Accepted Manuscript

Force-based optimization of pseudopotentials for non-equilibrium configurations

Casey N. Brock, Brandon C. Paikoff, Muhammad I. Md Sallih, Alan R. Tackett, D. Greg Walker

PII: S0010-4655(16)00007-2
DOI: http://dx.doi.org/10.1016/j.cpc.2016.01.001
Reference: COMPHY 5843

To appear in: Computer Physics Communications

Received date: 29 September 2015
Revised date: 29 December 2015
Accepted date: 3 January 2016

Please cite this article as: C.N. Brock, B.C. Paikoff, M.I. Md Sallih, A.R. Tackett, D.G. Walker, Force-based optimization of pseudopotentials for non-equilibrium configurations, Computer Physics Communications (2016), http://dx.doi.org/10.1016/j.cpc.2016.01.001

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.
Force-based optimization of pseudopotentials for non-equilibrium configurations

Casey N. Brock\textsuperscript{a}, Brandon C. Paikoff\textsuperscript{b}, Muhammad I. Md Sallih\textsuperscript{b}, Alan R. Tackett\textsuperscript{c}, D. Greg Walker\textsuperscript{b,}\textsuperscript{*}

\textsuperscript{a}Materials Science Program, Vanderbilt University, Nashville, TN 37235
\textsuperscript{b}Mechanical Engineering, Vanderbilt University, Nashville, TN 37235
\textsuperscript{c}Advanced Computing Center for Research and Education, Vanderbilt University, Nashville, TN 37235

Abstract

We have used a multi-objective genetic algorithm to optimize pseudopotentials for force accuracy and computational efficiency. Force accuracy is determined by comparing interatomic forces generated using the pseudopotentials and forces generated using the full-potential linearized augmented-plane wave method. This force-based optimization approach is motivated by applications where interatomic forces are important, including material interfaces, crystal defects, and molecular dynamics. Our method generates Pareto sets of optimized pseudopotentials containing various compromises between accuracy and efficiency. We have tested our method for LiF, Si\textsubscript{0.5}Ge\textsubscript{0.5}, and Mo and compared the performance of our pseudopotentials with pseudopotentials available from the ABINIT library. We show that the optimization can generate pseudopotentials with comparable accuracy (in terms of force matching and equation of state) to pseudopotentials in the literature while sometimes significantly improving computational efficiency. For example, we generated pseudopotentials for one system tested that reduced computational work by 71\% without loss of accuracy. These results suggest our method can be used to generate pseudopotentials on demand that are tuned for a user’s specific application, affording gains in computational efficiency.

\textsuperscript{*}Corresponding author

\textsuperscript{Email address: greg.walker@vanderbilt.edu (D. Greg Walker)}
1. Introduction

Pseudopotential methods for density functional theory (DFT) calculations [1, 2] can drastically reduce computational time by reducing calculations involving core electrons, and have been widely used in computational materials science since their introduction in 1979 [3]. However, pseudopotentials that are both accurate in reproducing all-electron results and efficient in the computational work required are difficult to generate because these objectives tend to compete with each other. In this work, we generate pseudopotentials based on their accuracy in predicting atomic forces in non-equilibrium crystal configurations and their computational work demands. Our force-based approach is different from previous work, where pseudopotentials have been verified based on equilibrium properties such as lattice constant and bulk modulus [4, 5].

Motivations for our force-based approach are applications where atoms are not in a perfect-crystal environment, such as defects in lattices, material interfaces, or molecular dynamics (MD). In the context of MD, Car-Parinello [6] or Ehrenfest dynamics [7] require forces obtained from \textit{ab initio} electronic structure calculations using DFT and the pseudopotential approximation, therefore pseudopotentials should be designed to produce accurate forces.

Pseudopotentials are typically tuned by adjusting several parameters until physical quantities are calculated with sufficiently minimal errors as compared to all-electron results. Commonly these quantities are equilibrium lattice constant and bulk modulus [4, 5], but other metrics based on perfect crystals are also used [8]. If a pseudopotential produces low error for the physical quantity being calculated then the pseudopotential is said to be good. The question that remains is will the pseudopotential optimized for equilibrium perfect crystals work for atoms that are not in their equilibrium positions? In other words, if we break symmetry by adding a defect or perturbing atoms in the lattice, how can we know that the pseudopotential that was derived for a perfect periodic crystal is still “good?” To answer this question, we have chosen to optimize based on...
forces in non-perfect crystals to ensure that subsequent force calculations are accurate.

We have used a multi-objective genetic algorithm (MOGA) to optimize pseudopotentials for accuracy and computational work by altering a set of parameters used to create pseudopotentials. The use of genetic algorithms for tuning pseudopotentials was introduced in Ref. [9]. In that work, though, the genetic algorithm was used to generate accurate, efficient, and transferable pseudopotentials based on equilibrium properties. We are not claiming in this work that pseudopotentials generated using the present approach are transferable. Instead we claim they are applicable to arbitrarily perturbed atomic configurations of the chosen material system. The accuracy of a pseudopotential is partially based on a comparison between forces calculated using the pseudopotential method and forces calculated using the all-electron full-potential linearized augmented-plane wave (FP-LAPW) method. The computational work of a pseudopotential is based on an estimation of the floating-point operations required to run the force calculations. After a specified number of pseudopotentials have been created, the non-dominated pseudopotentials are chosen for the Pareto set. A pseudopotential is non-dominated if it cannot be improved in one objective without worsening the other. The resulting Pareto set contains design points with varying compromises between accuracy and efficiency. Our method could make pseudopotentials more accessible to a broader range of scientists and engineers by automating the optimization of pseudopotentials. The optimizer handles the design, generation, and evaluation of pseudopotentials, resulting in not just one, but many optimized pseudopotentials with varying user selectable compromises between accuracy and efficiency.

Currently, users of pseudopotential-based tools must either choose prebuilt pseudopotentials from libraries or generate custom pseudopotentials. Pseudopotential libraries are tuned by hand with the help of experts in the field. For example, the GBRV pseudopotential library [4] was published in 2014 with Rabe, who helped develop the RRRKJ method 25 years ago [10], and Vanderbilt, who developed the widely used Vanderbilt Ultra-soft pseudopotential 25 years ago.
For another example, the ABINIT pseudopotential library [5] published in 2014 was developed with the help of Holzwarth, who developed the atompaw code for generating PAW pseudopotentials in 2001 [12].

