SAND2014-17598J

A Machine-Learning Method for Automated Generation of Quantum-Accurate Interatomic Potentials

A.P. Thompson[a,*], L.P. Swiler[b], C.R. Trott[c], S.M. Foiles[d], G.J. Tucker[d,e]

[a] *Multiscale Science Department, Sandia National Laboratories, P.O. Box 5800, MS 1322, Albuquerque, NM 87185*
[b] *Optimization and Uncertainty Quantification Department, Sandia National Laboratories, P.O. Box 5800, MS 1318, Albuquerque, NM 87185*
[c] *Scalable Algorithms Department, Sandia National Laboratories, P.O. Box 5800, MS 1322, Albuquerque, NM 87185*
[d] *Computational Materials and Data Science Department, Sandia National Laboratories, P.O. Box 5800, MS 1411, Albuquerque, NM 87185*
[e] *Materials Science and Engineering Department, Drexel University, Philadelphia, PA 19104*

Abstract

We present a new interatomic potential for solids and liquids called Spectral Neighbor Analysis Potential (SNAP). The SNAP potential has a very general form and uses machine-learning techniques to reproduce the energies, forces, and stress tensors of a large set of small configurations of atoms, which are obtained using high-accuracy quantum electronic structure (QM) calculations. The local environment of each atom is characterized by a set of bispectrum components of the local neighbor density projected on to a basis of hyperspherical harmonics in four dimensions. The bispectrum components are the same bond-orientational order parameters employed by the GAP potential [1]. The SNAP potential, unlike GAP, assumes a linear relationship between atom energy and bispectrum components. The linear SNAP coefficients are determined using weighted least-squares linear regression against the full QM training set. This allows the SNAP potential to be fit in a robust, automated manner to large QM data sets using many bispectrum coefficients.

*Corresponding author

Email addresses: athomps@sandia.gov (A.P. Thompson), lpswile@sandia.gov (L.P. Swiler), crtrott@sandia.gov (C.R. Trott), foiles@sandia.gov (S.M. Foiles), gtucker@coe.drexel.edu (G.J. Tucker)

The calculation of the bispectrum components and the SNAP potential are implemented in the LAMMPS parallel molecular dynamics code. We demonstrate that a previously unnoticed symmetry property can be exploited to reduce the computational cost of the force calculations by more than one order of magnitude. We present results for a SNAP potential for tantalum, showing that it accurately reproduces a range of commonly calculated properties of both the crystalline solid and the liquid phases. In addition, unlike simpler existing potentials, SNAP correctly predicts the energy barrier for screw dislocation migration in BCC tantalum.

Keywords: interatomic potential, machine learning, spectral neighbor analysis potential, SNAP, Gaussian approximation potentials, molecular dynamics

---

1. Introduction

Classical molecular dynamics simulation (MD) is a powerful approach for describing the mechanical, chemical, and thermodynamic behavior of solid and fluid materials in a rigorous manner [2]. The material is modeled as a large collection of point masses (atoms) whose motion is tracked by integrating the classical equations of motion to obtain the positions and velocities of the atoms at a large number of timesteps. The forces on the atoms are specified by an interatomic potential that defines the potential energy of the system as a function of the atom positions. Typical interatomic potentials are computationally inexpensive and capture the basic physics of electron-mediated atomic interactions of important classes of materials, such as molecular liquids and crystalline metals. Efficient MD codes running on commodity workstations are commonly used to simulate systems with $N = 10^5 - 10^6$ atoms, the scale at which many interesting physical and chemical phenomena emerge. Quantum molecular dynamics (QMD) is a much more computationally intensive method for solving a similar physics problem [3]. Instead of assuming a fixed interatomic potential, the forces on atoms are obtained by explicitly solving the quantum electronic structure of the valence electrons at each timestep. Because MD potentials are short-ranged, the computational complexity of MD generally scales as $O(N)$, whereas QMD calculations require global self-consistent convergence of the electronic structure, whose computational cost is $O(N_e^\alpha)$, where $2 < \alpha < 3$ and N_e is the number of electrons. For the same reasons, MD is amenable to

spatial decomposition on parallel computers, while QMD calculations allow only limited parallelism.

As a result, while high accuracy QMD simulations have supplanted MD in the range $N = 10-100$ atoms, QMD is still intractable for $N > 1000$, even using the largest supercomputers. Conversely, typical MD potentials often exhibit behavior that is inconsistent with QMD simulations. This has led to great interest in the development of MD potentials that match the QMD results for small systems, but can still be scaled to the interesting regime $N = 10^5 - 10^6$ atoms [1, 4, 5]. These quantum-accurate potentials require many more floating point operations per atom compared to conventional potentials, but they are still short-ranged. So the computational cost remains $O(N)$, but with a larger algorithm pre-factor.

In this paper, we present a new quantum-accurate potential called SNAP. It is designed to model the migration of screw dislocations in tantalum metal under shear loading, the fundamental process underlying plastic deformation in body-centered cubic metals. In the following section we explain the mathematical structure of the potential and the way in which we fit the potential parameters to a database of quantum electronic structure calculations. We follow that with a brief description of the implementation of the SNAP potential in the LAMMPS code. We demonstrate that a previously unnoticed symmetry property can be exploited to reduce the computational cost of the force calculations by more than one order of magnitude. We then present results for the SNAP potential that we have developed for tantalum. We find that this new potential accurately reproduces a range of properties of solid and liquid tantalum. Unlike simpler potentials, it correctly matches quantum MD results for the screw dislocation core structure and minimum energy pathway for displacement of this structure, properties that were not included in the training database.

2. Mathematical Formulation

2.1. Bispectrum coefficients

The quantum mechanical principle of near-sightedness tells us that the electron density at a point is only weakly affected by atoms that are not near. This provides support for the common assumption that the energy of a configuration of atoms is dominated by contributions from clusters of atoms that are near to each other. It is reasonable then to seek out descriptors of local structure and build energy models based on these descriptors.

Typically, this is done by identifying geometrical structures, such as pair distances and bond angles, or chemical structures, such as bonds. Interatomic potentials based on these approaches often produce useful qualitative models for different types of materials, but it can be difficult or impossible to adjust these potentials to accurately reproduce known properties of specific materials. Recently, Bartók et al. have studied several infinite classes of descriptor that are related to the density of neighbors in the spherically symmetric space centered on one atom [1, 6, 7]. They demonstrated that by adding descriptors of successively higher order, it was possible to systematically reduce the mismatch between the potential and the target data. One of these descriptors, the bispectrum of the neighbor density mapped on to the 3-sphere, forms the basis for their Gaussian Approximation Potential (GAP) [1]. We also use the bispectrum as the basis for our SNAP potential. We derive this bispectrum below, closely following the notation of Ref.[7].

The density of neighbor atoms around a central atom i at location $\mathbf{r}$ can be considered as a sum of δ-functions located in a three-dimensional space:

$$\rho_i(\mathbf{r}) = \delta(\mathbf{r}) + \sum_{r_{ii'}<R_{cut}} f_c(r_{ii'}) w_{i'} \delta(\mathbf{r} - \mathbf{r}_{ii'}) \tag{1}$$

where $\mathbf{r}_{ii'}$ is the vector joining the position of the central atom i to neighbor atom i'. The $w_{i'}$ coefficients are dimensionless weights that are chosen to distinguish atoms of different types, while the central atom is arbitrarily assigned a unit weight. The sum is over all atoms i' within some cutoff distance R_{cut}. The switching function $f_c(r)$ ensures that the contribution of each neighbor atom goes smoothly to zero at R_{cut}. The angular part of this density function can be expanded in the familiar basis of spherical harmonic functions $Y_m^l(\theta, \phi)$, defined for $l = 0, 1, 2, \ldots$ and $m = -l, -l+1, \ldots, l-1, l$ [8]. The radial component is often expanded in a separate set of radial basis functions that multiply the spherical harmonics. Bartók et al. made a different choice, mapping the radial distance r on to a third polar angle θ_0 defined by,

$$\theta_0 = \theta_0^{max} \frac{r}{R_{cut}} \tag{2}$$

The additional angle θ_0 allows the set of points (θ, ϕ, r) in the 3D ball of possible neighbor positions to be mapped on to the set of points (θ, ϕ, θ_0) that are a subset of the 3-sphere. Points south of the latitude θ_0^{max} are excluded. It is advantageous to use most of the 3-sphere, while still excluding the

region near the south pole where the configurational space becomes highly compressed.

