

Implementation of ultrasoft pseudopotentials in large-scale grid-based electronic structure calculations

Miroslav Hodak,¹ Shuchun Wang,¹ Wenchang Lu,^{1,2} and J. Bernholc^{1,2}

¹*Center for High Performance Simulation and Department of Physics, North Carolina State University, Raleigh, North Carolina 27695, USA*

²*Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831-6367, USA*

(Received 15 May 2007; published 7 August 2007)

An implementation of Vanderbilt ultrasoft pseudopotentials in real-space grid-based electronic structure calculations is presented. Efficient utilization of these pseudopotentials requires the use of different grids for (i) wave functions, (ii) charge density, and (iii) sharply peaked operators within the atomic core radii. High-order interpolation between the various grids is important for accuracy, as is high-order discretization for the differential operators. However, efficiency is also of paramount importance, especially when parallelizing over hundreds or thousands of processors. We describe algorithms and procedures used to achieve an effective implementation in the real-space multigrid code, and provide test results for first-row diatomics, bulk transition metals, and energy-conserving quantum molecular dynamics of water. The code parallelizes efficiently over several thousands of processors on modern parallel supercomputers, such as the Cray XT3 and XT4.

DOI: [10.1103/PhysRevB.76.085108](https://doi.org/10.1103/PhysRevB.76.085108)

PACS number(s): 71.15.Dx, 31.15.Ew, 71.15.Nc

I. INTRODUCTION

The construction of broadly transferable, first-principles norm-conserving pseudopotentials¹⁻³ (NCPPs) was one of the most important developments in the field of *ab initio* methods. These pseudopotentials, which are successfully used to calculate many solid-state properties, are, however, inefficient for atoms with highly localized orbitals. For systems that include first-row and transition metal atoms (such as O, Cu, or Ni), the restriction of normalization condition makes it impossible to construct pseudo-wave-functions that are smooth enough to enable low-cutoff plane-wave calculations or corresponding calculations in real space.

In the early 1990s, a new method to generate the ultrasoft pseudopotentials (UPPs) was introduced by Vanderbilt.⁴ In his scheme, the norm-conserving condition is relaxed and the pseudo-wave-functions are allowed to be as soft as possible within the core region, yielding a considerable reduction of the cutoff energy required to describe them. The UPPs have already been successfully used in plane-wave calculations, but they have not yet been implemented in the real-space methods, which limits the capabilities of real-space calculations in dealing with systems containing atoms with highly localized orbitals.

The plane-wave-based methods have been used to perform electronic structure calculations for a wide range of systems over the past several decades. Although highly successful, an efficient implementation of these methods on massively parallel supercomputers, which are required to treat very large systems, is difficult. This is mostly due to the frequent use of Fourier transforms, which are global operations requiring large amounts of interprocessor communication. For modern massively parallel supercomputers with very fast microprocessors, communication between distant processors becomes a bottleneck, limiting the parallelization efficiency.

Real-space methods are inherently local, and therefore, they can be efficiently used on parallel supercomputers. We

have previously developed the real-space multigrid (RMG) method^{5,6} for electronic structure calculations, which uses norm-conserving pseudopotentials. It obtains results that are very close to those of plane-wave calculations.⁵ This method is by now well established and has already been applied to a large number of systems, including the C₆₀ molecule, nanotubes, defects and impurities in wide gap semiconductors, surfaces and interfaces, and recently also polymers and large biomolecules. It uses a real-space mesh to represent the wave functions, the charge density, and the ionic pseudopotentials. This is advantageous both for parallelization, where each processor can be assigned a region of space, and for convergence acceleration, since multiple length scales can be dealt with separately. It is particularly suitable for application to very large systems on massively parallel computers, because of the compact nature of the resulting equations and the ability to automatically accelerate convergence at all length scales by using the multigrid preconditioner. The density functional theory (DFT) equations are discretized on a three-dimensional real-space grid, employing a high-order *Mehrstellen* discretization⁷ that uses a weighted sum of the wave function and potential values to improve the accuracy of the discretization of the *entire* differential equation, not just the kinetic energy operator. This representation is short ranged in real space, leading to an efficient implementation on massively parallel computers.

This paper describes an efficient implementation of ultrasoft pseudopotentials in a grid-based method, which enables both electronic structure calculations and high-accuracy quantum molecular dynamics on the Born-Oppenheimer surface while preserving the parallelization aspects. For both efficiency and accuracy, our implementation of UPPs uses three grids, for the wave functions, the charge density, and the short-ranged projectors associated with UPPs. To further reduce the grid density, high-order *Mehrstellen* discretizations are developed and presented. This method has been tested on first-row diatomics, bulk transition metals, and molecular dynamics of liquid water. In all cases, the results

agree very well with prior plane-wave calculations and experiment. A full-scale application of this method to nonlinear ballistic electron transport across organic molecules connected to gold electrodes has recently been published.⁸ These calculations included over 4000 valence electrons.

This paper is organized as follows: In Sec. II, an overview of the Vanderbilt ultrasoft pseudopotential scheme is presented; Sec. III deals with the real-space implementation of UPPs; Sec. IV describes test applications to first-row diatomics, bulk transition metals, and molecular dynamics of liquid water; while Sec. V discusses parallelization and test results for up to 4096 CPUs. Sec. VI presents the summary and conclusions.

