

See discussions, stats, and author profiles for this publication at:
<https://www.researchgate.net/publication/229347602>

Parallel implementation of the projector augmented plane wave method for charged systems

ARTICLE *in* COMPUTER PHYSICS COMMUNICATIONS · MARCH 2002

Impact Factor: 3.11 · DOI: 10.1016/S0010-4655(01)00413-1

CITATIONS

17

READS

43

4 AUTHORS, INCLUDING:



[Eric J. Bylaska](#)

Pacific Northwest National Laboratory

92 PUBLICATIONS 3,116 CITATIONS

SEE PROFILE



[Marat Valiev](#)

Pacific Northwest National Laboratory

85 PUBLICATIONS 2,139 CITATIONS

SEE PROFILE



[John H Weare](#)

University of California, San Diego

163 PUBLICATIONS 4,865 CITATIONS

SEE PROFILE



Parallel implementation of the projector augmented plane wave method for charged systems

Eric J. Bylaska^{a,*}, Marat Valiev^b, Ryoichi Kawai^c, John H. Weare^b

^a William R. Wiley Environmental Molecular Sciences Laboratory, Pacific Northwest National Laboratory,
P.O. Box 999, Richland, WA 99352, USA

^b Department of Chemistry, University of California, San Diego, La Jolla, CA 92093, USA

^c Department of Physics, University of Alabama at Birmingham, Birmingham, AL 35294, USA

Accepted 23 August 2001

Abstract

A parallel implementation of the projector augmented plane wave (PAW) method with the applications to several transition metal complexes is presented. A unique aspect of our PAW code is that it can treat both charged and neutral cluster systems. We discuss how this is achieved via accurate numerical treatment of the Coulomb Green's function with free space boundary conditions. The strategy for parallelizing the PAW code is based on distributing the plane wave basis across processors. This is a versatile approach and is implemented using a parallel three-dimensional Fast Fourier Transformation (FFT). We report parallel performance analysis of our program and of the three-dimensional FFT's and discuss large-scale parallelization issues of the PAW code. Using a series of transition metal monoxides and dioxides, as well as two iron aqueous complexes, it is shown that a free space PAW code can give structural parameters and energies in good accord with Gaussian based methods. © 2002 Elsevier Science B.V. All rights reserved.

1. Introduction

Many numerical methods have been developed to solve Density Functional Theory (DFT) equations. Three of the more popular methods include Gaussian basis set methods [1], linear-augmented-plane-wave (LAPW) methods [2], and pseudopotential plane-wave (PSPW) methods [3]. All three methods can be made very accurate, and have been capable of predicting structures, frequencies, and energetics for a wide class of compounds. However, the current consensus in the quantum chemistry and condensed matter physics communities is that only the first two of these three methods are straightforward to apply for first-row transition metals [4]. It would be beneficial for PSPW methods to work well for these systems, because Gaussian basis set and LAPW methods lack certain capabilities. In particular, PSPW methods can perform ab initio molecular dynamics extremely efficiently [5], and treat unit cells up to a few hundred atoms [3,6,7]. Another advantage of PSPW methods is their transferability from molecules

* Corresponding author.

E-mail address: eric.bylaska@pnl.gov (E.J. Bylaska).

to surfaces to solids. In contrast, Gaussian-based methods have different basis set requirements for gas and solid phase applications, complicating the transferability of these methods.

The problems with applying the PSPW method to first-row metals are related to their sharply peaked valence states. This leads to prohibitively large plane-wave expansions and as a result PSPW methods become quickly expensive. In addition, because of the significant overlap between core and valence states in transition metals, the PSPW methods are faced with the problem of needing to include nonlinear core corrections for the exchange correlation potential in these systems [8,9]. Despite these difficulties there have been notable successes of PSPW methods applied to first-row transition metals. Calculations by Lindan et al. [10,11], and others [12] have relied on sophisticated optimization schemes for pseudopotentials developed by Rappe [13] and others [14]. However, implementing the capability to handle pseudopotentials for first-row transition metals in this way is an arduous task requiring extensive logarithmic derivative transferability checks and numerous test calculations. There also exists a PSPW approach that is based on Vanderbilt's ultrasoft pseudopotentials [15]. Compared to standard norm-conserving methods, the ultrasoft pseudopotential methods are very efficient for first-row transition metals, and in principle can be made as accurate as an all-electron calculations with a frozen core. However, accurate implementations of Vanderbilt pseudopotentials [16] for calculating first-row transition metals [17–22] still have to rely on smoothing steps and several levels of numerical grids.

Another method that has been used to calculate transition metals is the projector-augmented-wave (PAW) method developed by Blöchl [23]. The PAW method has many similarities to Vanderbilt pseudopotentials, but there are important differences. Perhaps the most important one is that the PAW method is an all-electron approach and therefore no smoothing is required. No part of the wave function is ever discarded and the core states contributions to the exchange-correlation potential are treated exactly within the framework of an all electron calculation. In addition, the PAW method does not require several levels of numerical grids present in most Vanderbilt pseudopotential implementations [16].

In this paper we present an overview of our parallel PAW program and its application to some compounds containing first-row transition-metal atoms. The serial version of this PAW program was previously presented by Valiev and Weare [24]. Several important conclusions came out of that work:

- (i) the PAW method can be used to calculate the structure, vibrational spectra, and energetics of a number of systems that are difficult to treat with norm-conserving PSPW methods [23–27],
- (ii) the accuracy of the PAW method is similar to the accuracy of a local basis calculation,
- (iii) the execution times are similar to standard PSPW methods. Similar conclusions were also reached by others [23,25,26].