The natural basis for functions on the 3-sphere is formed by the 4D hyperspherical harmonics $U^j_{m,m'}(\theta_0, \theta, \phi)$, defined for $j = 0, \frac{1}{2}, 1, \ldots$ and $m, m' = -j, -j+1, \ldots, j-1, j$ [8]. These functions also happen to be the elements of the unitary transformation matrices for spherical harmonics under rotation by angle $2\theta_0$ about the axis defined by (θ, ϕ). When the rotation is parameterized in terms of the three Euler angles, these functions are better known as $D^j_{m,m'}(\alpha, \beta, \gamma)$, the Wigner D-functions, which form the representations of the $SO(3)$ rotational group [9, 8]. Dropping the atom index i, the neighbor density function can be expanded in the $U^j_{m,m'}$ functions

$$\rho(\mathbf{r}) = \sum_{j=0,\frac{1}{2},\ldots}^{\infty} \sum_{m=-j}^{j} \sum_{m'=-j}^{j} u^j_{m,m'} U^j_{m,m'}(\theta_0, \theta, \phi) \tag{3}$$

where the expansion coefficients are given by the inner product of the neighbor density with the basis function. Because the neighbor density is a weighted sum of δ-functions, each expansion coefficient can be written as a sum over discrete values of the corresponding basis function,

$$u^j_{m,m'} = U^j_{m,m'}(0,0,0) + \sum_{r_{ii'} < R_{cut}} f_c(r_{ii'}) w_{i'} U^j_{m,m'}(\theta_0, \theta, \phi) \tag{4}$$

The expansion coefficients $u^j_{m,m'}$ are complex-valued and they are not directly useful as descriptors, because they are not invariant under rotation of the polar coordinate frame. However, the following scalar triple products of expansion coefficients can be shown to be real-valued and invariant under rotation [7].

$$B_{j_1,j_2,j} = \sum_{m_1,m_1'=-j_1}^{j_1} \sum_{m_2,m_2'=-j_2}^{j_2} \sum_{m,m'=-j}^{j} (u^j_{m,m'})^* H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'} \tag{5}$$

The constants $H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}}$ are coupling coefficients, analogous to the Clebsch-Gordan coefficients for rotations on the 2-sphere. These invariants are the components of the bispectrum. They characterize the strength of density correlations at three points on the 3-sphere. The lowest-order components

describe the coarsest features of the density function, while higher-order components reflect finer detail. An analogous bispectrum can be defined on the 2-sphere in terms of the spherical harmonics. In this case, the components of the bispectrum are a superset of the second and third order bond-orientational order parameters developed by Steinhardt et al. [10]. These in turn are specific instances of the order parameters introduced in Landau's theory of phase transitions [11].

The coupling coefficients are non-zero only for non-negative integer and half-integer values of j_1, j_2, and j satisfying the conditions $\|j_1 - j_2\| \leq j \leq j_1 + j_2$ and $j_1 + j_2 - j$ not half-integer [9]. In addition, $B_{j_1,j_2,j}$ is symmetric in j_1 and j_2. Hence the number of distinct non-zero bispectrum coefficients with indices j_1, j_2, j not exceeding a positive integer J is $(J+1)^3$. Furthermore, it is proven in the appendix that bispectrum components with reordered indices are related by the following identity:

$$\frac{B_{j_1,j_2,j}}{2j+1} = \frac{B_{j,j_2,j_1}}{2j_1+1} = \frac{B_{j_1,j,j_2}}{2j_2+1}. \tag{6}$$

We can exploit this equivalence by further restricting $j_2 \leq j_1 \leq j$, in which case the number of distinct bispectrum coefficients drops to $(J+1)(J+2)(J+\frac{3}{2})/3$, a three-fold reduction in the limit of large J.

2.2. SNAP Potential Energy Function

Given the bispectrum components as descriptors of the neighborhood of each atom, it remains to express the potential energy of a configuration of N atoms in terms of these descriptors. We write the energy of the system containing N atoms with positions $\mathbf{r}^N$ as the sum of a reference energy E_{ref} and a local energy E_{local}

$$E(\mathbf{r}^N) = E_{ref}(\mathbf{r}^N) + E_{local}(\mathbf{r}^N). \tag{7}$$

The reference energy includes known physical phenomena, such as long-range electrostatic interactions, for which well-established energy models exist. E_{local} must capture all the additional effects that are not accounted for by the reference energy. Following Bartók et al. [1, 7] we assume that the local energy can be decomposed into separate contributions for each atom,

$$E_{local}(\mathbf{r}^N) = \sum_{i=1}^{N} E_i(\mathbf{q}_i) \tag{8}$$

where E_i is the local energy of atom i, which depends on the set of descriptors $\mathbf{q}_i$, in our case the set of K bispectrum components $\mathbf{B}^i = \{B_1^i, \ldots, B_K^i\}$. The original GAP formulation of Bartók et al. [1] expressed the local energy in terms of a Gaussian process kernel. For the materials that we have examined so far, we have found that energies and forces obtained from quantum electronic structure calculations can be accurately reproduced by linear contributions from the lowest-order bispectrum components, with linear coefficients that depend only on the chemical identity of the central atom:

$$E_{SNAP}^i(\mathbf{B}^i) = \beta_0^{\alpha_i} + \sum_{k=1}^{K} \beta_k^{\alpha_i} B_k^i = \beta_0^{\alpha_i} + \boldsymbol{\beta}^{\alpha_i} \cdot \mathbf{B}^i \tag{9}$$

where α_i is the chemical identity of atom i and β_k^α are the linear coefficients for atoms of type α. Hence the problem of generating the interatomic potential has been reduced to that of choosing the best values for the linear SNAP coefficients. Since our goal is to reproduce the accuracy of quantum electronic structure calculations for materials under a range of conditions, it makes sense to select SNAP coefficients that accurately reproduce quantum calculations for small configurations of atoms representative of these conditions.

In quantum methods such as density functional theory [3] the most readily computed properties are total energy, atom forces, and stress tensor. The linear form of the SNAP energy allows us to write all of these quantities explicitly as linear functions of the SNAP coefficients. We restrict ourselves here to the case of atoms of a single type, but the results are easily extended to the general case of multiple atom types. In the case of total energy, the SNAP contribution can be written in terms of the bispectrum components of the atoms

$$E_{SNAP}(\mathbf{r}^N) = N\beta_0 + \boldsymbol{\beta} \cdot \sum_{i=1}^{N} \mathbf{B}^i \tag{10}$$

where $\boldsymbol{\beta}$ is the K-vector of SNAP coefficients and β_0 is the constant energy contribution for each atom. $\mathbf{B}^i$ is the K-vector of bispectrum components for atom i. The contribution of the SNAP energy to the force on atom j can be written in terms of the derivatives of the bispectrum components w.r.t. $\mathbf{r}_j$, the position of atom j

$$\mathbf{F}^j_{SNAP} = -\nabla_j E_{SNAP} = -\boldsymbol{\beta} \cdot \sum_{i=1}^{N} \frac{\partial \mathbf{B}^i}{\partial \mathbf{r}_j}, \tag{11}$$

where $\mathbf{F}^j_{SNAP}$ is the force on atom j due to the SNAP energy. Finally, we can write the contribution of the SNAP energy to the stress tensor

$$\mathbf{W}_{SNAP} = -\sum_{j=1}^{N} \mathbf{r}_j \otimes \nabla_j E_{SNAP} = -\boldsymbol{\beta} \cdot \sum_{j=1}^{N} \mathbf{r}_j \otimes \sum_{i=1}^{N} \frac{\partial \mathbf{B}^i}{\partial \mathbf{r}_j} \tag{12}$$

where $\mathbf{W}_{SNAP}$ is the contribution of the SNAP energy to the stress tensor and $\otimes$ is the Cartesian outer product operator.

All three of these expressions consist of the vector $\boldsymbol{\beta}$ of SNAP coefficients multiplying a vector of quantities that are calculated from the bispectrum components of atoms in a configuration. This linear structure greatly simplifies the task of finding the best choice for $\boldsymbol{\beta}$. We can define a system of linear equations whose solution corresponds to an optimal choice for $\boldsymbol{\beta}$, in that it minimizes the sum of square differences between the above expressions and the corresponding quantum results defined for a large number of different atomic configurations. This is described in more detail in the following section.

3. Automated Potential Generation Methodology

The previous section outlined the SNAP formulation. In practice, one needs to determine the values of the SNAP coefficients, $\boldsymbol{\beta}$. This section presents how we solve for the K-vector $\boldsymbol{\beta}$ of SNAP coefficients using a least-squares formulation.

3.1. Formulation of the Linear Least Squares Problem

The fitting problem is overdetermined, in the sense that the number of data points that we are fitting to far exceeds the number of SNAP coefficients. The cost of evaluating the bispectrum components $B_{j_1,j_2,j}$ increases strongly with the order of the indices j, j_1, and j_2. For this reason, K is limited to the range $10 - 100$. In contrast, with the availability of high-performance computers and highly optimized electronic structure codes, it is not difficult to generate data for hundreds or thousands of configurations of atoms. Note that we refer to a *configuration* as a set of atoms located at

particular positions in a quantum mechanical calculation. In most cases, the atoms define an infinite repeating structure with specified periodic lattice vectors. For a particular configuration s, containing N_s atoms, the electronic structure calculation yields $3N_s + 7$ data values: the total energy, the $3N_s$ force components, and the 6 independent components of the stress tensor. The same quantities are calculated for the reference potential. In addition, the bispectrum components and derivatives for each atom in the configuration are calculated. We can use all of this data to construct the following set of linear equations.

$$\begin{bmatrix} \vdots & \vdots \\ N_s & \sum_{i=1}^{N_s} \mathbf{B}^i \\ \vdots & \vdots \\ 0 & -\sum_{i=1}^{N_s} \frac{\partial \mathbf{B}^i}{\partial r_j^\alpha} \\ \vdots & \vdots \\ 0 & -\sum_{j=1}^{N_s} r_j^\alpha \sum_{i=1}^{N_s} \frac{\partial \mathbf{B}^i}{\partial r_j^\beta} \\ \vdots & \vdots \end{bmatrix} \cdot \begin{bmatrix} \beta_0 \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \vdots \\ E_s^{qm} - E_s^{ref} \\ \vdots \\ F_{j,\alpha}^{qm} - F_{j,\alpha}^{ref} \\ \vdots \\ W_{\alpha\beta,s}^{qm} - W_{\alpha\beta,s}^{ref} \\ \vdots \end{bmatrix} \tag{13}$$

This matrix formulation is of the type $\mathbf{A} \cdot \boldsymbol{\beta} = \mathbf{y}$ which can be solved for the coefficients $\boldsymbol{\beta}$. The optimal solution $\hat{\boldsymbol{\beta}}$ for this set of equations is [12]:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \|(\mathbf{A} \cdot \boldsymbol{\beta} - \mathbf{y})\|^2 = \mathbf{A}^{-1} \cdot \mathbf{y} \tag{14}$$

In practice, we do not explicitly take the inverse of the $\mathbf{A}$ matrix, but instead use a QR factorization to solve for $\boldsymbol{\beta}$. We have found the linear solve to obtain the optimal SNAP coefficients to be very fast and not poorly conditioned.