II. VANDERBILT ULTRASOFT PSEUDOPOTENTIALS

Vanderbilt's ultrasoft pseudopotentials⁴ are constructed so that the wave functions are as smooth as possible. This allows for efficient electronic structure calculations using relatively few plane waves or a relatively coarse grid to represent the wave functions. However, there are still operators that are sharply peaked and, therefore, need high-density grids. In Vanderbilt's scheme, the electron density consists of two parts: a soft contribution given by the squared moduli of the wave functions ϕ_i and a hard contribution given by angular momentum projector functions β_n^l and augmentation functions Q_{nm}^l , which are localized at ionic cores:

$$n(\mathbf{r}) = \sum_i \left[|\phi_i(\mathbf{r})|^2 + \sum_{nm,l} Q_{nm}^l(\mathbf{r}) \langle \phi_i(\mathbf{r}) | \beta_n^l(\mathbf{r}) \rangle \langle \beta_m^l(\mathbf{r}) | \phi_i(\mathbf{r}) \rangle \right]. \quad (1)$$

This formula and others in this section assume that only the Γ point is used for k -point sampling; generalization to different k -space samplings is straightforward.

Relaxation of norm conservation in UPPs leads to the generalized orthonormality condition

$$\langle \phi_i | S(\{\mathbf{R}_I\}) | \phi_j \rangle = \delta_{ij}, \quad (2)$$

where the overlap operator S , which depends on ionic positions $\{\mathbf{R}_I\}$, is defined by

$$S = 1 + \sum_{nm,l} q_{nm} |\beta_n^l \rangle \langle \beta_m^l|, \quad (3)$$

with

$$q_{nm} = \int Q_{nm}(\mathbf{r}) d\mathbf{r}. \quad (4)$$

The total energy functional is given by

$$\begin{aligned} E_{\text{tot}}[\{\phi_i\}, \{\mathbf{R}_I\}] &= \sum_i \langle \phi_i | -\frac{1}{2}\nabla^2 + V_{\text{NL}} | \phi_i \rangle \\ &+ \frac{1}{2} \int \int d\mathbf{r} d\mathbf{r}' \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + E_{\text{xc}}[n] \\ &+ \int d\mathbf{r} V_{\text{loc}}^{\text{ion}}(\mathbf{r})n(\mathbf{r}) + E_{\text{ion}}(\{\mathbf{R}_I\}), \end{aligned} \quad (5)$$

with

$$V_{\text{NL}} = \sum_{nm,l} D_{nm}^{(0)} |\beta_n^l \rangle \langle \beta_m^l|. \quad (6)$$

The coefficients $D_{nm}^{(0)}$ together with β_n^l , Q_{nm}^l , and $V_{\text{loc}}^{\text{ion}}$ are provided by the UPP. The procedure to generate these quantities is given in Ref. 4.

Variational minimization of the total energy functional subject to the constraint of Eq. (2) leads to a generalized eigenvalue problem

$$H|\phi_i\rangle = \epsilon_i S|\phi_i\rangle, \quad (7)$$

where

$$H = -\frac{1}{2}\nabla^2 + V_{\text{eff}} + \sum_{nm,l} D_{nm}^l |\beta_n^l \rangle \langle \beta_m^l| \quad (8)$$

and

$$V_{\text{eff}}(\mathbf{r}) = V_{\text{loc}}^{\text{ion}}(\mathbf{r}) + V_{\text{H}}(\mathbf{r}) + V_{\text{XC}}(\mathbf{r}). \quad (9)$$

The coefficients D_{nm}^l are given by

$$D_{nm}^l = D_{nm}^{(0)} + \int V_{\text{eff}}(\mathbf{r}) Q_{nm}^l(\mathbf{r}) d\mathbf{r}. \quad (10)$$

With UPPs, the forces acting on atoms become

$$\begin{aligned} F_I = & -\frac{dE_{\text{ion}}}{d\mathbf{R}_I} - \int d\mathbf{r} \frac{dV_{\text{loc}}^{\text{ion}}}{d\mathbf{R}_I} n(\mathbf{r}) - \int d\mathbf{r} V_{\text{eff}}(\mathbf{r}) \sum_{nm} \frac{dQ_{nm}^l(\mathbf{r})}{d\mathbf{R}_I} \rho_{nm}^l \\ & - \sum_{nm} D_{nm}^l \frac{\partial \rho_{nm}^l}{\partial \mathbf{R}_I} + \sum_{nm} \frac{\partial \omega_{nm}^l}{\partial \mathbf{R}_I} q_{nm}, \end{aligned} \quad (11)$$

where

$$\rho_{nm}^l = \sum_i \langle \phi_i | \beta_n^l \rangle \langle \beta_m^l | \phi_i \rangle, \quad (12)$$

$$\omega_{nm}^l = \sum_i \epsilon_i \langle \phi_i | \beta_n^l \rangle \langle \beta_m^l | \phi_i \rangle, \quad (13)$$

$$\frac{\partial \rho_{nm}^l}{\partial \mathbf{R}_I} = \sum_i \left[\left\langle \phi_i \left| \frac{\partial \beta_n^l}{\partial \mathbf{R}_I} \right\rangle \langle \beta_m^l | \phi_i \rangle + \langle \phi_i | \beta_n^l \right\rangle \left\langle \frac{\partial \beta_m^l}{\partial \mathbf{R}_I} \left| \phi_i \right\rangle \right], \quad (14)$$

$$\frac{\partial \omega_{nm}^l}{\partial \mathbf{R}_I} = \sum_i \epsilon_i \left[\left\langle \phi_i \left| \frac{\partial \beta_n^l}{\partial \mathbf{R}_I} \right\rangle \langle \beta_m^l | \phi_i \rangle + \langle \phi_i | \beta_n^l \right\rangle \left\langle \frac{\partial \beta_m^l}{\partial \mathbf{R}_I} \left| \phi_i \right\rangle \right], \quad (15)$$

and ϵ_i are eigenvalues from Eq. (7).