As an alternative to choosing pseudopotentials from libraries, users can generate their own pseudopotentials using codes such as atompaw [12]. This approach has the advantage of allowing the user to choose desired combinations of exchange and correlation functionals, semi-core states, and other pseudopotential parameters. However, tuning the parameters of these pseudopotentials can be a daunting task for users without years of experience with pseudopotentials. Gianozzi, a developer of the Quantum ESPRESSO DFT code [13], published a document online detailing steps for tuning a pseudopotential, describing some of the steps as “closer to cooking than to science.” [14] In an attempt to codify experience and to automate the process, we turn to computer-based optimization. In contrast to Ref. [15], our optimization technique evolves a Pareto set of optimal pseudopotentials, which is a direct consequence of independent accuracy and work objectives, instead of collapsing competing objectives into a single quality function.

For this report, we completed three force-based optimizations: one for lithium and fluorine in LiF, one for silicon and germanium in Si$_0.5$Ge$_0.5$ (referred to as SiGe for the rest of this paper), and another for elemental molybdenum. We chose these materials for two reasons. First, good pseudopotentials are already available to use as a reference when testing the pseudopotentials. Second, these materials are bonded differently, and we show that our method works well for a variety of chemical environments. The optimizations resulted in a Pareto set of optimal pseudopotentials for each material. To ensure that force-optimized pseudopotentials yield equilibrium properties, we also tested several of our optimized pseudopotentials for their ability to predict equilibrium lattice constant and bulk modulus. Moreover, we tested the optimized pseudopotentials in atomic configurations unknown to the optimizer. This step ensures that the optimized pseudopotentials are generally applicable to any atomic configuration and not just valid for the optimized configurations. Finally, results using our op-
timized pseudopotentials were compared with results given by pseudopotentials from the ABINIT library.

2. Methods

2.1. Pseudopotential optimization

A multi-objective genetic algorithm (MOGA), implemented using the DAKOTA software package [16], was used for optimizing pseudopotentials. The two objectives for the optimization were accuracy and computational work, which MOGA tries to minimize (the accuracy objective should be minimized because it is based on force residuals). The genetic algorithm (GA), which is based on biological evolution, starts with a random population of design points, where each design point is a unique set of values for the design variables. The design variables are inputs to atompaw used to create pseudopotentials (one pseudopotential for each element in the material), and the goal of MOGA is to find the best values for these design variables. For LiF and SiGe, which have two elements, each design point creates a pair of pseudopotentials, so the elements are being optimized simultaneously. The design variables used for each material system are shown in Table 1. Each design point is tested to determine its work and accuracy objectives, where lower objective values are desirable. Design points with lower objectives are given higher fitness values. Design points from each generation with low fitness values are discarded, and the remaining design points are subjected to genetic operations (mutation and crossover), which introduce diversity, to create the next generation of design points. We used MOGA’s random/shuffle crossover operator, which randomly selects design variable values from a specified number of parents to create the offspring. The specified number of parents was 10, the specified number of offspring was 20, and the crossover rate was 0.8. Mutation was carried out using uniform replacement, where every design variable for the design point is changed to a random value distributed uniformly in the allowed range of that design variable. The mutation rate was 0.3.
Figure 1: MOGA workflow. For each design point, a pair of pseudopotentials is created and tested simultaneously.

As MOGA progresses, the optimizer should find design points with higher fitness values. Because accuracy and work are competing objectives, there is no single best design point. Instead, all design points that are non-dominated are chosen for the Pareto set, which is the final product of MOGA (see Figure 3, for example). A non-dominated design point cannot be decreased in one objective without increasing the other objective. For this study, design points in the Pareto set that produce less error will require more work, and those that produce more error will require less work. In addition to searching for design points with higher fitness, the optimizer also attempts to fill in gaps in the Pareto front.

To determine the error given by the pseudopotential method, we compared atomic forces calculated using pseudopotentials to atomic forces calculated using the all-electron full-potential linearized augmented-plane wave (FP-LAPW)
Table 1: Design variables and specified ranges used for MOGA. RC stands for cutoff radius. EP and ED are reference energies for P and D orbitals.

<table>
<thead>
<tr>
<th>Material</th>
<th>Design Variable</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiF</td>
<td>RC\textsubscript{Li}</td>
<td>0.2 Bohr</td>
<td>4 Bohr</td>
</tr>
<tr>
<td></td>
<td>RC\textsubscript{F}</td>
<td>0.2 Bohr</td>
<td>4 Bohr</td>
</tr>
<tr>
<td></td>
<td>EP\textsubscript{F}</td>
<td>0 Ry</td>
<td>15 Ry</td>
</tr>
<tr>
<td>SiGe</td>
<td>RC\textsubscript{Si}</td>
<td>0.2 Bohr</td>
<td>4 Bohr</td>
</tr>
<tr>
<td></td>
<td>EP\textsubscript{Si}</td>
<td>0 Ry</td>
<td>15 Ry</td>
</tr>
<tr>
<td></td>
<td>RC\textsubscript{Ge}</td>
<td>0.2 Bohr</td>
<td>4 Bohr</td>
</tr>
<tr>
<td></td>
<td>EP\textsubscript{Ge}</td>
<td>0 Ry</td>
<td>15 Ry</td>
</tr>
<tr>
<td>Mo</td>
<td>RC\textsubscript{Mo}</td>
<td>0.2 Bohr</td>
<td>4 Bohr</td>
</tr>
<tr>
<td></td>
<td>EP\textsubscript{Mo}</td>
<td>0 Ry</td>
<td>15 Ry</td>
</tr>
<tr>
<td></td>
<td>ED\textsubscript{Mo}</td>
<td>0 Ry</td>
<td>15 Ry</td>
</tr>
</tbody>
</table>

method. Both the pseudopotential method and the all-electron method share the approximations of DFT, so any differences in results between the two methods can be attributed to the pseudopotential used. The all-electron method was considered exact for the purposes of this work.

In addition to comparing forces, we also used the atomic wave functions to effectively screen pseudopotentials containing ghost states. [17] Traditionally, ghost states are detected by comparing the logarithmic derivatives of the pseudopotential and all-electron radial wave functions, defined as

\[
\frac{1}{\Psi_{El}(r_c)} \frac{d\Psi_{El}}{dr},
\]

where \(\Psi_{El}\) is the wave function, \(r\) is the radial position, and \(r_c\) is the cutoff radius. Visual inspection of the logarithmic derivatives of the two wave functions allows pseudopotential designers to quickly assess the quality of a pseudopotential before it’s put into a crystal. This otherwise manual operation can be codified for the purposes of optimization by looking at the root mean square (RMS) of the logarithmic derivative residuals. However, directly minimizing this RMS is a poor choice for automated evaluation due to discontinuities in the logarithmic derivatives. Any slight misalignment between the all-electron and pseudo wave functions leads to a large change in the residual sum. Even worse, as the misalignment decreases, the residual sum steadily increases until
the wave functions significantly overlap, at which point the residual sum starts decreasing. As a result, trying to minimize this term using traditional gradient based techniques can lead to solutions where the all-election and pseudo wave functions have no overlap, resulting in a useless solution. Only if the initial pseudopotential parameters lead to a starting guess where the all-election and pseudo wave functions overlap sufficiently to be in the local minima or region of attraction will the optimization lead to a potentially useful solution.