We have also added the capability to perform weighted least squares to weight certain rows more than others. That is, we add a vector of weights $\mathbf{w}$ to the minimization formulation:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{w} \circ (\mathbf{A} \cdot \boldsymbol{\beta} - \mathbf{y})\|^2 \tag{15}$$

where $\circ$ is used to denote element by element multiplication by the weight vector. Thus, each row in the $\mathbf{A}$ matrix and the $\mathbf{y}$ vector are multiplied by a weight specified for that row. In this way, we are able to specify weights per configuration type (e.g. BCC crystals, liquids, etc.) and per quantity of

interest (e.g. energy, force, stress tensor). We have found that the ability to weight different rows in the **A** matrix is critical to ensure the regression works well. One reason is that the total energy, forces, and stress components can vary considerably in relative magnitude, depending on what units they are expressed in. However, the more important reason is that it is desirable to control the relative influence of different configurations, depending on the material properties that are of greatest importance. For example, if we want the SNAP potential to more accurately reproduce BCC elastic constants, we can increase the weight on the stress components of strained BCC configurations. We also found it helpful to convert all extensive quantities to intensive quantities, in order to counteract overweighting of configurations with large N_s. Total energy rows were scaled by the number of atoms and stress tensor components rows were scaled by the cell volume.

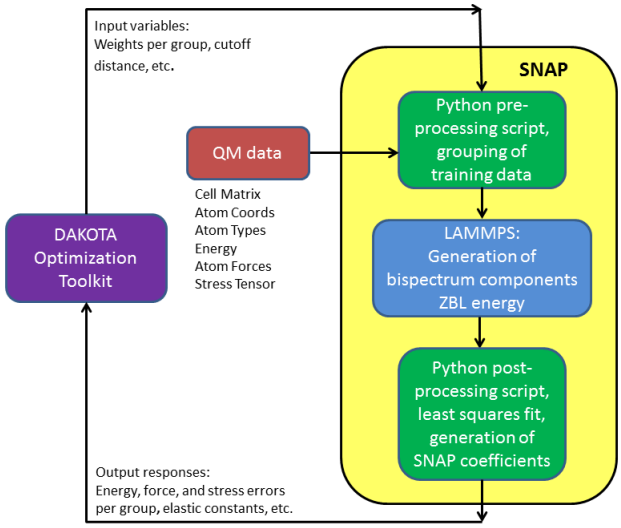

Figure 1: Flowchart of the optimization loop around the generation of the SNAP potential in LAMMPS

3.2. SNAP software implementation

We describe the Python framework to generate the SNAP fit within the LAMMPS software tool. We start with quantum mechanical (QM) training data, generated from ab initio calculations, which can be obtained from a wide variety of quantum electronic structure software packages. The training

data is first converted to a set of files, one per configuration, using standardized JavaScript Object Notation (JSON) format. The JSON files contain all relevant information such as atom coordinates, atom types, periodic cell dimensions, energies, forces, and stress tensors. Note that the performance of the SNAP potential will depend on the comprehensiveness of the configurations in the training set. In the example of tantalum below, we are interested in material plasticity and stability as well as elastic constants and the lattice parameter. For this reason, the training data included configurations for all the important crystal structures, generalized stacking faults, free surfaces, liquid structures, and randomly deformed primitive cells of the ground-state crystal.

Once the training data is generated, pre-processing and post-processing scripts must be run to generate the SNAP potential in the LAMMPS software. The pre-processing script converts the training data from the JSON format into the native LAMMPS input format. LAMMPS is then used to generate the bispectrum components for the training data configurations, as well as calculating the reference potential. The LAMMPS output is used to generate the energy, force and stress tensor rows of the A matrix defined in Eq. 13. The LAMMPS output for the reference potential is combined with the training data to generate the righthand side vector, which is the difference between the training data and the reference potential. A standard linear algebra library is used to perform QR factorization, yielding the least-squares optimal SNAP linear coefficients, $\hat{\boldsymbol{\beta}}$. The steps involved in the generation of the SNAP potential are shown in the yellow box on the right side of Fig. 1.

3.3. Optimization of hyperparameters governing the SNAP potential

Each SNAP potential developed in this manner depends on the particular values chosen for a set of "hyperparameters." These include things such as the weights for each training configuration, the list of bispectrum components to be calculated, and the cutoff distance defining the neighborhood of an atom. It is not obvious how to choose values of these hyperparameters which will lead to an "optimal" SNAP potential in the sense of minimizing the SNAP prediction error with respect to energy errors, force errors, or other quantities.

To determine the optimal hyperparameters governing the SNAP potential, we have used an optimization framework and placed it around the SNAP calculation to generate many instances of SNAP potentials. The optimization framework we use is the DAKOTA software [13], which is a toolkit of

optimization and uncertainty quantification methods designed to interface to scientific computing codes. The process of generating a SNAP potential within an optimization loop to optimize the governing parameters is shown in Fig. 1. DAKOTA varies input parameters such as the weights per configuration group and the cutoff distance. These values are then specified in the SNAP generation and the SNAP potential is calculated. Once the SNAP coefficients are generated, they are used to predict the energy and forces of the QM training data. The errors in the SNAP prediction (defined as the difference between SNAP configuration energies vs. QM configuration energies, SNAP forces vs. QM forces, etc.) are then aggregated into an objective function which is returned to DAKOTA and used as the quantity which DAKOTA tries to optimize.

Using this optimization framework, we can identify the SNAP potential that has the "best" result, according to minimizing an objective function. We have examined various objective functions. Currently the objective function we use involves a weighted sum of the errors with respect to energies, forces, and stress tensors, as well as errors with respect to the elastic constants. The objective function is not trivial to define: the resulting SNAP potential can be sensitive to which error measures are weighted more in the objective function.

4. Implementation

A detailed account of the SNAP implementation and its optimization for specific computing platforms is given elsewhere [14]. The force on each atom due to the SNAP potential is formally given by Eq. 11. In order to perform this calculation efficiently, we use a neighbor list, as is standard practice in the LAMMPS code [15, 16]. This list identifies all the neighbors of a given atom i. In order to avoid negative and half-integer indices, we have switched notation from $u^j_{m,m'}$ to $u^\eta_{\mu,\mu'}$, where $\eta = 2j$, $\mu = m + j$, and $\mu' = m' + j$. Analogous transformations are used for $H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}}$ and $B_{j_1,j_2,j}$. This also allows us to reclaim the symbol j for indexing the neighbor atoms of atom i. Finally, boldface symbols with omitted indices such as $\mathbf{u}_i$ are used to indicate a finite multidimensional arrays of the corresponding indexed variables.

Fig. 2 gives the resulting force computation algorithm, where Calc_U() calculates all expansion coefficients $u^\eta_{\mu,\mu'}$ for an atom i while Calc_Z() calculates the partial sums $Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta}$, which are defined as

$$Z^{m,m'}_{j_1,j_2,j} = \sum_{m_1,m_1'=-j_1}^{j_1} \sum_{m_2,m_2'=-j_2}^{j_2} H^{j,m,m'}_{j_1,m_1,m_1',j_2,m_2,m_2'} u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'}. \tag{16}$$

In the loop over neighbors, first the derivatives of $\mathbf{u}_i$ with respect to the distance vector between atoms i and j are computed in Calc_dUdR() and then the derivatives of B^i are computed in Calc_dBdR(), which is the most computationally expensive part of the algorithm. For the parameter sets used in this study, Calc_dBdR() is responsible for approximately 90% of all floating point and memory operations. Thus, we concentrate our description on this function. In Fig. 3 we show the original algorithm based on Eq. A.9. In Fig. 4 we show the improved algorithm based on Eq. A.10 that takes advantage of the symmetry relation Eq. 6. The elimination of the two innermost nested loops reduces the scaling of the computational complexity of the bispectrum component from $O(J^4)$ to $O(J^2)$ where J is the upper limit on j_1, j_2, and j.