III. REAL-SPACE IMPLEMENTATION OF ULTRASOFT PSEUDOPOTENTIALS

The prior implementation of the RMG method^{5,6} uses norm-conserving pseudopotentials. It served as a base for implementing ultrasoft pseudopotentials. In the RMG method, multigrid acceleration is used to efficiently solve Kohn-Sham and Poisson equations. Very good scaling with

respect to the number of processors on parallel supercomputers was achieved by domain decomposition, i.e., by assigning regions of space to specific processors, which then hold grid data for electronic density, wave functions, and potential corresponding to that region. In addition to standard steps for solving Kohn-Sham equations using pseudopotentials, one additional step is required in grid-based calculations, namely, the pseudopotentials need to be filtered to remove high-frequency components. This is necessary for both norm-conserving and ultrasoft pseudopotentials; otherwise, the total energies exhibit unphysical dependence on the position of atoms relative to the real-space grid, unless that grid is unnecessarily dense. The same filtering method as for norm-conserving pseudopotentials was used to remove high-frequency components from $Q_{nm}(\mathbf{r})$ and $\beta_n(\mathbf{r})$. In plane-wave calculations, this filtering is implicit in the use of cutoffs, since the various plane-wave expansions are truncated at the cutoff energies. In grid-based calculations, filtering must be performed explicitly for each atom after generating the pseudopotential from atomic all-electron calculations.

The prior RMG method used one uniform grid to represent the wave functions, the charge density, and the various components of the total potential, including the ionic pseudopotentials. The uniform grid is important for easily preserving hermiticity in the discretization of the kinetic energy operator, and for uniformity and accuracy of forces when performing energy-conserving *ab initio* molecular dynamics simulations. However, when using UPPs, a single uniform grid becomes impractical and inefficient. As mentioned in the previous section, the UPP charge density is hard around the ionic cores, while the wave functions are much smoother. To take advantage of these properties, the ultrasoft real-space implementation has to use two global grids: a fine one for the charge density and the $Q_{nm}(\mathbf{r})$ functions, and a coarse one for representing wave functions. The coarse wave-function grid also leads to faster solving of the Kohn-Sham equations, which usually is the most time-consuming part of electronic structure calculations. Furthermore, it decreases computer memory requirements for storing wave functions, which often is a limiting factor for systems with many electrons. The fine and coarse grids are implemented in such a way that the coarse grid is a subset of the fine grid. Our tests show that using a fine grid with twice as many points in each direction as the coarse grid is usually adequate.

In the calculations, one needs to transfer quantities defined on the fine grid to the coarse grid and vice versa. For example, the potential V_{eff} , Eq. (9), is defined on the fine grid, but it is used to solve generalized Kohn-Sham equations [Eq. (7)] on the coarse grid. In this case, since the coarse grid is a subset of the fine one, it is sufficient to use injection, i.e., use its already known values on the coarse-grid points.

However, the situation is more complicated when transferring from the coarse grid to the fine one. This is needed when calculating electron density according to Eq. (1), since the wave functions $\phi_i(\mathbf{r})$ are only known on the coarse grid, while the electron density is represented on the fine one. An interpolation scheme is therefore required, and our tests show that it must be quite accurate. Since our RMG imple-

mentation uses regular rectangular grids, an interpolation in three dimensions can be obtained as a tensor product of one-dimensional interpolations. Two types of interpolation have been implemented: (i) a local cubic interpolation that uses four points on a coarse grid to find fine-grid values between the second and third coarse-grid points, and (ii) a spline interpolation, using B-splines functions.^{9,10}

The advantage of the local interpolation is that values at any fine-grid point only depend on four nearby coarse-grid points. Therefore, on parallel supercomputers, only values of two grid points stored on other processors have to be retrieved, resulting in modest interprocessor communication. While this scheme is efficient, the resulting fine-grid quantities are not as smooth as those obtained from global interpolation methods, such as spline interpolations, which ensure that local interpolation functions are matched smoothly. The local interpolation methods do not provide any matching between the interpolation functions, and we find that the resulting ionic forces are not accurate enough for dynamical calculations. On the other hand, the accuracy of the calculated total energies is sufficient for use in static calculations that do not utilize forces.

The B-splines interpolation is a global method, and the values of the points on the fine grid depend on all points on the coarse grid. A straightforward implementation of B-splines would, therefore, require extensive interprocessor communication, which would negatively impact performance on massively parallel supercomputers. To avoid this, we use the B-splines interpolation method in a piecewise fashion on the local data available on each processor augmented by the data from nearest-neighbor processors, which usually amounts to three additional grid points per direction. The disadvantage of this approach is that the results of such interpolation are different when different numbers of processors are used. Our tests show that these differences are negligible and the obtained energies and forces stay essentially the same as the number of processors changes. The B-splines interpolation yields significantly more accurate forces than the local interpolation, and it leads to good total energy conservation in *ab initio* molecular dynamics when orders higher than 4 are used. Increasing the order of B-splines yields slightly better energy conservation, but this also increases the computational requirements. The orders of 5 or 6 are usually best to use; they give very good energy conservation and the computational cost of the interpolation is less than 5% of the whole self-consistent field cycle.

