. Blöchl, Projector augmented-wave method, Phys. Rev. B **50**, 17953-17979 (1994).

[42] J. P. Perdew, K. Burke, and M. Ernzerhof, Generalized gradient approximation made simple, Phys. Rev. Lett. **77**, 3865-3868 (1996).

[43] N. D. Mermin, Thermal properties of the inhomogeneous electron gas, Phys. Rev. **137**, A1441-1443 (1965).

[44] M. J. Gillan, Calculation of the vacancy formation energy in aluminium, J. Phys.: Condens. Matter **1**, 689-711 (1989).

[45] J. P. Perdew, J. A. Chevary, S. H. Vosko, K. A. Jackson, M. R. Pederson, D. J. Singh, and C. Fiolhais, Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation, Phys. Rev. B **46**, 6671-6687 (1992).

[46] J. Heyd, G. E. Scuseria, and M. Ernzerhof, Hybrid functionals based on a screened Coulomb potential, J. Chem. Phys. **118**, 8207-8215 (2003).

[47] C.-S. Zha and R. Boehler, Melting of sodium and potassium in a diamond anvil cell, Phys. Rev. B **31**, 3199-3201 (1985).

[48] E. Gregoryanz, O. Degtyareva, M. Somayazulu, R. J. Hemley, and H.-K. Mao, Melting of dense sodium, Phys. Rev. Lett. **94**, 185502 (2005).

[49] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, and M. Parrinello, Microscopic origins of the anomalous melting behavior of sodium under high pressure, Phys. Rev. Lett. **108**, 115701 (2012).

[50] G. B. Bachelet, D. R. Hamann, and M. Schlüter, Pseudopotentials that work: From H to Pu, Phys. Rev. B **26**, 4199-4228 (1982).

[51] A. van de Walle and G. Ceder, Automating First-Principles Phase Diagram Calculations, J. Phase Equilib. **23**, 348-359 (2002).

[52] A. van de Walle and M. Asta, Self-driven lattice-model Monte Carlo simulations of alloy thermodynamic properties and phase diagrams, Modelling Simul. Mater. Sci. Eng. **10**, 521-538 (2002).

[53] A. van de Walle, M. Asta and G. Ceder, The Alloy Theoretic Automated Toolkit: A User Guide, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **26**, 539-553 (2002).

[54] A. van de Walle, Multicomponent multisublattice alloys, nonconfigurational entropy and other additions to the Alloy Theoretic Automated Toolkit, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **33**, 266-278 (2009).

[55] A. van de Walle, P. Tiwary, M. M. de Jong, D. L. Olmsted, M. D. Asta, A. Dick, D. Shin, Y. Wang, L.-Q. Chen, and Z.-K. Liu, Efficient stochastic generation of Special Quasirandom Structures, CALPHAD: Comput. Coupling Phase Diagrams Thermochem. **42**, 13-18 (2013).

[56] R. P. Feynman and A. R. Hibbs, *Quantum Mechanics and Path Integrals* (McGraw-Hill, New York, 1965).

[57] D. Chandler and P. G. Wolynes, Exploiting the isomorphism between quantum theory and classical statistical mechanics of polyatomic fluids, J. Chem. Phys. **74**, 4078-4095 (1981).

[58] B. Pamuk, J. M. Soler, R. Ramirez, C. P. Herrero, P. W. Stephens, P. B. Allen, and M.-V. Fernandez-Serra, Anomalous nuclear quantum effects in ice, Phys. Rev. Lett. **108**, 193003 (2012).

[59] D. Frenkel and B. Smit, *Understanding Molecular Simulation* (Academic, San Diego, 1996).

[60] S. Nosé, A molecular-dynamics method for simulations in the canonical ensemble, Mol. Phys. **52**, 255-268 (1984).

[61] S. Nosé, A unified formulation of the constant temperature molecular-dynamics methods, J. Chem. Phys. **81**, 511-519 (1984).

[62] W. G. Hoover, Canonical dynamics - equilibrium phase-space distributions, Phys. Rev. A **31**, 1695-1697 (1985).