Alternatively one can treat the logarithmic derivative as

\[ \tan \Theta = \frac{1}{\Psi_{Ei}(r_c)} \frac{d\Psi_{Ei}}{dr} \quad (2) \]

and use RMS of the all-election and pseudo wave function \( \Theta \) residuals. This approach doesn’t suffer from having a region of repulsion surrounding a logarithmic discontinuity misalignment. In fact, as the original logarithmic derivatives align themselves, the \( \Theta \) residuals decrease. Each traversal of a discontinuity coincides with a \( \pi \) phase shift which accumulates over the plot. Consequently, extra discontinuities, including those that are barely visible due to the discrete energy sampling, have a dramatic effect on the resulting objective and hence the optimization. This makes using the RMS of the \( \Theta \) residuals an excellent choice for evaluating the pseudopotential quality from an automated point of view.

The projector augmented wave (PAW) pseudopotential method [18] was implemented using the socorro code from Sandia National Laboratories [19]. The all-electron method was implemented with the Elk code [20]. The exchange and correlation functionals used the PBE version of the generalized gradient approximation (GGA) [21]. For Elk, the exchange and correlation functionals were implemented using LibXC [22]. For socorro and atompaw the built-in implementations of the exchange and correlation functionals were used. A 4\( \times \)4\( \times \)4 Monkhorst-Pack k-point grid [23] was used for all calculations. This k-point mesh may not have been sufficient for convergence to physical values, but since both the all-electron and pseudopotential methods use the same k-point grid the results can be compared without convergence being required [4].

Atoms in the crystals studied were perturbed from the equilibrium posi-
tions to produce net forces on the atoms. The equilibrium configuration is the natural crystal structure of each system; rock-salt for LiF, zinc blende for SiGe, and BCC for Mo. The atomic forces were calculated for four different non-equilibrium random atomic configurations, which we call training configurations. After testing optimizations with two, four, and eight training sets, we concluded that four was sufficient for our systems because we did not observe significant changes in results of the optimizations when using more than four training configurations. Nevertheless, we concede that more training sets could be necessary for more complex systems. The generation of the random configurations used for the optimizations is discussed in more detail in Section 2.2.

We used the force residual to quantify force error between pseudopotential forces and all-electron forces. The force residual is the magnitude of the vector difference between the two forces on an atom:

$$ r_f = |f_p - f_{AE}|, $$

(3)

where $f_p$ is the force on the atom from the pseudopotential method and $f_{AE}$ is the force on the atom from the all-electron method. A lower force residual represents a better match with the all-electron results and more accurate force. If the pseudopotential and all-electron methods give equal force vectors, the force residual is 0. For all systems we defined the unit cell with two atoms, meaning the atomic forces will be equal and opposite. Therefore, we consider the forces on atom 2 only. The $\Theta$ residual is defined as

$$ r_{\Theta} = \Theta_p - \Theta_{AE}, $$

(4)

where $\Theta_p$ is the arctan of the log-derivative for the pseudized wave function and $\Theta_{AE}$ is the arctan of the log-derivative for the all-electron wave function.

The accuracy objective for a design point is constructed as

$$ S_A = s_f + W_{\Theta}s_{\Theta}, $$

(5)
where \( s_f \) is the sum of the force residuals over the atomic configurations \((N)\),

\[
s_f = \left( \frac{1}{N} \sum_{i=1}^{N} r_{f,i}^2 \right)^{1/2}.
\]

(6)

And \( s_\Theta \) is a sum of the arctan of log-derivatives over each element and each state in each element \((l)\),

\[
s_\Theta = \frac{2}{\pi L} \sum_{\text{elements}} \sum_{l} \left( \frac{1}{M} \sum_{i=1}^{M} (r_{\Theta,i})^2 \right)^{1/2},
\]

(7)

where \( M \) is the number of sample points. The weight \( W_\Theta \) in equation 5 is chosen so that both terms in \( S_A \) are similar orders of magnitude. The weights used for this paper were 0.005 for LiF, 0.022 for SiGe, and 5.3 for Mo.

The work objective was an approximation of the number of floating point operations required to calculate the kinetic energy, potential, electron density, projected eigen-problem, Hamiltonian, overlap operator, and residuals. The number of floating point operations is approximated as

\[
S_W = m \left[ 2n_w + 10b^2 + b^3 + (b + 1)n_d \log_2 n_d + b(3n_w + 4pn_w + 2n_w \log_2 n_w) \right]
\]

(8)

where \( b \) is the number of bands, \( n_w \) is the size of the FFT grid for wave functions and potential, \( n_d \) is the size of the density FFT grid, \( p \) is the number of projectors, and \( m \) is the number of iterations. We chose to estimate computational work using this formula instead of wall time to avoid biases due to different computing environments.

Different pseudopotential pairs required different wave function cutoff \((E_{\text{cut}})\) values for the solution to converge. To determine the required wave function cutoff, we increased it from 10 Ry in steps of 10 Ry and considered the solution converged when

\[
\left( \frac{E_{i-1} - E_i}{E_{i-1}} \right)^2 < 10^{-8},
\]

(9)

where \( E_i \) is the cell energy for the wave function cutoff being tested and \( E_{i-1} \) is the cell energy for the previous wave function cutoff tested. The density cutoff was set to \( 4E_{\text{cut}} \). Pseudopotential pairs that didn’t converge by \( E_{\text{cut}} = 70 \) Ry
were given artificially high work and accuracy objectives so the optimizer would recognize them as bad design points.

Pseudopotentials were generated using the \texttt{atompaw} code [12]. The core and valence configurations used are shown in Table 2. We chose to vary cutoff radii (RC) and projector energies (EP and ED) as design variables for these optimizations. Table 1 shows the \texttt{atompaw} input parameters that were used as design variables for MOGA. The PAW pseudopotentials generated by \texttt{atompaw} have four specified cutoff radii: $r_{\text{paw}}$, $r_{\text{shape}}$, $r_{\text{vloc}}$, and $r_{\text{core}}$ [24]. For simplicity, we used a single value for all four radii. If \texttt{atompaw} could not create a pseudopotential given a set of parameters, an artificially high objective was returned to the optimizer to identify it as poor design point. An artificially high objective was also returned if pseudopotentials took longer than a specified wall time to converge or the pseudopotential resulted in non-convergence of a calculation.