In Vanderbilt's UPP scheme, it is often required to perform inner products between the wave functions and the nonlocal projectors, or their derivatives with respect to the ionic positions. As mentioned above, the wave functions are smooth, while the nonlocal projectors usually vary rapidly inside the core regions. To perform these inner products accurately and efficiently, the double grid technique¹¹ is used. In this technique, an inner product calculated on a grid that is denser than the original coarse grid is evaluated on the original grid, with nonlocal projectors replaced by weight factors, which are dependent on the interpolation used to obtain values of wave functions on the denser grid. Note that this grid has no relation to the global fine grid. Denoting the ratio between spacings in coarse and dense grids by n , the dense

TABLE I. Discretization weights for the sixth-order orthorhombic Mehrstellen operators for the central, first, second, third, fourth, and fifth nearest-neighbor grid points. The cubic-grid operator corresponds to $h_i = h_{\text{grid}}$.

	a	b_n	$c_{n,m}$	$d_{n,m,l}$	e_n	$f_{n,m}$
$\mathbf{A}_{\text{Meh}}^l$	$\sum_i \frac{116}{90h_i^2}$	$-\frac{49}{60h_n^2} + \sum_i \frac{31}{80h_i^2}$	$-\frac{1}{10h_n^2} - \frac{1}{10h_m^2} + \sum_i \frac{1}{72h_i^2}$	$-\sum_i \frac{1}{140h_i^2}$	$\frac{1}{120h_n^2} - \sum_i \frac{1}{120h_i^2}$	$\frac{1}{240h_n^2}$
$\mathbf{B}_{\text{Meh}}^l$	$\frac{61}{120}$	$\frac{13}{180}$	$\frac{1}{144}$	0	$-\frac{1}{240}$	0

(coarse) grid volume elements by ΔV_{dens} (ΔV_{coarse}), and the number of the coarse grid points in the core region by N_{core} , the number of dense points is $n^3 N_{\text{core}}$. The inner product can then be written as follows:

$$\langle \beta^l | \phi \rangle = \sum_{j=0}^{n^3 N_{\text{core}}} \beta_j^l \phi_j \Delta V_{\text{dens}} = \sum_{J=0}^{N_{\text{core}}} \Phi_J \omega_J^l \Delta V_{\text{coarse}}, \quad (16)$$

where values of the wave function on the dense (coarse) grid are denoted by ϕ_j (Φ_J). ω_J^l is the weight factor and its form depends on the interpolation method used to obtain values on the dense grid. In this case, it is particularly advantageous to use Fourier interpolation,¹² in which case ω_J^l takes the following form:

$$\omega_J^l = \sum_{k=0}^{N_{\text{core}}} \tilde{\omega}_k e^{i\mathbf{G}_k \cdot \mathbf{X}_J} e^{-i\mathbf{G}_k \cdot \mathbf{R}_l}, \quad (17)$$

with

$$\tilde{\omega}_k = \frac{1}{L^3} \sum_{j=0}^{n^3 N_{\text{core}}} \beta_j e^{-i\mathbf{G}_k \cdot \mathbf{x}_j} \Delta V_{\text{dens}}, \quad (18)$$

where \mathbf{G}_k is the reciprocal space coordinate and \mathbf{x}_j (\mathbf{X}_J) is the real-space coordinate of the dense (coarse) grid. The term representing the position of the ion, R_l , appears only in Eq. (17), while Eq. (18) is independent of R_l . Therefore $\tilde{\omega}_k$, requiring expensive Fourier transform on the dense grid, has to be calculated only once for each atomic species in the entire simulation. The Fourier transform on the original coarse grid [Eq. (17)] has to be recalculated during the simulation, but only when the ionic positions change. In actual simulations, the Fourier transforms consume a negligible fraction of the total time (less than 1%).

The double grid technique is also used for gradients of nonlocal projectors needed to calculate forces [Eq. (11)], where the gradients are similarly replaced by the interpolation weights. Since the nonlocal projectors β_j are strictly localized around the ionic cores, the required Fourier transforms are also local and, thus, do not affect the scaling efficiency.

Another important aspect of the real-space implementation of the UPPs is that since the spacing in the coarse grid is generally much larger than the one used in calculations utilizing NCPPs, the fourth-order Mehrstellen discretization used with NCPPs⁵ for the Kohn-Sham equations is not accu-

rate enough when using UPPs. Instead of the fourth-order expression, we therefore use a sixth-order Mehrstellen discretization, which samples the functions at 57 points

$$\begin{aligned} \mathbf{A}_{\text{Meh}}^l &= a f(\mathbf{x}) + \sum_{n=1}^3 b_n f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n) \\ &+ \sum_{n < m} c_{n,m} f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n \pm h_m \hat{\mathbf{x}}_m) \\ &+ \sum_{n < m < l} d_{n,m,l} f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n \pm h_m \hat{\mathbf{x}}_m \pm h_l \hat{\mathbf{x}}_l) \\ &+ \sum_{n=1}^3 e_n f(\mathbf{x} \pm 2h_n \hat{\mathbf{x}}_n) + \sum_{n < m} f_{n,m} f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n \pm 2h_m \hat{\mathbf{x}}_m), \\ \mathbf{B}_{\text{Meh}}^l &= a' f(\mathbf{x}) + \sum_{n=1}^3 b'_n f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n) \\ &+ \sum_{n < m} c'_{n,m} f(\mathbf{x} \pm h_n \hat{\mathbf{x}}_n \pm h_m \hat{\mathbf{x}}_m) + \sum_{n=1}^3 e'_n f(\mathbf{x} \pm 2h_n \hat{\mathbf{x}}_n). \end{aligned}$$