4.1. Scaling Results

Both of these algorithms have been implemented in the LAMMPS parallel molecular dynamics package [15, 16]. In Fig. 5 we compare the performance of the old and new algorithms. Timings are based on a 10,000 step MD simulations of BCC tantalum crystal using the SNAP potential described in Section 5. The calculations were performed on Sandia's Chama high-performance cluster with a dual socket Intel Sandy Bridge processor with 16 cores on each node. Three different system sizes were used, containing 512,

```
Compute_SNAP():
for i in natoms() {
    u_i = Calc_U(i)
    Z_i = Calc_Z(i,u_i)
    for j in neighbors(i) {
        ∇_j u_i = Calc_dUdR(i, j,u_i)
        ∇_j B^i = Calc_dBdR(i, j,u_i,Z_i,∇_j u_i)
        F_ij = −β · ∇_j B^i
        F_i += −F_ij; F_j += F_ij
}   }
```

Figure 2: Base algorithm for the SNAP force calculation.

Function Calc_dBdR(i, j):

for (η, η_1, η_2) in GetBispectrumIndices() {

$\quad \nabla_j B_{\eta_1,\eta_2,\eta} = 0$

$\quad$ for $(\mu = 0; \mu \leq \eta; \mu{+}{+}\ )$ {

$\quad\quad$ for $(\mu' = 0; \mu' \leq \eta; \mu'{+}{+})$ {

$\quad\quad\quad \nabla_j Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta} = 0$

$\quad\quad\quad$ for $(\mu_1 = \max(0, \mu + (\eta_1 - \eta_2 - \eta)/2);$

$\quad\quad\quad\quad \mu_1 \leq \min(\eta_1, \mu + (\eta_1 + \eta_2 - \eta)/2); \mu_1{+}{+})$ {

$\quad\quad\quad\quad$ for $(\mu'_1 = \max(0, \mu' + (\eta_1 - \eta_2 - \eta)/2);$

$\quad\quad\quad\quad\quad \mu'_1 \leq \min(\eta_1, \mu' + (\eta_1 + \eta_2 - \eta)/2); \mu'_1{+}{+})$ {

$\quad\quad\quad\quad \mu_2 = \mu - \mu_1\ ;\ \mu'_2 = \mu' - \mu'_1$

$\quad\quad\quad\quad \nabla_j Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta} \mathrel{+}= H^{\eta,\mu,\mu'}_{\eta_1,\mu_1,\mu'_1,\eta_2,\mu_2,\mu'_2}$

$\quad\quad\quad\quad\quad (u^{\eta_1}_{\mu_1,\mu'_1} \nabla_j u^{\eta_2}_{\mu_2,\mu'_2} + u^{\eta_2}_{\mu_2,\mu'_2} \nabla_j u^{\eta_1}_{\mu_1,\mu'_1})$

$\quad\quad\quad$ } }

$\quad\quad\quad \nabla_j B_{\eta_1,\eta_2,\eta} \mathrel{+}= (u^{\eta}_{\mu,\mu'})^* \nabla_j Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta} + Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta} (\nabla_j u^{\eta}_{\mu,\mu'})^*$

} } }

Figure 3: Original algorithm for the derivative of the bispectrum components of atom i w.r.t. the position of atom j using Eq. A.9.

4096, and 32768 atoms. Each system size was run on 8, 16, 32, 64, 128, and 256 nodes. We plot the time required to calculate one MD time step versus the number of atoms per node. In this form, results for the three different system sizes are almost indistinguishable, indicating that both single node performance and strong scaling efficiency are determined primarily by the number of atoms per node. When the number of atoms per node is large, the parallel scaling is close to ideal, and the improved algorithm is consistently about 16x faster than the old algorithm. As the number of atoms per node decreases, this margin also decreases, due to the parallel efficiency of the new algorithm decreasing more rapidly. This is seen more clearly in Fig. 6 which shows the parallel efficiency of each calculation relative to the timing for the largest system on 8 nodes. Both the old and new algorithms show a decrease in parallel efficiency as the number of atoms per node decreases, but this happens at an earlier point for the new algorithm, because there is a lot less computation per atom, while the amount of communication is unchanged.

Function Calc_dBdR(i, j):

for (η, η_1, η_2) in GetBispectrumIndices() {

$\quad \nabla_j B_{\eta_1,\eta_2,\eta} = 0$

$\quad$ for $(\mu = 0; \mu \leq \eta; \mu{+}{+}\ )$ {

$\quad\quad$ for $(\mu' = 0; \mu' \leq \eta; \mu'{+}{+})$ {

$\quad\quad\quad \nabla_j B_{\eta_1,\eta_2,\eta} \mathrel{+}= Z^{\mu,\mu'}_{\eta_1,\eta_2,\eta} (\nabla_j u^{\eta}_{\mu,\mu'})^*$

$\quad$ } }

$\quad$ for $(\mu_1 = 0; \mu_1 \leq \eta_1; \mu_1{+}{+}\ )$ {

$\quad\quad$ for $(\mu'_1 = 0; \mu'_1 \leq \eta_1; \mu'_1{+}{+})$ {

$\quad\quad\quad \nabla_j B_{\eta_1,\eta_2,\eta} \mathrel{+}= \frac{\eta+1}{\eta_1+1} Z^{\mu_1,\mu'_1}_{\eta,\eta_2,\eta_1} (\nabla_j u^{\eta_1}_{\mu_1,\mu'_1})^*$

$\quad$ } }

$\quad$ for $(\mu_2 = 0; \mu_2 \leq \eta_2; \mu_2{+}{+}\ )$ {

$\quad\quad$ for $(\mu'_2 = 0; \mu'_2 \leq \eta_2; \mu'_2{+}{+})$ {

$\quad\quad\quad \nabla_j B_{\eta_1,\eta_2,\eta} \mathrel{+}= \frac{\eta+1}{\eta_2+1} Z^{\mu_2,\mu'_2}_{\eta_1,\eta,\eta_2} (\nabla_j u^{\eta_2}_{\mu_2,\mu'_2})^*$

$\quad$ } }

}

Figure 4: Improved algorithm for the derivative of the bispectrum components of atom i w.r.t. the position of atom j using Eq. A.10.

5. SNAP Potential for Tantalum

5.1. Training Data

The training set data as well as the validation data for these potentials were computed using density functional theory (DFT) electronic structure calculations as implemented in the Vienna Ab initio Simulation Package (VASP) [17]. The pseudopotentials employed are of the projector augmented wave (PAW) form [18] with the exchange correlation (XC) energy evaluated using the Perdew, Burke and Ernzerhof (PBE) formulation [19] of the generalized gradient approximation (GGA). The pseudopotential employed treats 11 electrons ($5p5d6s$) as valent. Compared to the use of pseudopotentials with 5 valence states, this improved the prediction of the elastic constants and is expected to be important in geometries with highly compressed bond lengths. A plane-wave cut-off energy of 500 eV was used to insure the convergence of the computed stresses and forces in addition to the energies. The k-space integrations were performed using the Monkhorst-Pack (MK) integration scheme [20] with mesh sizes that depended on the particular geometry

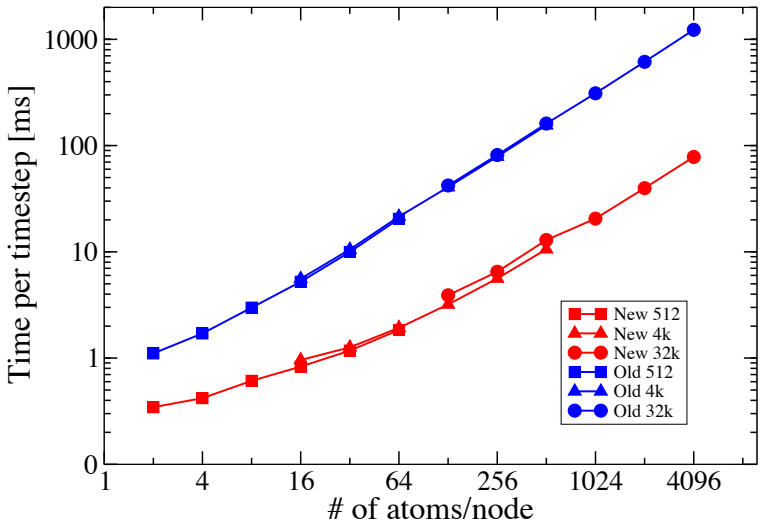

Figure 5: CPU time per MD time step versus atoms per node for benchmark simulations of BCC tantalum using the old and improved implementations of the SNAP potential. Results are shown for systems containing 512, 4096, and 32768 atoms. The number of nodes ranged from 8 to 256 nodes.

and chosen to obtain convergence of the energy, forces and stress tensor. The lattice predicted for Ta is 0.332 nm compared to the experimental value of 0.331 nm. The computed elastic constants, C_{11}, C_{12} and C_{44} are 267.5, 159.7, and 71.1 GPa. This compares well with the experimental results of 266.3, 158.2, and 87.4 GPa by Featherston and Neighbours [21] and 266, 161.1, and 82.5 GPa by Katahara et al. [22].