2.2. Random atomic configurations

Atoms in the crystal structure must be moved from the equilibrium position to produce a net force on the atoms. The unit cells for LiF and SiGe have lattice vectors ($\frac{a}{2}$, $\frac{a}{2}$, 0), (0, $\frac{a}{2}$, $\frac{a}{2}$), and ($\frac{a}{2}$, 0, $\frac{a}{2}$) where $a$ is the corresponding lattice constant. The unit cell for Mo has lattice vectors (0, 0, 0), (0, $a$, 0), and (0, 0, $a$). We generated atomic configurations for the optimization by randomly placing atom 2 in the unit cell while leaving atom 1 fixed at the unit cell origin.

To characterize the random atomic configurations, we have used the normalized nearest neighbor distance. This value is proportional to the distance between the closest two atoms in the crystal lattice for a given atomic configuration. It arises because the perturbation of an atom from its equilibrium position...
position will move the atom closer to other atoms in the lattice (although in the zinc blende structure, perturbations sometimes increase the nearest neighbor distance). The closest two atoms in the crystal are not necessarily in the same unit cell. For each configuration, the normalized nearest neighbor distance is defined as

\[ d = \frac{h}{h_e} \]  

where \( h \) is the (non-normalized) nearest neighbor distance for that configuration and \( h_e \) is the nearest neighbor distance for the equilibrium crystal structure. Normalizing to \( h_e \) aids in analysis of atomic distances by letting \( d = 1 \) for equilibrium and \( d < 1 \) when atoms move closer. For LiF (rock-salt structure), \( h_e = a/2 = 3.804 \) Bohr. For SiGe (zinc blende structure), \( h_e = \sqrt{3}a/4 = 4.531 \) Bohr. For Mo (BCC), \( h_e = \sqrt{3}a/2 = 5.157 \) Bohr. The lattice constants used for the force calculations only need to be approximate as longs they are the same for both the all-electron and pseudopotential calculations, ensuring a valid comparison between forces. The lattice constants used were \( a = 7.6 \) Bohr for LiF, \( a = 10.5 \) Bohr for SiGe, and \( a = 6.0 \) Bohr for Mo.

A random atomic configuration was not used if \( d < 0.8 \). This constraint prevents atoms from being too close and producing unrealistically large repulsive forces. Because many atomic positions violate this constraint, atom 2 is only allowed in a certain region in the unit cell. To illustrate this, we generated 100,000 random configurations and plotted the atomic locations in the unit cell (Figure 2). Atom 1 can be seen at the unit cell origin. The positions of atom 2 represent the region of allowed positions for atom 2. Table 3 contains the four random configurations generated for each crystal system that were used for the optimization.

2.3. Evaluation of optimized pseudopotentials

Ideally, pseudopotentials generated to calculate forces accurately would also predict equilibrium properties with similar accuracy. We calculated equilibrium lattice constant and bulk modulus using several design points from the Pareto fronts. The results using our optimized pseudopotentials were compared with
Figure 2: Left: Equilibrium positions of the two unit cell atoms for LiF in the rock-salt structure, SiGe in the zinc blende structure, and Mo in the BCC structure. Right: Illustrations of the region of possible positions of atom 2 for LiF and SiGe. Atom 1 is fixed at the origin. These images were created by generating 100000 random configurations using the method in Section 2.2.
Table 3: Atomic configurations for LiF and SiGe used for this paper. The training configurations were used for the optimization of pseudopotentials. The lattice vectors for LiF and SiGe are \((\frac{a}{2}, \frac{a}{2}, 0)\), \((0, \frac{a}{2}, \frac{a}{2})\), and \((\frac{a}{2}, 0, \frac{a}{2})\). The lattice vectors for Mo are \((a, 0, 0)\), \((0, a, 0)\), and \((0, 0, a)\). Atom 1 position is \((0,0,0)\). The training configurations for each material are sorted by \(d\).

3. Results

3.1. Pareto fronts

For all crystal systems, MOGA created and tested a set number of design points—2500 for LiF and Mo and 7500 for SiGe. The initial generation size for each material was 300 design points. Subsequent generation sizes were determined by MOGA, and new generations were created until the specified number of design points were tested. At the end of the optimization, the non-dominated design points were identified for the Pareto front. The resulting objectives for
each design point tested are shown in Figure 3 and points on the Pareto front are identified in blue. The best design points have low error and work objectives, so they are found towards the bottom and left in the plots. The Pareto front offers a diverse set of design points. In Figure 3, the design points vary from accurate but expensive (top left) to inexpensive but not as accurate (bottom right) with a distribution of points in between.

For comparison, we calculated the work and accuracy objectives of pseudopotentials available from a pseudopotential library. Lithium, fluorine, silicon, germanium, and molybdenum PAW pseudopotentials are available from the ABINIT website [26]. These pseudopotentials, which we used as reference pseudopotentials, are expected to be very accurate. Figure 3 shows the objectives of these reference pseudopotentials in relation to our Pareto front. We were able to generate pseudopotentials with comparable accuracy and efficiency compared to reference pseudopotentials from the literature, as is evident from the accuracy and work objectives of the reference pseudopotentials in Figure 3.

We chose three design points from each of our Pareto fronts to test further. These representative design points are labeled accurate, balanced, and fast based on their positions on the Pareto front. Figure 3 shows the locations on the Pareto fronts of these design points. For LiF and SiGe, the design point on the Pareto front with the best accuracy objective was chosen for accurate. For Mo, the second Pareto point was chosen for accurate instead of the first. Mo Pareto point 1’s work objective is 160% higher than Pareto point 2’s work objective while offering only a 3% improvement in accuracy objective, making it an unlikely choice for production runs. For all materials, the design point with the lowest work objective was chosen for fast. We chose a design point near the middles of our Pareto fronts for the balanced pseudopotentials to represent a balance between accuracy and work.

3.2 Genetic algorithm vs. random sampling and parametric sweep

Genetic algorithms (GA) are known to be advantageous for complex design spaces because they are less likely to be caught in a local minimum. However,
we performed tests of the GA compared to brute force techniques to determine whether the GA (1) explored the entire design space, and (2) found the superior Pareto set. For the first test we randomly sampled the design space with 7500 points—the same number of total design points used in the GA optimization. For the second, we gridded the 4D design space with 10 points in each of the two cutoff radius dimensions and 9 points in each of the two projector energy dimensions, for a total of 8100 design points. We used 8100 design points instead of 7500 because we needed a product of 4 approximately equal integers to form the uniform grid. We avoided other optimization techniques because the majority of common optimization techniques can optimize only a single objective. Therefore, comparison to our study, which relies heavily on the quality of a Pareto optimal set with two competing objectives, is not meaningful.

The difference between an optimized Pareto front (using MOGA) and one created from random sampling or a parametric sweep were not as dramatic as we expected. Nevertheless, the GA does seem to push the Pareto front farther towards lower objectives as seen in Figure 4.