With this discretization of Kohn-Sham equations, Eq. (7) takes the form

$$\begin{aligned} \mathbf{H}_{\text{Meh}}^l[\phi_i] &= \frac{1}{2} \mathbf{A}_{\text{Meh}}^l[\phi_i] + \mathbf{B}_{\text{Meh}}^l \left[V_{\text{eff}} \phi_i \right. \\ &\quad \left. + \sum_{nm,l} D_{nm}^l |\beta_n^l \rangle \langle \beta_m^l | \phi_i \right] \\ &= \epsilon_l \mathbf{B}_{\text{Meh}}^l[S \phi_i]. \end{aligned} \quad (19)$$

Table I lists the corresponding weights of the sixth-order Mehrstellen operators used in the present RMG code. Note that even the sixth-order expression is short range, extending only by two neighbors in each Cartesian direction. This is significant in limiting the amount of communication.

On the global fine grid, the fourth-order Mehrstellen discretization is accurate enough; thus, the above formula is used only for quantities defined on the global coarse grid.

IV. TEST RESULTS

This section describes the results of test calculations using UPPs and real-space grids for first-row diatomics, bulk transition metals, and *ab initio* simulations of liquid water, and

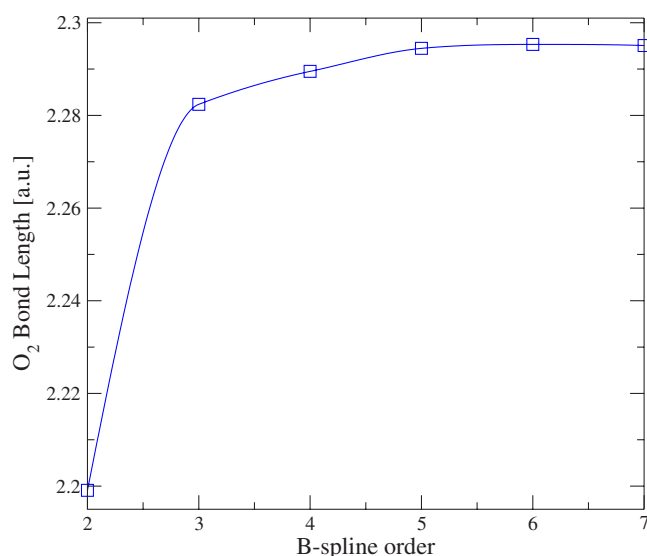


FIG. 1. (Color online) Bond length in O₂ molecule calculated with different B-splines orders. The bond length calculated with local cubic interpolation is 2.269 a.u.

compares them to the corresponding results obtained using plane-wave methods. To make the comparison meaningful, the grid spacing has to be related to the plane-wave kinetic energy cutoff. This can be done by calculating the cutoff of a plane-wave calculation that uses a Fourier transform with grid spacing equal to that of a real-space calculation. The resulting formula⁵ is $E_{\text{cut}} = \pi^2/2h^2$, where h is the grid spacing in the real-space calculation.

For all tests presented in this section, the global fine grid has twice as many points in each direction as the coarse grid. The double grid ratio n was set to 4. The equivalent energy cutoffs reported below are for the wave-function grid, i.e., the global coarse grid.

A. Bond lengths of diatomic molecules

The first test compares bond lengths of diatomic molecules. The simulation supercell is a cube of length 15 a.u. and the grid spacing is 0.421 a.u., corresponding to an energy cutoff of 25 Ry. The k -space sampling was restricted to the Γ point. All calculations were performed within the local-density approximation (LDA) for the exchange and correlation energies.

Figure 1 displays the bond length in O₂ as a function of the order of B-splines used in the calculation. It shows that an order of at least 4 is needed to obtain reliable bond lengths. We chose to use the fifth order and calculated bond lengths of N₂ and CO in addition to O₂. The results are listed in Table II. For comparison, we also list experimental values and the results of plane-wave calculations with similar cutoffs. Our results agree well with both of these sets.

B. Ground state of bulk gold

To test the ability of the RMG method to handle the transition metals, we calculate the equation of state for bulk gold. For this calculation, a 16-atom Au tetragonal supercell

TABLE II. Bond lengths (a.u.) of diatomic molecules.

	N ₂	O ₂	CO
Experiment ^a	2.07	2.28	2.13
This work	2.08	2.29	2.16
Plane-wave UPP ^b		2.33	2.130

^aReference 14.

^bReference 15 and 16.

is used, with 64 points included in k -space sampling. The ultrasoft pseudopotential for Au uses the generalized gradient approximation in the Perdew-Burke-Ernzerhof form¹⁷ for the exchange and correlation terms. Partial core correction is added in order to account for nonlinear effects.¹⁸ Cubic local interpolation is used to calculate the charge density, and the equivalent cutoff energy is 20 Ry.