A variety of different types of atomic configurations were used to construct the full set of training data, and these are summarized in Table 1. Configurations of different types were chosen to adequately sample the important regions of the potential energy surface. Configurations of type "Displaced" were constructed by randomly displacing atoms from their equilibrium lattice sites in supercells of the A15, BCC, and FCC crystal structures. The maximum displacement in any direction was limited to 10% of the nearest neighbor distance. Configurations of type"Elastic" were constructed by applying random strains to primitive cells of the BCC crystal. The maximum

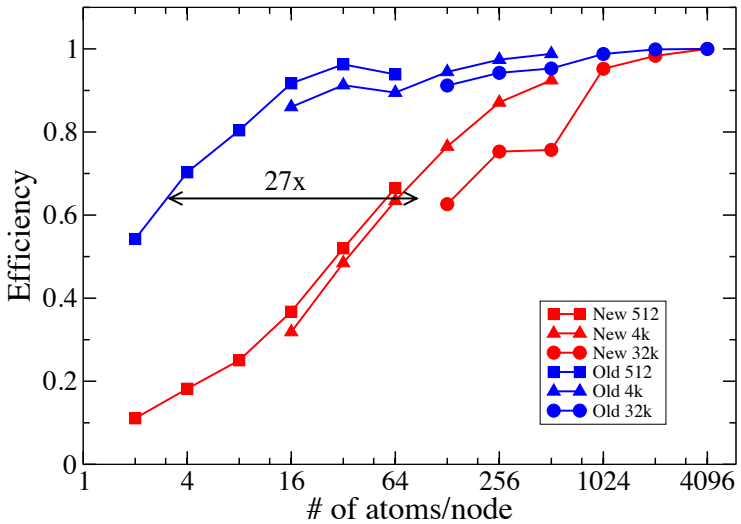

Figure 6: Parallel efficiency versus atoms per node for benchmark simulations of BCC tantalum using the old and improved implementations of the SNAP potential. Results are shown for systems containing 512, 4096, and 32768 atoms. The number of nodes ranged from 8 to 256 nodes.

linear strains were limited to 0.5%. The configurations of type "GSF" consist of both relaxed and unrelaxed generalized stacking faults on the (110) and (112) crystallographic planes in the $\langle 111 \rangle$ direction. The configurations of type "Liquid" were taken from a high-temperature quantum molecular dynamics simulations of molten tantalum. The configurations of type "Surface" consisted of relaxed and unrelaxed (100), (110), (111), and (112) BCC surfaces. For each type of configuration we specified a weight for the energy, force, and stress. We set the force and energy weights of the"Elastic" configurations to zero and we set the stress weights of all other configurations to zero.

In addition to the training data and way in which different quantities were weighted, the quality of the SNAP potential also was somewhat dependent on the choices made for the reference potential and the SNAP hyperparameters. Because the training data did not sample highly compressed configurations, it was important that the reference potential provide a good physical descrip-

Type	N_{conf}	N_{atoms}	Energy	Force	Stress
Displaced A15	9	64	100	1	-
Displaced BCC	9	54	100	1	-
Displaced FCC	9	48	100	1	-
Elastic BCC	100	2	-	-	0.0001
GSF 110	22	24	100	1	-
GSF 112	22	30	100	1	-
Liquid	3	100	100	1	-
Surface	7	30	100	1	-

Table 1: DFT data used to fit the SNAP potential for Tantalum

tion of Pauli repulsion that dominates the interaction at close separation. We chose the Ziegler-Biersack-Littmark (ZBL) empirical potential that has been found to correctly correlate the high-energy scattering of ions with their nuclear charge Z_{zbl} [23]. Because the ZBL potential decays rapidly with radial separation, we used a switching function to make the energy and force go smoothly to zero at a distance $R_{zbl,o}$, while leaving the potential unchanged for distances less than $R_{zbl,i}$. The values for these three parameters are given in Table 2. For the neighbor density switching function, we use the same functional form as Bartók et al. [1].

$$f_c(r) = \frac{1}{2}(\cos(\pi r/R_{cut}) + 1), r \leq R_{cut} \quad (17)$$

$$= 0, r > R_{cut} \quad (18)$$

The DAKOTA package was used to optimize the value of R_{cut} so as to minimize the error in the energies, forces, and elastic constants relative to the training data. The resultant value of $R_{cut} = 4.67637$ is physically reasonable, as it includes the 14 nearest neighbors in the BCC crystal, and the first coordination shell in the melt. The effect of using fewer or more bispectrum components was examined experimentally by varying J. We found that the fitting errors decreased monotonically with increasing J, but the marginal improvement also decreased. We found that truncating at $J = 3$ provided a good trade off between accuracy and computational efficiency. The full set of ZBL and SNAP parameters values are given in Table 2, while the values of the SNAP linear coefficients corresponding to each bispectrum component

J	3
R_{cut}	4.67637 Å
θ_0^{max}	0.99363π
$R_{zbl,i}$	4.0 Å
$R_{zbl,o}$	4.8 Å
Z_{zbl}	73.0

Table 2: SNAP and ZBL potential parameters used to model tantalum. [15]

are listed in Table 3. [15]

5.2. Validation Results

One of the crucial requirements for interatomic potential models is that they predict the correct minimum energy crystal structure and that the energetics of competing crystal structures be qualitatively correct. Fig. 7 plots the energy per atom computed with the SNAP potential as a function of volume for the BCC, FCC, A15, and HCP phases. The energy of diamond structure Ta was also computed. As expected, it was found to be substantially ($\sim$2.8 eV/atom) higher than the BCC phase and is not included on the plot. In addition, energies computed from density functional theory are included as crosses. It is seen that the relative energy of these phases is correctly predicted. The BCC phase is the most stable throughout the volume range considered, with the A15 phase somewhat higher. The FCC phase is next with a minimum energy about 0.2 eV/atom above that of BCC. Note that these energy differences are very consistent with the DFT calculations. The SNAP predicted HCP energy is also shown. Note that the SNAP potential is able to differentiate HCP and FCC crystal structures which are structurally very similar. The SNAP potential predicts that the HCP structure is higher in energy than the FCC structure. This is a prediction in that no HCP data was used in the potential construction. The relative energies of HCP and FCC are in agreement with our DFT calculations, which show that lowest energy HCP structure (not shown) lies 0.04 eV/atom above the minimum energy FCC structure. Further, the SNAP potential predicts the HCP c/a ratio to be 1.72, which is considerably greater than ideal value $c/a \approx 1.63$. The DFT calculations for HCP (not shown) predict an even larger value of $c/a = 1.77$.

k	$2j_1$	$2j_2$	$2j$	β_k
0				-2.92477
1	0	0	0	-0.01137
2	1	0	1	-0.00775
3	1	1	2	-0.04907
4	2	0	2	-0.15047
5	2	1	3	0.09157
6	2	2	2	0.05590
7	2	2	4	0.05785
8	3	0	3	-0.11615
9	3	1	4	-0.17122
10	3	2	3	-0.10583
11	3	2	5	0.03941
12	3	3	4	-0.11284
13	3	3	6	0.03939
14	4	0	4	-0.07331
15	4	1	5	-0.06582
16	4	2	4	-0.09341
17	4	2	6	-0.10587
18	4	3	5	-0.15497
19	4	4	4	0.04820
20	4	4	6	0.00205
21	5	0	5	0.00060
22	5	1	6	-0.04898
23	5	2	5	-0.05084
24	5	3	6	-0.03371
25	5	4	5	-0.01441
26	5	5	6	-0.01501
27	6	0	6	-0.00599
28	6	2	6	-0.06373
29	6	4	6	0.03965
30	6	6	6	0.01072

Table 3: SNAP linear coefficients for tantalum. [15]

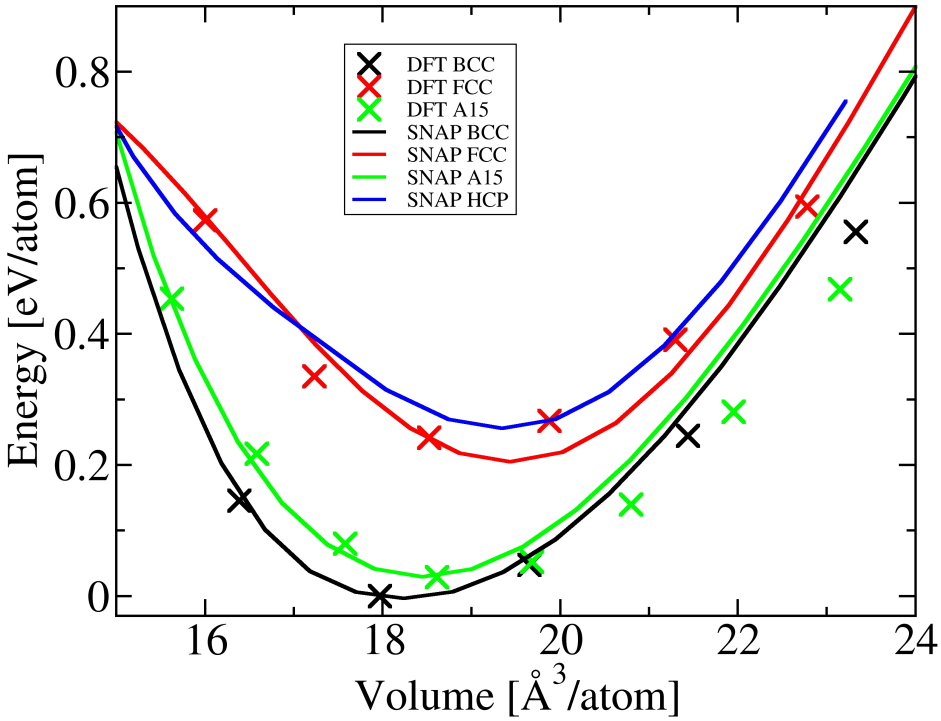

Figure 7: Energy versus volume for various crystal structures as indicated in the legend, as computed by SNAP (solid curves) and from DFT (x).