Interestingly, the GA approaches its final Pareto front in the first few generations, implying that the GA could be stopped much sooner and still give comparable results. Figure 5 shows the design points after 10 generations (900 design points) of the GA alongside 900 randomly sampled design points. The final Pareto front from the GA after 7500 design points is shown as a solid line. Clearly the design points are near optimal at this point in the optimization since they are near the final Pareto front. Notice the GA design points have approached the corner of the Pareto front, which is arguably the most important region.

3.3. Accuracy of forces in training configurations

The force magnitudes for the training configurations are plotted in Figure 6. Each plot shows the force magnitude on atom 2 in the unit cell calculated with our representative pseudopotentials and reference pseudopotentials for LiF, SiGe, and Mo. In general, the reference, accurate, and fast pseudopotentials
all produce forces reasonably close to the all-electron solution while the fast pseudopotential forces are off by a larger amount. The fast pseudopotentials for all materials are probably unusable due to high force errors. However, other points on the Pareto front offer fast solutions without the excessive accuracy penalty.

Trends in the results are noticeable in Figure 6a for configurations 2-4 and Figure 6c for configurations 2-4. For LiF, reference consistently underpredicts the forces. LiF-accurate is very near the all-electron solution with LiF-balanced and LiF-fast producing higher errors as expected. Similarly for SiGe, SiGe-reference consistently underpredicts the forces. SiGe-accurate and SiGe-balanced are close to the all-electron solution while SiGe-fast both overpredicts and underpredicts by large amounts. For Mo, all pseudopotentials tend to over-predict forces except for training configuration 1.

3.4. Accuracy of forces in random test configurations

We calculated forces for 100 random atomic configurations that were generated after the optimization was complete. These test configurations were generated with the same \( d > 0.8 \) constraint on positions that was used when generating the training configurations. Because these random configurations were not used for the optimization, they can be used to show how well different pseudopotentials predict forces in configurations unknown to the optimizer.

When generating the training configurations, we allowed \( d \) to be any value greater than 0.8. However, the configurations were generated randomly and because of the small sample size, \( d \) is in the range 0.824-0.877 for LiF, 0.825-0.988 for SiGe, and 0.804-0.817 for Mo. Table 3 lists \( d \) for all training configurations, and these optimization ranges are shaded in Figure 7. Because the test configurations represent a larger sample, the range of \( d \) for the test configurations is wider, and many configurations fall outside the range of \( d \) used in the optimization.

Figure 7 shows the percent error in the forces for each test configuration,
calculated as

\[ r_f^* = 100 \times \frac{r_f}{|f_{AE}|}, \]  

(11)

where \( f_{AE} \) is the force on the atom from the all-electron method, and \( r_f \) is the force residual defined in section 2.1. Recall, lower force residuals mean better accuracy, with \( r_f^* = 0 \) being the exact solution. These plots show how well our pseudopotentials calculate forces both inside and outside the optimization ranges. The fast pseudopotentials had by far the largest force residuals for every configuration. With the exception of the Mo training configuration 1 (leftmost training config in Fig. 7c), the residuals for the test configurations are consistent with the residuals for the training configurations. This implies that force residuals for a few training configurations can represent the residuals for a wide range of configurations. In other words, optimizing for a few training configurations will produce pseudopotentials with predictable accuracy in configurations unknown to the optimizer. For LiF, the accurate and balanced pseudopotentials consistently produced lower force residuals than the reference pseudopotential within the optimization region and for \( d \) greater than the optimization region. For the highest values of \( d \), LiF-balanced produced slightly higher force residuals. For \( d \) lower than the optimization region, the residuals for both LiF-accurate and LiF-balanced increased for many configurations and were higher than the reference pseudopotential. The increase in residuals for these smaller \( d \) may result from increased sphere overlap that the optimizer did not anticipate. SiGe-accurate and SiGe-balanced produced lower residuals than the reference pseudopotentials across the entire \( d \) range, with only a few configurations having higher residuals. Compared to differences in force residuals for the other systems, Mo-accurate and Mo-balanced produced practically the same force residuals as the reference pseudopotentials, although Mo-reference did tend to have slightly higher residuals.

3.5. Accuracy of lattice constant and bulk modulus

The pseudopotentials optimized using forces were also evaluated for their ability to estimate equilibrium lattice constant and bulk modulus. The results
were compared with the lattice constant and bulk modulus calculated using both the reference pseudopotentials and the all-electron method. The percent errors in lattice constant and bulk modulus for LiF, SiGe, and Mo are reported in Tables 4, 5, and 6. With the exception of the few points near the bottom of each Pareto front, the optimized pseudopotentials produced lattice constants within 1% of the all-electron lattice constant and bulk modulus values within 5% of the all-electron bulk modulus.

3.6. Design variables

Because LiF has only three design variables—$R_{C_{Li}}$, $R_{C_{F}}$, and $E_{P_{F}}$—its design points can be plotted in a 3D parameter space. Figure 8 shows the work objective, represented by color, as a function of the design variables. This figure (as well as Figure 9) is meant to be cross-viewed in 3D and helps illustrate the distribution of points sampled by the optimizer. Any dependence of the work objective on $E_{P_{F}}$ is visible as a variation of color in the vertical direction. In Figure 8, little variation of color appears in the vertical direction, meaning the work objective has only weak dependence on $E_{P_{F}}$. Indeed, Tables 4, 5, and 6 show no obvious trends in projector energies for any of the systems. Similarly, Figure 9 shows only slight dependence on $E_{P_{F}}$. Although the dependence is weak, in some regions of the design space, $E_{P_{F}}$ is used to refine the pseudopotential and is consequently kept as a design variable. Nevertheless, because of the weak dependence on projector energies, we focus the following discussion on cutoff radii.

We can compare the design variable values for our representative design points with the reference pseudopotentials from the literature (Tables 4, 5, and 6). The atompaw inputs for the reference pseudopotentials are different from our input files in several ways other than $R_C$s and projector energies, so identical $R_C$s and projector energies would not produce identical pseudopotentials. Therefore, the comparison of $R_C$s and projector energies is not a direct comparison. However, the $R_C$s play an important role in pseudopotential properties and should be examined. Upon a cursory inspection, decreasing work corresponds
to larger RCs, which is consistent with common understanding. For LiF, all of the RCs for our representative design points were greater than the reference pseudopotentials, which used $R_{\text{Li}} = 1.6$ Bohr and $R_{\text{F}} = 1.4$ Bohr. Figure 8 shows that the region for $R_{\text{Li}} = 1.6$ Bohr and $R_{\text{F}} = 1.4$ Bohr is a region of high work objective, probably explaining why all of our representative design points have higher RCs, which would fall in regions of lower work objective.