The total energy is calculated as a function of volume and fitted to the Murnaghan equation of state¹⁹

$$E(V) = \frac{B_0 V}{B'_0(B'_0 - 1)} \left[B'_0 \left(1 - \frac{V_0}{V} \right) + \left(\frac{V_0}{V} \right)^{B'_0} - 1 \right] + E(V_0). \quad (20)$$

The calculated data points and the fit to the equation of state are displayed in Fig. 2. The quality of the fit is excellent.

The fitted curve is used to obtain the equilibrium lattice constant and the equilibrium bulk modulus B_0 . The values of these quantities are listed in Table III. We also list results with the equivalent energy cutoff being increased to 30 Ry. Note that increasing the cutoff leads to significant improvement in the value of the ground-state bulk modulus.

C. Ground state and band structure of bulk copper

The second transition metal application is calculating the band structure of bulk copper. Generally, for a norm-conserving pseudopotential, an energy cutoff of 300 Ry (Ref. 21) is necessary to handle the copper atom due to its highly localized $3d$ electrons. Recently, several smooth norm-conserving pseudopotentials have been generated for copper,

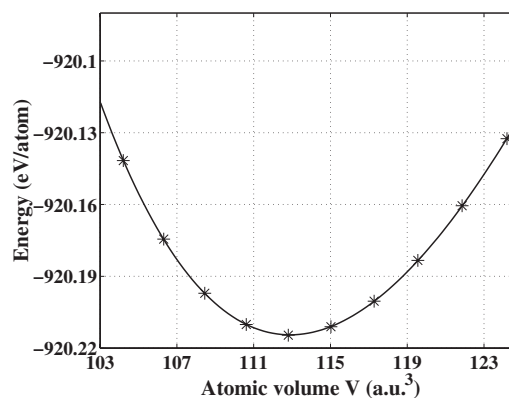


FIG. 2. Equation of state for bulk gold in fcc structure. The continuous curve is the Murnaghan equation of state. The points are the calculated energies at the indicated volume.

TABLE III. Ground-state properties of bulk gold.

	Energy cutoff (Ry)	Lattice constant (a.u.)	Bulk modulus (kbar)
Experiment ^a		7.67	1720
Present work	20	7.66	1919
Present work	30	7.67	1797
Plane-wave UPP ^b	20	7.66	1823

^aReference 20.^bReference 16.

which allow one to use energy cutoffs ranging from 60 to 72 Ry.²² Within the ultrasoft scheme, we were able to use a cutoff energy of only 25 Ry. We used LDA for the exchange-correlation energy and local cubic interpolation between the global coarse and fine grids. The band structure along the Γ - L and Γ - X lines is plotted in Fig. 3. The calculated bandwidths at high symmetry points and the experimental values and results from plane-wave calculations utilizing norm-conserving pseudopotentials are listed in Table IV.

In Table V, we report the fcc lattice constant and the bulk modulus of bulk copper obtained from a fit to the Murnaghan equation of state [Eq. (20)]. The same supercell and k -point sampling as for bulk gold are used. The results are in good agreement with plane-wave calculations and experimental data.

D. Molecular dynamics simulation of liquid water

In this section, the performance of the real-space implementation of ultrasoft pseudopotentials is tested in a quantum molecular dynamics simulation for water. The cubic supercell contains 16 water molecules at a liquid density. The wave function grid spacing is 0.37 a.u., corresponding to a plane-wave energy cutoff of 35 Ry. The RMG code employs Born-Oppenheimer molecular dynamics and a time step of 20 a.u. (0.48 fs) is used. A weight of 2 amu was chosen for the hydrogen atoms. The temperature was kept at 300 K by

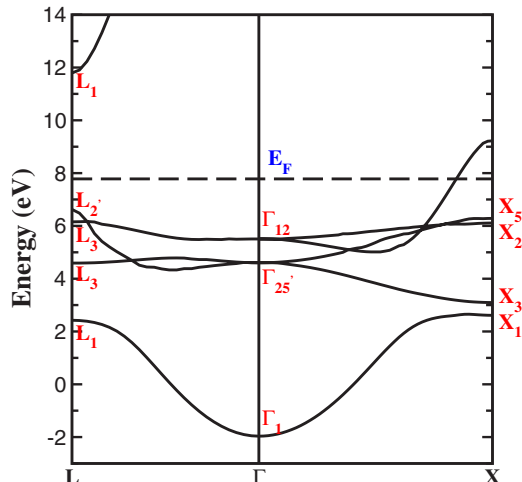


FIG. 3. (Color online) Band structure of bulk copper.

TABLE IV. Cu bandwidths at high-symmetry points. All energies are in eV.

		Present work (with UPP)	Plane-wave method ^a (with NCPP)	Experiment ^b
Widths of d bands	Γ_{12} - $\Gamma_{25'}$	0.90	0.91	0.81
	X_5 - X_3	3.19	3.23	2.79
	X_5 - X_1	3.67	3.70	3.17
	L_3 - L_3	1.56	1.58	1.37
	L_3 - L_1	3.73	3.72	2.91
L gap	L_1 - $L_{2'}$	5.17	4.21	5.20

^aReference 22.^bReference 23.

using five Nosé-Hoover thermostats.^{25,26} In molecular dynamics simulations, it is obviously important to conserve the total energy of the system. To achieve this, the sixth order of B-splines interpolation was used.