5.3. Melting point and Liquid structure

The melting point predicted by the SNAP potential has been determined. An atomistic slab was created and brought to temperature above the melting point using a Langevin thermostat until the surface of the slab was melted. The molecular dynamics simulation was then continued in an NVE ensemble. This resulted in a system containing two solid-liquid interfaces. The temperature of the MD system now fluctuated around the equilibrium melting temperature. It is important that the solid phase be at the correct melting point density. This was ensured through a simple iterative procedure. An estimate of the melting point was obtained for an assumed lattice constant, the lattice constant of the solid at that temperature was determined from an NPT simulation of the solid, and the melting point was determined with the interfacial area determined by this lattice constant. This process was iterated until the assumed and predicted melting points agreed. This

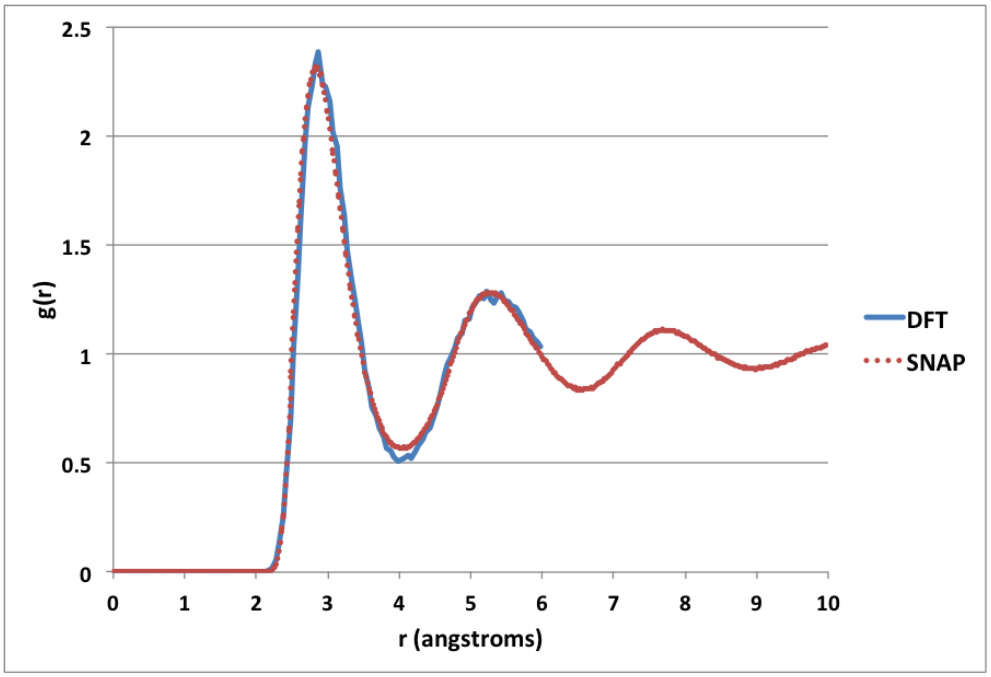

Figure 8: Comparison of pair correlation function for molten tantalum calculated using DFT (blue) and SNAP (red)

procedure predicted a melting point of 2790 K. This value is in reasonable agreement with the experimental melting point of 3293 K. It should be noted that the melting point is typically a difficult quantity for interatomic potentials to predict accurately. Further, the comparison with experiment is not a direct test of the agreement of the SNAP potential with DFT calculations. The experimental value reflects contributions of the free energy of electronic excitations. Further, the DFT prediction for the melting point is not known.

While configurations that correspond to molten Ta were used in the training set, it is important to determine if the potentials actually reproduce the correct distribution of spatial density correlations in the liquid state. The liquid is an important test of potential models since it samples configurations that are far from those of the equilibrium solid crystals. In particular, the liquid structure depends strongly on the repulsive interactions that occur when two atoms approach each other. We calculated the pair correlation function of the liquid, $g(r)$, both using DFT and from the SNAP potential. These simulations were performed for the same temperature, 3250 K, and

atomic density of 49.02 atom/nm^3. The DFT simulation treated 100 atoms for a period of 2 ps while the SNAP simulations considered a cell containing 1024 atoms and averaged over 500 ps. Fig. 8 compares $g(r)$ obtained in the two simulations. The agreement is excellent except perhaps in the region of the first minimum. Note that there is substantially more statistical uncertainty in the DFT result due to the short simulation time and the DFT structure can not be determined beyond about 0.6 nm, due to the smaller simulation cell. These results indicate that the SNAP potential provides a good representation of the molten structure.

5.4. Planar and Point Defects

A key metric for the applicability of an interatomic potential model is its ability to describe common defects in a material. Table 4 presents a comparison of the energy associated with surfaces, unstable stacking faults, vacancies and self-interstitial atoms. The results obtained from the SNAP potential are compared to our DFT calculations and also against two other interatomic potential models, the embedded atom method (EAM) model developed by Zhou et al. [24] and the angular dependent potential (ADP) due to Mishin and Lozovoi [25].

The training data includes the surfaces and unstable stacking faults presented in Table 4. The unstable stacking fault energy is the maximum of the energy associated with translating half of the crystal in order to create a stacking fault on the specified plane. This energy is expected to be related to the structure and behavior of dislocations. The overall magnitude of the surface energies is in good agreement with the DFT data. The SNAP potential correctly predicts that the (110) surface is the lowest energy surface plane though it does not correctly predict the relative energies of the other higher energy surface planes. The surface energies predicted by the SNAP potential are generally somewhat better than those predicted by the EAM and ADP potentials. All of the potentials predict unstable stacking fault energies less than the DFT results though all of the potentials predict that the unstable stacking fault energy on the (112) plane is higher than on the (110) plane.

The point defect energies were not included in the training data for the SNAP potential. The predicted vacancy formation energy is 0.15 eV lower than the density function value. The EAM and ADP potentials are in better agreement with the DFT value. However, both of those potentials were also explicitly fit to the vacancy formation energy. The formation energy for self-interstitials are presented for four different configurations. Self-interstitial

	DFT	SNAP	EAM	ADP
Lattice Constant (Å)	3.320	3.316	3.303	3.305
B (Mbar)	1.954	1.908	1.928	1.971
$C' = \frac{1}{2}(C_{11} - C_{12})$ (Mbar)	50.7	59.6	53.3	51.0
C_{44} (Mbar)	75.3	73.4	81.4	84.6
Vacancy Formation Energy (eV)	2.89	2.74	2.97	2.92
(100) Surface Energy (J/m^2)	2.40	2.68	2.34	2.24
(110) Surface Energy (J/m^2)	2.25	2.34	1.98	2.13
(111) Surface Energy (J/m^2)	2.58	2.66	2.56	2.57
(112) Surface Energy (J/m^2)	2.49	2.60	2.36	2.46
(110) Relaxed Unstable SFE (J/m^2)	0.72	1.14	0.75	0.58
(112) Relaxed Unstable SFE (J/m^2)	0.84	1.25	0.87	0.74
Self-Interstitial − Octahedral Site (eV)	6.01	7.10	5.06	7.61
Self-Interstitial − Crowdion (eV)	4.73	5.74	5.09	7.02
Self-Interstitial − $\langle 100 \rangle$ dumbbell (eV)	6.12	6.89	5.24	7.59
Self-Interstitial − $\langle 110 \rangle$ dumbbell (eV)	5.63	5.43	4.93	6.99

Table 4: Calculated lattice constant, elastic constants, and formation energies for various crystal surfaces and defects using the SNAP, EAM [24] , and ADP [25] potentials, compared to DFT calculations.

energies and preferred geometries are challenging quantities for interatomic potential models since they typically involve bond lengths which are substantially shorter than equilibrium values. The overall values for the SNAP potential are in reasonable accord with the DFT, but they incorrectly indicate that the $\langle 110 \rangle$ dumbbell is more favorable than the Crowdion configuration predicted by the DFT calculations.