For LiF, the lattice constant and bulk modulus results are noticeably worse after Pareto point 8, which coincides with a large jump in cutoff radii. It is well known that PAW sphere overlap can yield poor and unpredictable results. This motivates an examination of the cutoff radii for the more accurate points on the Pareto front. Pareto point 1 has large cutoff radii, much larger than LiF-reference and large enough to cause sphere overlap. In fact, most design points tested caused sphere overlap. Initially, the possibility of sphere overlap did not concern us because we assumed cutoff radii that were large enough to cause overlap would produce bad force results and thus be ruled out by MOGA. Evidently, this was not the case since the overlap did not cause bad results for our Pareto points. This suggests we could improve our optimizations by constraining the cutoff radii or using more training configurations. Using more training configurations would increase the chances of overlap resulting in design points with smaller cutoff radii.

3.7. Larger unit cell

We tested our pseudopotentials in larger unit cells to understand the scalability of our results to larger systems. For LiF and SiGe, we used the respective rock-salt and zinc blende conventional unit cells, which contain 8 atoms. For Mo, we used a $2 \times 2 \times 2$ conventional BCC unit cell, containing 16 atoms total. We perturbed the atoms from their equilibrium positions by a random amount in each direction. The maximum allowed perturbations were $1.9$ Bohr for LiF, $2.1$ Bohr for SiGe, and $1.2$ Bohr for Mo. The configurations were constrained to $d > 0.85$ to produce configurations in the optimization ranges for LiF and SiGe. Random configurations were generated using the allowed perturbations until a
configuration was found satisfying this constraint. We used the same $d > 0.85$ constraint for Mo even though $d = 0.85$ is not in optimization range. According to Fig. 7c, the Mo pseudopotentials perform similarly across the entire range of $d$ so the choice of $d$ for Mo should not matter. Having already shown that the fast pseudopotentials are too inaccurate to be useful, we only show results for accurate and balanced pseudopotentials in this section. Figure 10 shows $r^*_f$ for each atom in the larger unit cells for LiF, SiGe, and Mo.

For LiF, our optimized pseudopotentials produced much higher force residuals than LiF-reference in the large unit cell, especially for the fluorine atoms. This result was unexpected since our pseudopotentials produced better residuals in the test configurations. We think this result is due to the PAW sphere overlap discussed in section 3.6, despite the interatomic spacings being in the same range as in Figure 7. For LiF, the large cutoff radii did not hurt our force results for the training configurations, or the design points would have produced poor force results and been given worse accuracy objectives. Also, the sphere overlap was not an issue for the test configurations, as shown in Figure 7. Perhaps this disparity in results is because our random perturbations did not sample Li-Li interactions or F-F interactions, instead only varying distances and angles between neighboring atoms of Li and F. This implies improvements could be made in the method by sampling interactions for like atoms by using a larger unit cell.

We chose an additional LiF design point from the optimization to test, design point 2454. This point had smaller cutoff radii than any of the Pareto points and a higher work objective. It is indicated in Figure 3. If this point had a slightly lower accuracy objective, it would have been on the Pareto front. Figure 10a shows that this design point produces similar residuals to the reference pseudopotential. We expect that if the Li-Li and F-F interactions had been sampled more, smaller cutoff radii would have been necessary to produce accurate forces and the optimizer would have selected points with smaller cutoff radii.

For SiGe, the lack of Si-Si sampling or Ge-Ge sampling did not seem to de-
grade the force results when using the larger unit cell. The force residuals for
our optimized pseudopotentials were mostly lower than reference pseudopotentials, as expected from the results in Figure 7. Perhaps it was acceptable not to
to sample Si-Si and Ge-Ge interactions in this case because silicon and germanium
are chemically similar elements, making an Si-Si or Ge-Ge interaction similar
to a Si-Ge interaction. The low residuals produced by SiGe-accurate and SiGe-
reference in the low d range in Figure 7 also imply that SiGe is not sensitive to
the sphere overlap in this case when calculating forces. SiGe-balanced had low
residuals despite being farther down the Pareto front. This is due to the shape
of the Pareto front (Figure 3b). The left section of the Pareto front is nearly
vertical, meaning all of the points on the left edge of the front have very similar
accuracies. This feature of the Pareto front is advantageous because one can
decrease the work significantly while maintaining high accuracy, as shown by
the low force residuals produced by SiGe-balanced.

For Mo, the trends in force residuals for the larger unit cell are comparable
to trends in force residuals for the test configurations, with residuals being
similar between the different pseudopotentials. Our hypothesis—that the lack
of same-element sampling in LiF caused poor results for larger LiF unit cell—is
supported by the Mo findings because the residuals for this plot are the same
as in Figure 7.

3.8. SiGe in cesium chloride structure

For our pseudopotentials to be useful, they must have some degree of local
transferability, meaning they should work in materials related to the optimized
material even if they aren’t meant to be universally transferable to all materi
als. To examine the local transferability of our optimized pseudopotentials, we
tested our SiGe pseudopotentials in the cesium chloride structure. The CsCl
structure is chemically different from the optimized structure because it uses a
cubic primitive cell instead of a rhombohedron, meaning the atoms are coordi
nated differently. The unit cell for the CsCl structure has lattice vectors (a, 0, 0),
(0, a, 0), and (0, 0, a) where a = 6.14 Bohr for CsCl structured SiGe. A new set
of test configurations was generated for the CsCl structure with the constraint
\( d > 0.8 \) where \( h_e = \sqrt{3}a/2 \). The procedure for testing the pseudopotentials
in the CsCl structure is the same as in section 3.4. We tested the pseudopo-
tentials in 100 random test configurations by comparing forces calculated with
pseudopotentials to forces calculated with all-electron DFT.

As expected, SiGe-accurate and SiGe-balanced produced reasonable force
values for the 100 test configurations (Figure 11). The optimization range in this
figure is the same as in Figure 7(b), but plotted using absolute nearest-neighbor
distances instead of normalized nearest neighbor distances. For larger \( h \), both
SiGe-accurate and SiGe-balanced produced higher errors than SiGe-reference,
but still within reasonable values. These high \( h \) configurations had values of \( h \)
outside the optimization range, probably explaining the increase in force errors
in that region. The higher force residuals outside the optimization region suggest
that these pseudopotentials could be further improved with a better sample
of configuration space using larger unit cells or various unit cells. In other
words, an intelligent selection of configurations could improve the optimization
compared to a random selection of configurations.

3.9. Computational work

For LiF, all of the Pareto points have lower work objectives than the reference
pseudopotentials, meaning the Pareto points should all be faster. The slowest
LiF Pareto point had a work objective 34% lower than LiF-reference, and the
fastest had a work objective 76% lower. LiF design point 2454, which was
discussed in Section 3.7, has a work objective 15% lower than LiF-reference.