Figure 4 shows energies and temperatures in our molecular dynamics simulation. The total energy, shown as a dashed line, is a sum of the DFT total energy, the kinetic energy of ions, and the energy of the Nosé-Hoover thermostats. Although it appears as a straight line in the figure, it exhibits small oscillations of about 0.2 meV around its average value. These oscillations are mostly due to movement of ions relative to the grid points. It is well known that in real-space methods, this causes small energy variations.⁶

V. PARALLELIZATION AND SCALING

Modern parallel supercomputers contain tens of thousands of processors and are capable of running at hundreds of TFLOPS. These numbers are expected to increase rapidly over the next few years. To utilize these resources for simulations of large systems, efficient parallelization over many processors is essential. To test parallelization efficiency, performance tests were run on two liquid water systems on the Cray XT3 supercomputer using up to 4096 processors. The XT3 supercomputer consists of 2.6 GHz dual-core AMD Opteron processors and 4 gigabytes of memory per processor is available. The processors can be used in single-core or dual-core modes. In the dual-core mode, each core is utilized as an independent CPU using 2 gigabytes of memory, while

TABLE V. Lattice constant and bulk modulus of bulk copper.

	Energy cutoff (Ry)	Lattice constant (a.u.)	Bulk modulus (kbar)
Experiment ^a		6.81	1380
Present work	25	6.76	1572
Plane-wave UPP ^b	20	6.72	1659

^aReference 24.^bReference 16.

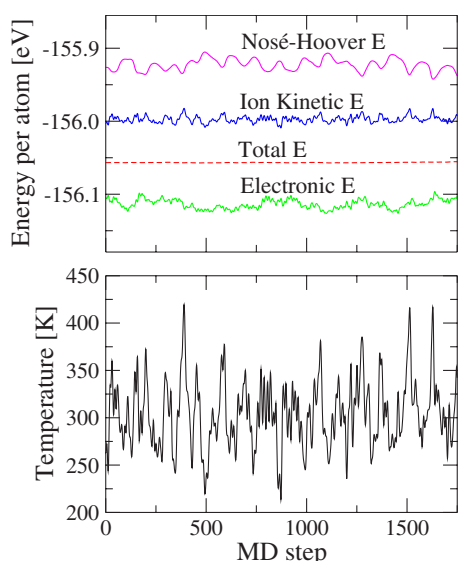


FIG. 4. (Color online) Energies and temperature in molecular dynamics simulation of liquid water. In the upper plot, the dashed line represents the total energy. The ionic and Nosé-Hoover energies were shifted by -156.035 and -155.945 eV in order to appear in one plot with other energies.

in the single-core mode, only one of the cores is active, allowing it to use 4 gigabytes of memory.

To achieve efficient parallel implementation, it is important that the amount of interprocess communication is kept low. In the present grid-based method, there are two types of communication: (i) summation over all processors, which is used to evaluate global quantities such as energy (the cost of this communication is expected to increase as the number of cores used in the calculation increases); and (ii) exchange of boundary data between processors dealing with neighboring regions, which decreases slightly with an increase of the number of cores, because the amount of boundary data per processor decreases.

In the first test, a system of 432 water molecules was used. The total number of electrons was 3456 and the number of wave functions was 1828. The number of cores used in simulation was gradually doubled from 64 to 4096. The scaling performance is shown in Fig. 5. Without the overhead of the interprocess communication, the curve might be expected to be linear or slightly superlinear. The reason for the superlinearity is that when more processors are used, each processor stores less data, which increases the number of the cache hits. However, the cost of the interprocessor communication decreases performance and causes flattening of the scaling curve. In our test case, the effect of communication becomes apparent when the number of processors exceeds 1024.

The second test was performed on a system twice as large, containing 819 water molecules. Numbers of electrons and wave functions in the calculation were 6552 and 3376, respectively. Because of the higher number of wave functions and grid points, this system has higher memory requirements than the first one and can only be run with at least 256 processors in the single-core mode. To utilize both cores, the number of processors has to be 512. The performance data

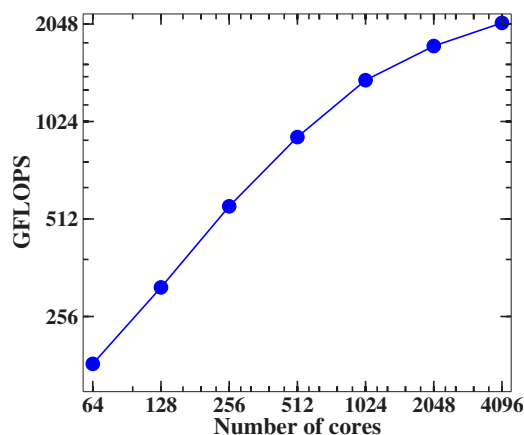


FIG. 5. (Color online) Performance data for a system of 432 water molecules on Cray XT3. Both cores on each processor were used. Points represent data from calculations; the line is a guide for the eyes.

are shown in Fig. 6. The single-core calculation shows slightly better performance than the dual-core one. This is mainly due to the fact that in the dual-core mode, the cost of interprocess communication is higher; our timings show that the time required for communication in the dual-core mode is about 40% larger than the time in the single-core node. The reason for this difference is sharing of the interconnect by the cores on a processor: when both cores use the same interconnect, the communication is less efficient than when each core has its dedicated interconnect.

On the new Cray XT4, our code performs about 20% faster than on the XT3, due to improved speed of memory to processor communications. In summary, the RMG code is well parallelized and suitable for studies of very large systems on modern parallel supercomputers.