5.5. Dislocations

Fig. 9 shows the screw dislocation migration energy barrier calculated using DFT and the SNAP, EAM [24] , and ADP [25] potentials. One of the dominant deformation mechanisms for metallic materials is the motion of dislocations. For the case of body-centered cubic materials such as tantalum, the screw dislocations are known to play a crucial role. The structure and motion of screw dislocations in BCC metals has been examined for many years and is discussed in detail by Gröger et al. [26] and by references therein. A crucial feature of screw dislocations in BCC metals is the Peierls

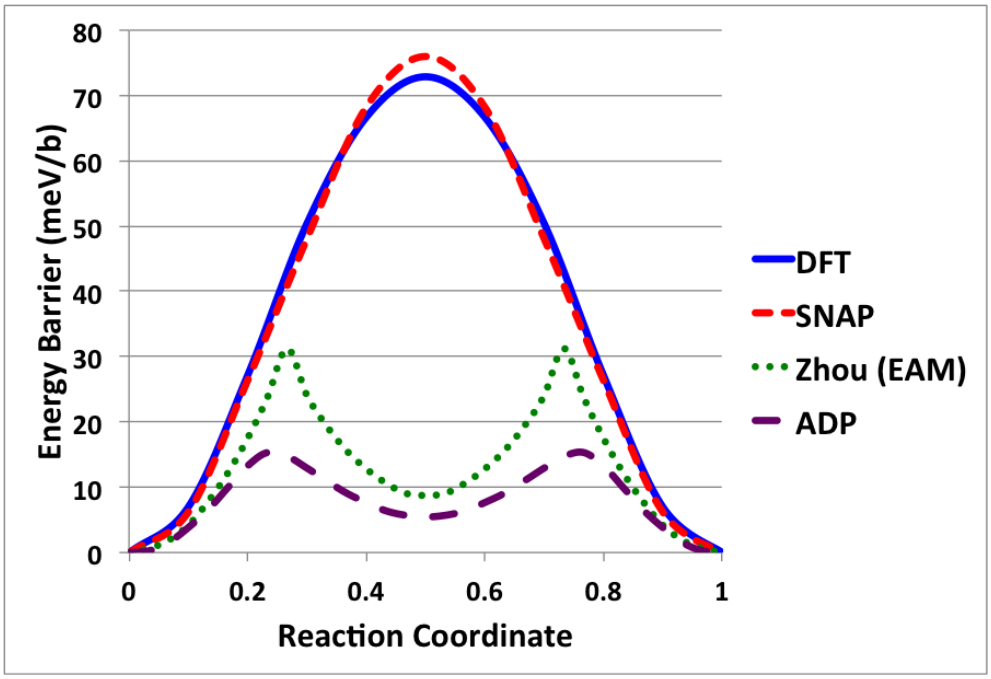

Figure 9: Comparison of screw dislocation migration energy barrier calculated using DFT and the SNAP, EAM [24], and ADP [25] potentials.

barrier which is the energy barrier to move the dislocation to its next stable configuration. Unlike face-centered-cubic metals where the Peierls barrier is generally negligible, the barrier in the case of BCC metals is substantial and plays a significant role in mechanical deformation. As has been shown recently by Weinberger et al. [27], many empirical potentials for BCC metals predict qualitatively incorrect Peierls barriers. DFT calculations show that the Peierls has a simple shape with a single hump while many empirical potentials predict a transition path with two maxima and a metastable intermediate state. The Peierls barrier computed via the SNAP potential is shown in Fig. 9 along with the DFT barrier and the barriers predicted by the EAM and ADP potentials. In all cases the barriers were computed based on a dislocation dipole configuration as described by Weinberger et al. [27]. While the ADP and EAM potentials both predict incorrect barriers with an intermediate metastable state, the SNAP potential predicts a barrier with a single maximum in agreement with the DFT results. Further, the magnitude of the barrier is in excellent agreement with the DFT prediction.

6. Summary

In this paper we have introduced a new class of interatomic potentials based on the 4D bispectrum components first proposed by Bartók et al. [1]. Our SNAP potentials differ from Bartók's GAP potentials primarily in the use of an explicit linear dependence of the energy on the bispectrum components. We have developed a powerful machine-learning methodology for fitting SNAP potentials to large data sets of high-accuracy quantum electronic structure calculations. We have demonstrated the effectiveness of this approach for tantalum. The new SNAP potential accurately represents a wide range of energetic properties of solid tantalum phases, as well as both the structure of molten tantalum and its melting point. Most importantly, we have found that the SNAP potential for tantalum correctly predicts the size and shape of the Peierls barrier for screw dislocation motion in BCC tantalum. This is a critical property for describing plasticity in tantalum under shear loading and is not correctly described by other published potentials for tantalum [24, 25].

One possible drawback of the SNAP methodology is computational cost when used in large-scale atomistic simulations. Even with the algorithmic improvements described in this paper, the cost of SNAP is one to two orders of magnitude more expensive than its competitors. However, it is important to note the high computational intensity also makes the method very amenable to massively parallel algorithms. For a fixed size problem, the SNAP potential has been shown to scale to a much greater extent than simpler potentials [14]. We anticipate that atomistic simulations of materials behavior using high-accuracy computationally-intensive potentials such as SNAP will become more common as access to petascale computing resources increases.

Acknowledgement

The authors acknowledge helpful discussions with Stan Moore and Jonathan Moussa on the symmetry properties of bispectrum components. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

[1] A. Bartók, M. C. Payne, K. Risi, G. Csányi, Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons, Physical Review Letters 104 (2010) 136403.

[2] M. Griebel, S. Knapek, G. Zumbusch, Numerical Simulation in Molecular Dynamics, Springer Verlag, Heidelberg Germany, 2007.

[3] A. E. Mattsson, P. A. Schultz, M. P. Desjarlais, T. R. Mattsson, K. Leung, Designing meaningful density functional theory calculations in material science–A primer, Modelling Simul. Mater. Sci. Eng. 13 (2005) R1–R31.

[4] N. Artrith, J. Behler, High-dimensional neural network potentials for metal surfaces: A prototype study for copper, Physical Review B 85 (2012) 045439.

[5] Y. Li, D. J. Siegel, J. B. Adams, X.-Y. Liu, Embedded-atom-method tantalum potential developed by the force-matching method, Physical Review B 67 (2003) 125101.

[6] A. Bartók-Partay, The Gaussian Approximation Potential: An Interatomic Potential Derived from First Principles Quantum Mechanics, Springer Theses, 2010.

[7] A. Bartók, M. J.Gillan, F. R. Manby, G. Csányi, On representing chemical environments, Physical Review B 87 (2013) 184115.

[8] D. A. Varshalovich, A. N. Moskalev, V. K. Khersonskii, Quantum Theory of Angular Momentum, World Scientific, Singapore, 1987.

[9] A. V. Meremianin, Multipole expansions in four-dimensional hyperspherical harmonics, Journal Of Physics A-Mathematical And General 39 (2006) 3099–3112.

[10] P. J. Steinhardt, D. R. Nelson, M. Ronchetti, Bond-Orientational Order In Liquids And Glasses, Physical Review B 28 (1983) 784–805.

[11] L. D. Landau and E. M. Lifshitz, Course of Theoretical Physics, Volume 5, Statistical Physics, Part 1, Elsevier Butterworth Heinemann, Oxford, 1980.

[12] G. Strang, Linear Algebra and Its Applications, Academic Press, New York, NY, 1980.

[13] B. M. Adams, K. R. Dalbey, M. S. Eldred, D. M. Gay, L. P. Swiler, W. J. Bohnhoff, J. P. Eddy, K. Haskell, DAKOTA users manual version 5.1, Sandia Technical Report SAND2010-2183, Sandia National Laboratories, Albuquerque, NM, 2011.

[14] C. R. Trott, S. D. Hammond, A. P. Thompson, SNAP: Strong scaling high fidelity molecular dynamics simulations on leadership-class computing platforms, in: Proceedings of the 30th International Supercomputing Conference, ISC 2013, Leipzig, Germany, June 22-26, 2014., volume 7905 of *Lecture Notes in Computer Science*, Springer Verlag, 2014.

[15] The SNAP implementation in LAMMPS and the input files for the SNAP tantalum potential have been released in the open-source distribution of LAMMPS available at http://lammps.sandia.gov, 2014.

[16] S. Plimpton, Fast Parallel Algorithms For Sort-Range Molecular-Dynamics, J. Comput. Phys 117 (1995) 1–19.

[17] G. Kresse, J. Furthmller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, Physical Review B 54 (1996) 11169–11186.

[18] G. Kresse, D. Joubert, From ultrasoft pseudopotentials to the projector augmented-wave method, Physical Review B 59 (1999) 1758–1775.

[19] J. P. Perdew, K. Burke, M. Ernzerhof, Generalized Gradient Approximation Made Simple, Physical Review Letters 77 (1996) 3865–3868.

[20] H. J. Monkhorst, J. Pack, Special points for Brillouin-zone integrations, Physical Review B 13 (1976) 5188–5192.

[21] F. H. Featherston, J. Neighbours, Elastic Constants of Tantalum, Tungsten and Molybdenum, Physical Review 130 (1963) 1324–1333.

[22] K. W. Katahara, M. H. Manghnani, E. S. Fisher, Pressure derivatives of the elastic moduli of niobium and tantalum, Journal of Applied Physics 47 (1976) 434–439.

[23] J. Ziegler, J. P. Biersack, U. Littmark, The Stopping and Range of Ions in Matter 1 (1985).

[24] X. W. Zhou, R. A. Johnson, H. N. G. Wadley, Misfit-energy-increasing dislocations in vapor-deposited cofe/nife multilayers, Phys. Rev. B 69 (2004) 144113.

[25] Y. Mishin, A. Lozovoi, H. Wadley, Angular-dependent interatomic potential for tantalum, Acta Materialia 54 (2006) 5013–5026.