The results for LiF design point 2454 alone justify using optimization to develop
pseudopotentials to achieve speedup with arguably comparable accuracy. This
point is further supported by results for SiGe, where all Pareto points again
had a lower work objective than SiGe-reference. SiGe-accurate is 11% lower,
SiGe-balanced is 71% lower, and SiGe-fast is 81% lower. However, these results
are not universal.

In contrast to LiF and SiGe, most Mo Pareto points had higher work objec-
tives than Mo-reference, with 54 Pareto points being higher, one being equal, and the remaining 15 being lower. Mo-accurate had a work objective 33 times higher than Mo-reference, while Mo-balanced was 17% lower and Mo-fast was 64% lower. To understand why we didn’t get the same favorable results for Mo as we did for LiF and SiGe, we consider the crystal environments.

For Mo, the optimizer created a pseudopotential, Mo-balanced, that is practically identical to Mo-reference in terms of accuracy and work. Even though both pseudopotentials produced very similar results, they are not equivalent pseudopotentials. In fact, the design variables in Table 6 indicate that these are very different pseudopotentials. Nevertheless, the performance between these two pseudopotentials for the test configurations, large unit cell, and lattice constant/bulk modulus is similar. We believe this similarity in the performance of the pseudopotentials is a result of optimizing Mo in the same target structure (BCC) as Mo-reference [5]. Considering both pseudopotentials were optimized for the same crystal structure, it is not surprising that the optimized pseudopotential is comparable to the reference pseudopotential. The optimization for the same crystal structure may explain why the results from the Mo pseudopotential optimization do not show any significant gains in speed.

In contrast, for Si and Ge the target structure of the optimization was different than those of the reference Si and Ge pseudopotentials, which were separately optimized as elemental crystals. [5] We were able to optimize Si and Ge together to produce speed gains that optimizing Si and Ge individually may not have produced. This result also supports our idea that generating pseudopotentials on demand for specific systems can allow speedup without sacrificing accuracy. Perhaps large scale simulations can be sped up dramatically while maintaining a high level of accuracy simply by tuning the pseudopotentials for the environment.

4. Conclusions

We have developed a method for automatically optimizing pseudopotentials based on interatomic forces and computational work. Our method produced
Pareto sets of pseudopotentials, with varying compromises between accuracy and efficiency. We have seen that our force-based optimization can produce pseudopotentials of comparable accuracy compared to pseudopotentials from the literature when predicting forces or equilibrium lattice constant and bulk modulus.

But why would we want to employ an optimization for pseudopotential generation? Currently the predominant resources for pseudopotentials are pseudopotential libraries. Our method is not meant to replace the use of pseudopotential libraries. Instead, the method is an alternative approach that could be useful for certain applications. Specifically, the force-based method may offer advantages for applications such as MD where forces are important, or other applications such as defects and interfaces where symmetries are broken. One can even imagine a situation where multiple pseudopotentials are used for the same element in different parts of the same system. For example, a silicon pseudopotential optimized for bulk silicon could be used far away from an interface while a different pseudopotential optimized to interact with the interface material could be used for silicon atoms closer to the interface. Also, our method has the advantage of automation, meaning custom pseudopotentials can be generated on demand.

Although the performance gains are compelling, we concede that several characteristics of our method could have contributed to the favorable results in this paper: the automated optimization, the material-specific optimization, the force-based accuracy metric, or the $\Theta$ accuracy metric. With the current data we can not deconvolute the effects of these different influences on the results. The automatic nature of this method allows for a more thorough search of the design space than would be allowed with traditional hand-tuning methods. Because more pseudopotentials are tested, the probability of finding a high-quality pseudopotential is increased. Additionally, the material-specific optimization allows for generation of pseudopotentials that work well for the chosen system without needing to worry about transferability to unrelated materials, possibly allowing softer pseudopotentials. Finally, both $r_\Theta$ and $r_f$ encourage accurate pseudopo-
tentials, but both are terms in the single accuracy objective, so it is unclear which one has the strongest effect on pseudopotential quality. Arguably both metrics are needed since $\Theta$ discourages ghost states and the forces account for properties that arise in the solid. Regardless of the exact reason for our results, the method successfully optimizes pseudopotentials to produce accurate forces, lattice constant, and bulk modulus values.

To provide direction for optimizing pseudopotentials using the force-based objective, we summarize our conclusions as follows.

1. Bigger unit cells are needed for binary systems to capture like element interactions.
2. Multiple crystal structures are advantageous for training to improve local transferability.
3. Only a few generations ($< 10$) are needed for the MOGA optimization with a reasonable initial population (300).
4. Intelligent selection of the training configurations samples the breadth of the design space better.

Our method optimizes for forces as opposed to equilibrium properties, but the resulting pseudopotentials still accurately reproduce equilibrium lattice constant and bulk modulus values. Because force-optimized pseudopotentials still produce accurate equilibrium properties, the force-based method is valid for producing good pseudopotentials. Additional evidence for the quality of our method is that our pseudopotentials predicted forces well even for atomic configurations unknown to the optimizer, suggesting that pseudopotentials can be optimized using few training configurations to produce accurate forces for that material.

Perhaps the most important finding is that computational work represents an opportunity for improvement of pseudopotentials. For SiGe, we were able to find pseudopotentials whose accuracies are comparable to those of the reference pseudopotentials but were much faster. Our technique suggests that pseudopotentials can be generated as needed for users’ specific applications to achieve
gains in computational efficiency.

Acknowledgements

This work was funded by a research contract from Samsung Semiconductor, Inc.
Figure 3: Design points and Pareto fronts for (a) LiF, (b) SiGe, and (c) Mo. More accurate design points have lower accuracy objectives and are towards the left in the plots. Faster design points have lower work objectives and are towards the bottom in the plots. We calculated the work and accuracy objectives of pseudopotentials from the literature; the objectives of these reference pseudopotentials are indicated on the plots. We chose three design points at different locations on the Pareto front to further evaluate. The chosen design points, accurate, balanced, and fast, are also indicated on the plots.
Figure 4: Pareto fronts for three different optimizations—one using the GA, one using a random sample of the design space, and one using a parametric sweep of the design space. The Pareto fronts are similar but the GA pushes the Pareto front farther towards lower objectives.

Figure 5: Design points after 10 generations (900 design points) of the GA alongside 900 randomly sampled design points. The final Pareto front from the GA after 7500 design points is shown as a solid line.
Figure 6: Magnitude of force on atom 2 for (a) LiF, (b) SiGe, and (c) Mo in the training configurations as generated by the all-electron method, the reference pseudopotentials, and the representative pseudopotentials.
Figure 7: Normalized force residuals for (a) LiF, (b) SiGe, and (c) Mo in their test configurations for the reference and representative pseudopotentials. The optimization region is indicated by the shaded area. Lower force residuals indicate more accurate force results. The hollow points are the normalized force residuals for the training configurations. The lines between points are for visualization and are not meant to imply a certain trend.
Figure 8: Every design point tested for LiF plotted by $R_{CLi}$, $R_{CF}$, and $EP_{F}$ with color representing the work objective $S_{W}$ (arbitrary units). This figure is a cross-view stereoscopic image that can be viewed in 3D. The black squares represent design points that returned artificially high objectives. The lack of significant color variation in the vertical direction indicates that $EP_{F}$ has little effect on the work objective.