VI. SUMMARY AND CONCLUSIONS

Vanderbilt ultrasoft pseudopotentials allow for more efficient electronic structure calculations than norm-conserving ones. In this paper, we have shown how these pseudopotentials can be used on real-space grids and have described an

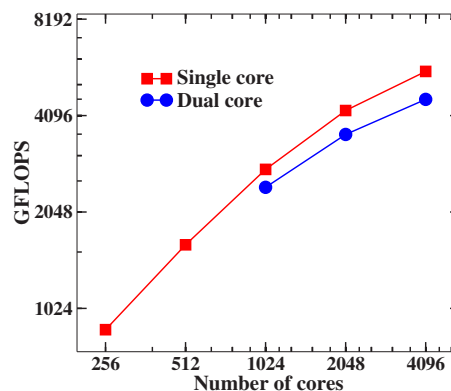


FIG. 6. (Color online) Performance data for a system of 819 water molecules on Cray XT3. Points represent data from calculations; lines are guides for the eyes.

implementation that parallelizes easily and is suitable for very-large-scale massively parallel simulations. To take advantage of the smoothness of the ultrasoft wave functions, different global grids have to be used for the wave functions and the electron density. The choice of interpolation between the two grids has an important influence on the accuracy of the results. Efficient implementation also requires employing the double grid technique for calculating inner products between wave functions and nonlocal projectors. When using ultrasoft pseudopotentials, the wave functions are represented on coarser grids than when norm-conserving pseudopotentials are employed. For that reason, a higher order of discretization of Kohn-Sham equations is required. A sixth-order Mehrstellen discretization was developed for general orthorhombic lattices. It was shown to be both accurate and

efficient, with relatively modest communication needs. Our tests show that a real-space implementation of ultrasoft pseudopotentials yields results that agree very well with those obtained from comparable plane-wave calculations. Performance tests for water molecules at liquid density were run on Cray XT3 and XT4 supercomputers. The results show excellent parallelization efficiency and speed over thousands of processors, proving that the RMG code can be used to study very large systems on modern parallel supercomputers.

ACKNOWLEDGMENTS

We gratefully acknowledge support by NSF, DOE, DOD, and ORNL-LDRD, and grants of supercomputer time provided by DOE and the DOD Challenge Program.

-
- ¹D. R. Hamann, M. Schlüter, and C. Chiang, *Phys. Rev. Lett.* **43**, 1494 (1979).
- ²G. P. Kerker, *J. Phys. C* **13**, L189 (1980).
- ³A. Zunger and M. L. Cohen, *Phys. Rev. B* **18**, 5449 (1978).
- ⁴D. Vanderbilt, *Phys. Rev. B* **41**, 7892 (1990).
- ⁵E. L. Briggs, D. J. Sullivan, and J. Bernholc, *Phys. Rev. B* **54**, 14362 (1996).
- ⁶E. L. Briggs, D. J. Sullivan, and J. Bernholc, *Phys. Rev. B* **52**, R5471 (1995).
- ⁷L. Collatz, *The Numerical Treatment of Differential Equations* (Springer-Verlag, Berlin, 1960), p. 164.
- ⁸S. Wang, W. Lu, Q. Zhao, and J. Bernholc, *Phys. Rev. B* **74**, 195430 (2006).
- ⁹C. D. Boor, *A Practical Guide to Splines* (Springer-Verlag, New York, 1978).
- ¹⁰The B-splines interpolation code was based on the bsplib library (<http://www.netlib.org/a/bsplib.tgz>), which was rewritten into C, optimized and customized to fit the rest of the RMG code.
- ¹¹T. Ono and K. Hirose, *Phys. Rev. Lett.* **82**, 5016 (1999).
- ¹²FFTW library (Ref. 13) is used to perform Fourier transforms.
- ¹³M. Frigo, in *Proceedings of 1999 ACM SIGPLAN Conference on Programming Language Design and Implementation* (ACM, New York, 1999), Vol. 34, pp. 169–180.
- ¹⁴K. P. Huber and G. Herzberg, *Constants of Diatomic Molecules* (Van Nostrand, New York, 1979).
- ¹⁵K. Laasonen, A. Pasquarello, R. Car, C. Lee, and D. Vanderbilt, *Phys. Rev. B* **47**, 10142 (1993).
- ¹⁶A. Dal Corso, A. Pasquarello, and A. Baldereschi, *Phys. Rev. B* **56**, R11369 (1997).
- ¹⁷J. P. Perdew, K. Burke, and M. Ernzerhof, *Phys. Rev. Lett.* **77**, 3865 (1996).
- ¹⁸S. G. Louie, S. Froyen, and M. L. Cohen, *Phys. Rev. B* **26**, 1738 (1982).
- ¹⁹F. D. Murnaghan, *Proc. Natl. Acad. Sci. U.S.A.* **3**, 244 (1944).
- ²⁰J. W. Lynn, H. G. Smith, and R. M. Nicklow, *Phys. Rev. B* **8**, 3493 (1973).
- ²¹P. Ballone and G. Galli, *Phys. Rev. B* **42**, 1112 (1990).
- ²²A. Marini, G. Onida, and R. DelSole, *Phys. Rev. B* **64**, 195125 (2001).
- ²³R. Courths and S. Hufner, *Phys. Rep.* **112**, 53 (1984).
- ²⁴G. Nilsson and S. Rolandson, *Phys. Rev. B* **7**, 2393 (1973).
- ²⁵S. Nosé, *J. Chem. Phys.* **81**, 511 (1984).
- ²⁶W. G. Hoover, *Phys. Rev. A* **31**, 1695 (1985).