[26] R. Gröger, A. G. Bailey, V. Vitek, Multiscale modeling of plastic deformation of molybdenum and tungsten: I. Atomistic studies of the core structure and glide of 1/2 ¡ 111 ¿ screw dislocations at 0 K, Acta Materialia 56 (2008) 5401–5411.

[27] C. R. Weinberger, G. J. Tucker, S. M. Foiles, Peierls potential of screw dislocations in bcc transition metals: Predictions from density functional theory, Physical Review B 87 (2013) 054114.

[28] K. Kadau, T. Germann, P. Lomdahl, Molecular Dynamics Comes of Age: 320 Billion Atom Simulation on BlueGene/L, International Journal of Modern Physics C 17 (2006) 1755–1761.

[29] J. N. Glosli, D. F. Richards, K. J. Caspersen, R. E. Rudd, J. A. Gunnels, F. H. Streitz, Extending Stability Beyond CPU Millennium: A Micron-scale Atomistic Simulation of Kelvin-Helmholtz Instability, in: Proceedings of the 2007 ACM/IEEE Conference on Supercomputing, SC '07, ACM, New York, NY, USA, 2007, pp. 58:1–58:11.

[30] S. Swaminarayan, T. Germann, K. Kadau, G. Fossum, 369 Tflop/s Molecular Dynamics Simulations on the Roadrunner General-Purpose Heterogeneous Supercomputer (2008) 1–10.

[31] T. Hamada, T. Narumi, R. Yokota, K. Yasuoka, K. Nitadori, M. Taiji, 42 TFlops Hierarchical N-body Simulations on GPUs with Applications in Both Astrophysics and Turbulence, in: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis, SC '09, ACM, New York, NY, USA, 2009, pp. 62:1–62:12.

[32] L.-W. Wang, B. Lee, H. Shan, Z. Zhao, J. Meza, E. Strohmaier, D. H. Bailey, Linearly scaling 3d fragment method for large-scale electronic

structure calculations, in: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing, SC '08, IEEE Press, Piscataway, NJ, USA, 2008, pp. 65:1–65:10.

[33] G. Goedecker, Linear Scaling Electronic Structure Methods, Rev. Mod. Phys 71 (1999) 1085.

[34] A. V. Meremianin, Hyperspherical harmonics with arbitrary arguments, Journal Of Mathematical Physics 50 (2009).

Appendix A. Proof of additional symmetry relation on bispectrum components

The bispectrum components for an arbitrary function defined on the 3-sphere are given by a sum over triple products of the expansion coefficients

$$B_{j_1,j_2,j} = \sum_{m,m'}^{j} (u^j_{m,m'})^* \sum_{m_1,m_1'}^{j_1} \sum_{m_2,m_2'}^{j_2} H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'} \tag{A.1}$$

where $u^j_{m,m'}$ are expansion coefficients, given by the inner product with the corresponding basis function $U^j_{m,m'}$. The constants $H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}}$ are the Clebsch-Gordan coupling coefficients that can be written as products of the more well-known Clebsch-Gordan coefficients for functions on the 2-sphere

$$H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} = C^{j,m}_{j_1,m_1,j_2,m_2} C^{j,m'}_{j_1,m_1',j_2,m_2'} \tag{A.2}$$

It follows from several elementary symmetry properties of the Clebsch-Gordan coefficients [8] that coupling coefficients for which the indices j_2 and j are interchanged satisfy the following identity

$$\frac{H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}}}{(2j+1)(-1)^{j+m}(-1)^{j+m'}} = \frac{H^{j_2,-m_2,-m_2'}_{\substack{j_1,m_1,m_1' \\ j,-m,-m'}}}{(2j_2+1)(-1)^{j_2-m_2}(-1)^{j_2-m_2'}} \tag{A.3}$$

Similarly, the 4D hyperspherical harmonics satisfy the following symmetry sign-inversion relation [8]

$$U^j_{m,m'} = (-1)^{m'-m} (U^j_{-m,-m'})^* \tag{A.4}$$

It follows from linearity of the inner product that the same relation holds for the expansion coefficients $u^j_{m,m'}$. Substituting both of these symmetry relations in Eq. A.1 gives,

$$B_{j_1,j_2,j} = (-1)^\epsilon \frac{2j+1}{2j_2+1} \sum_{m,m'}^{j} u^j_{-m,-m'} \sum_{m_1,m_1'}^{j_1} \sum_{m_2,m_2'}^{j_2} H^{j_2,-m_2,-m_2'}_{j_1,m_1,m_1'\ j,-m,-m'} \; u^{j_1}_{m_1,m_1'} (u^{j_2}_{-m_2,-m_2'})^* \tag{A.5}$$

where the parity exponent ϵ is given by the following expression in integer and half-integer indices

$$\begin{aligned} \epsilon &= (m'-m) + (m_2'-m_2) + (j+m) + (j+m') + (j_2-m_2) + (j_2-m_2') \\ &= 2(j+m') + 2(j_2-m_2) \end{aligned} \tag{A.6}$$

Both of the quantities in parentheses are integers for all values of j, m', j_2, and m_2 and so the factor $(-1)^\epsilon$ is unity. Finally, reversing the order of the summations over m_2, m_2', m, and m' yields

$$B_{j_1,j_2,j} = \frac{2j+1}{2j_2+1} \sum_{m_2,m_2'}^{j_2} (u^{j_2}_{m_2,m_2'})^* \sum_{m_1,m_1'}^{j_1} \sum_{m,m'}^{j} H^{j_2 m_2 m_2'}_{j_1 m_1 m_1'\ j m m'} \; u^{j_1}_{m_1,m_1'} u^j_{m,m'} \tag{A.7}$$

Comparison with the original Eq. A.1 we see that the nested sum on the right-hand side is B_{j_1,j,j_2}. A similar result can be obtained by interchanging j_1 and j, and so we find that $B_{j_1,j_2,j}$, B_{j,j_2,j_1}, and B_{j,j,j_2} are all related by

$$\frac{B_{j_1,j_2,j}}{2j+1} = \frac{B_{j,j_2,j_1}}{2j_1+1} = \frac{B_{j_1,j,j_2}}{2j_2+1}. \tag{A.8}$$

In addition to eliminating redundant bispectrum components, this relation greatly simplifies the calculation of forces. In the original calculation, the spatial derivative of the bispectrum components was written as

$$
\begin{aligned}
\nabla B_{j_1,j_2,j} \;=\; & \sum_{m,m'}^{j} (\nabla u^{j}_{m,m'})^* \sum_{m_1,m_1'}^{j_1} H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'} \\
& + \sum_{m,m'}^{j} (u^{j}_{m,m'})^* \sum_{m_1,m_1'}^{j_1} H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; \nabla u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'} \\
& + \sum_{m,m'}^{j} (u^{j}_{m,m'})^* \sum_{m_1,m_1'}^{j_1} H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; u^{j_1}_{m_1,m_1'} \nabla u^{j_2}_{m_2,m_2'}
\end{aligned}
\tag{A.9}
$$

where the symbol ∇ denotes the derivative of what follows with respect to the position of some neighbor atom. We have dropped the double summation over m_2 and m_2', as the coupling coefficients are non-zero only for $m_2 = m - m_1$, likewise for m_2'. In the first term, the inner double sum over m_1 and m_1' contains no derivatives, and so can be pre-calculated. Hence, for the highest order bispectrum component ($j_1 = j_2 = j = J$), the computational complexity of this term is $O(J^2)$. However, in the second and third term, the inner double sums contain derivatives and so must be calculated separately for each neighbor atom and for each entry in the outer double sums over m and m'. As a result, the computation complexity of the second and third terms is $O(J^4)$. Using the symmetry relation given by Eq. 6, we can rewrite this as

$$
\begin{aligned}
\nabla B_{j_1,j_2,j} \;=\; & \sum_{m,m'}^{j} (\nabla u^{j}_{m,m'})^* \sum_{m_1,m_1'}^{j_1} H^{jmm'}_{\substack{j_1 m_1 m_1' \\ j_2 m_2 m_2'}} \; u^{j_1}_{m_1,m_1'} u^{j_2}_{m_2,m_2'} \\
& + \frac{2j+1}{2j_1+1} \sum_{m_1,m_1'}^{j_1} (\nabla u^{j_1}_{m_1,m_1'})^* \sum_{m,m'}^{j} H^{j_1 m_1 m_1'}_{\substack{jmm' \\ j_2 m_2 m_2'}} \; u^{j}_{m,m'} u^{j_2}_{m_2,m_2'} \\
& + \frac{2j+1}{2j_2+1} \sum_{m_2,m_2'}^{j_2} (\nabla u^{j_2}_{m_2,m_2'})^* \sum_{m_1,m_1'}^{j_1} H^{j_2 m_2 m_2'}_{\substack{j_1 m_1 m_1' \\ jmm'}} \; u^{j_1}_{m_1,m_1'} u^{j}_{m,m'}
\end{aligned}
\tag{A.10}
$$

Written in this way, all of the inner double sums are free of derivatives and can be pre-calculated. This has the effect of reducing overall the computational complexity from $O(J^4)$ to $O(J^2)$. For typical cases this results in more than one order of magnitude reduction in computational cost.