Figure 9: Every design point tested for LiF plotted by $R_{CLi}$, $R_{CF}$, and $EP_{F}$ with color representing the accuracy objective $S_{A}$ (arbitrary units). This figure is a cross-view stereoscopic image that can be viewed in 3D. The black squares represent design points that returned artificially high objectives. The lack of significant color variation in the vertical direction indicates that $EP_{F}$ has little effect on the accuracy objective.
Figure 10: Force percent error, $r^*_f$, for each atom in the large unit cell for (a) LiF, (b) SiGe, and (c) Mo as calculated by the reference pseudopotentials and various optimized pseudopotentials.
Figure 11: Normalized force residuals in CsCl structured SiGe test configurations for the reference and representative pseudopotentials. The shaded area represents the optimization region. Lower force residuals indicate more accurate force results. The lines between points are for visualization and are not meant to imply a certain trend.
<table>
<thead>
<tr>
<th>design variables</th>
<th>objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>RC&lt;sub&gt;Li&lt;/sub&gt;</td>
<td>RC&lt;sub&gt;F&lt;/sub&gt;</td>
</tr>
<tr>
<td>ref</td>
<td>1.61</td>
</tr>
<tr>
<td></td>
<td>1.20</td>
</tr>
<tr>
<td>2454</td>
<td>2.07</td>
</tr>
<tr>
<td>1</td>
<td>2.06</td>
</tr>
<tr>
<td>2</td>
<td>1.62</td>
</tr>
<tr>
<td>3</td>
<td>1.62</td>
</tr>
<tr>
<td>4</td>
<td>1.62</td>
</tr>
<tr>
<td>5</td>
<td>1.88</td>
</tr>
<tr>
<td>6</td>
<td>1.88</td>
</tr>
<tr>
<td>7</td>
<td>1.88</td>
</tr>
<tr>
<td>8</td>
<td>1.89</td>
</tr>
<tr>
<td>9</td>
<td>1.82</td>
</tr>
<tr>
<td>10</td>
<td>3.95</td>
</tr>
<tr>
<td>11</td>
<td>3.88</td>
</tr>
</tbody>
</table>

Table 4: Design variables, accuracy objective, work objective, percent errors in lattice constant and bulk modulus, and converged $E_{\text{cut}}$ for LiF reference pseudopotentials, LiF design point 2454, and all points on the Pareto front. LiF-accurate, LiF-balanced, and LiF-fast are shaded.

The RCs specified for the reference pseudopotentials are $r_{\text{paw}}$, $r_{\text{shape}}$, $r_{\text{vloc}}$, and $r_{\text{core}}$ [24].

If only one value is specified for RC, then all four cutoff radii are equal. The percent errors are relative to $a = 7.702$ Bohr and $B = 64.59$ GPa, which are derived from the all-electron calculation.
<table>
<thead>
<tr>
<th>Design Variables</th>
<th>Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_{C_{Si}}$</td>
<td>$E_{P_{Si}}$</td>
</tr>
<tr>
<td>ref 1.90 4.50 2.30 3.00</td>
<td>0.00436</td>
</tr>
<tr>
<td>1.60</td>
<td>2.00</td>
</tr>
<tr>
<td>1.75</td>
<td>1.80</td>
</tr>
<tr>
<td>1.20</td>
<td>1.90</td>
</tr>
<tr>
<td>1 2.65 8.19 1.92 4.08</td>
<td>0.00450</td>
</tr>
<tr>
<td>2 2.65 7.90 1.92 4.08</td>
<td>0.00451</td>
</tr>
<tr>
<td>3 2.65 7.86 1.92 4.08</td>
<td>0.00451</td>
</tr>
<tr>
<td>4 2.74 8.69 1.84 4.67</td>
<td>0.00454</td>
</tr>
<tr>
<td>5 2.65 6.91 1.91 4.08</td>
<td>0.00458</td>
</tr>
<tr>
<td>6 2.65 6.91 1.92 3.41</td>
<td>0.00460</td>
</tr>
<tr>
<td>7 2.48 7.75 1.84 4.67</td>
<td>0.00470</td>
</tr>
<tr>
<td>8 2.46 4.73 2.11 1.89</td>
<td>0.00475</td>
</tr>
<tr>
<td>9 2.34 2.61 2.12 1.89</td>
<td>0.00480</td>
</tr>
<tr>
<td>10 2.34 5.40 2.19 1.14</td>
<td>0.00522</td>
</tr>
<tr>
<td>11 2.09 4.49 2.19 0.51</td>
<td>0.00539</td>
</tr>
<tr>
<td>12 1.82 12.60 2.72 0.23</td>
<td>0.01137</td>
</tr>
<tr>
<td>13 1.95 11.46 2.72 0.23</td>
<td>0.01149</td>
</tr>
<tr>
<td>14 1.95 5.88 2.72 0.23</td>
<td>0.01149</td>
</tr>
<tr>
<td>15 2.65 6.91 3.04 0.23</td>
<td>0.03920</td>
</tr>
<tr>
<td>16 2.65 5.32 2.78 7.02</td>
<td>0.07254</td>
</tr>
<tr>
<td>17 3.53 9.94 3.12 5.31</td>
<td>0.07879</td>
</tr>
</tbody>
</table>

Table 5: Design variables, accuracy objective, work objective, percent errors in lattice constant and bulk modulus, and converged $E_{\text{cut}}$ for SiGe reference pseudopotentials and all points on the Pareto front. SiGe-accurate, SiGe-balanced, and SiGe-fast are shaded. The RCs specified for the reference pseudopotentials are $r_{paw}$, $r_{shape}$, $r_{vloc}$, and $r_{core}$ [24]. If only one value is specified for RC, then all four cutoff radii are equal. The percent errors are relative to $a = 10.59$ Bohr and $B = 72.48$ GPa, which are derived from the all-electron calculation.
Table 6: Design variables, accuracy objective, work objective, percent errors in lattice constant and bulk modulus, and converged $E_{\text{cut}}$ for the Mo reference pseudopotential and some points on the Pareto front. Mo’s front contained 70 points because of the shape of the front, so all Pareto points are not shown in this table. Mo-accurate, Mo-balanced, and Mo-fast are shaded. The percent errors are relative to $a = 5.990$ Bohr and $B = 256.5$ GPa, which are derived from the all-electron calculation.
References