The success of the PAW method motivated us to extend the previous work and to develop a parallel PAW program and also to perform more extensive tests on compounds that contain first-row transition-metal atoms.

In Section 2, our implementation of the PAW method is explained. This implementation is derived from previous work of Valiev and Weare [24] and for the most part coincides with that of Blöchl [23]. Unlike the previously reported PAW program by Valiev and Weare [24], this version of the program employs free space boundary conditions to the calculation of Coulomb energy terms. The approach adopted here is described in Section 3. The parallelization scheme and its efficiency are described in Section 4. Our strategy is similar to that used in standard PSPW programs where the plane-wave basis set is distributed across processors. In Section 5 we compare our free-space PAW program with more traditional Gaussian based DFT methods using a series of transition metal monoxides and dioxides as examples. Finally, Section 6 compares the calculations of the one-electron oxidation energies of two iron aqueous complexes, $\text{Fe}^{2+}(\text{H}_2\text{O})$ and $\text{Fe}^{2+}(\text{H}_2\text{O})_6$, and concluding remarks are given in Section 7.

2. PAW methodology

In this section and Appendix A, equations for the PAW energy and PAW energy variation are presented. The emphasis of this section is to show that the equations for the most part are very similar to the ones found in non-

local pseudopotential approaches, with the PAW method adding only a few terms. Full implementation strategies for the PAW method can be found in several references [23–27], with our implementation following that of Valiev and Weare [24].

In the PAW method an invertible linear transformation, Y , is defined which relates the stiff one-electron wavefunctions $\{\Psi_n\}$ to a set of smooth one-electron wavefunctions $\{\tilde{\Psi}_n\}$ representable by a small plane-wave basis:

$$\tilde{\Psi}_n = Y \Psi_n, \quad \Psi_n = Y^{-1} \tilde{\Psi}_n. \quad (1)$$

The transformation Y is defined using a local PAW basis, which consists of atomic orbitals $\{\phi_\alpha^a\}$, smooth atomic orbitals $\{\tilde{\phi}_\alpha^a\}$ and projector functions $\{\tilde{p}_\alpha^a\}$. Here a is the atomic index and α is the orbital index. The smooth atomic orbitals coincide with atomic orbitals outside a defined atomic sphere. The projector functions are constructed such that they are localized within the atomic sphere and in addition are orthogonal to the smooth atomic orbitals:

$$\langle \tilde{p}_\beta^a | \tilde{\phi}_\alpha^a \rangle = \delta_{\alpha\beta}. \quad (2)$$

The invertible linear transformation Y and its inverse can then be written as

$$\begin{aligned} Y &= \mathbf{1} + \sum_a \sum_\alpha (|\tilde{\phi}_\alpha^a\rangle - |\phi_\alpha^a\rangle) \langle p_\alpha^a|, \\ Y^{-1} &= \mathbf{1} + \sum_a \sum_\alpha (|\phi_\alpha^a\rangle - |\tilde{\phi}_\alpha^a\rangle) \langle \tilde{p}_\alpha^a|, \\ \langle \tilde{p}_\alpha^a| &= \sum_\beta \langle \tilde{p}_\alpha^a | \phi_\beta^a \rangle \langle p_\beta^a|. \end{aligned} \quad (3)$$

In order for a PAW calculation to be manageable, it is important for the basis not to be too large, and for the smooth atomic orbitals to be adequately described by a relatively small plane-wave basis. At the same time, for the calculation to be accurate, the basis must also accurately describe the regions near the atomic centers. Procedures used to determine the PAW basis can be found in several places [23,24,26], and for the most part these procedures are the same as the ones used to generate (Vanderbilt) pseudopotentials, with additional Gram–Schmidt steps to enforce the orthogonality relationships of Eq. (2).

The total energy functional in DFT can be written as

$$\begin{aligned} E &= \sum_n f_n \langle \Psi_n | -\frac{1}{2} \nabla^2 | \Psi_n \rangle \\ &+ \frac{1}{2} \iint \frac{(n(\mathbf{r}) - \sum_a n_Z^a(\mathbf{r}))(n(\mathbf{r}') - \sum_{a'} n_Z^{a'}(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\ &+ \int \epsilon_{xc}(n(\mathbf{r})) n(\mathbf{r}) d\mathbf{r}. \end{aligned} \quad (4)$$

Here the total electron density $n(\mathbf{r})$ is given by

$$n(\mathbf{r}) = \sum_n f_n |\Psi_n(\mathbf{r})|^2 + \sum_a n_{\text{core}}^a(\mathbf{r}),$$

and the densities $n_{\text{core}}^a(\mathbf{r})$ and $n_Z^a(\mathbf{r}) = Z^a \delta(\mathbf{r} - \mathbf{R}^a)$ denote respectively the frozen core density and the ion charge density for the a th atom. Using Eqs. (1) and (3), and introducing a localized compensation charge density $n_{\text{cmp}}(\mathbf{r})$ (vide infra), the expression for the energy functional can be written as a sum of three parts:

$$E = \tilde{E} + \sum_a (E^a - \tilde{E}^a). \quad (5)$$

The smooth part of the energy is given by

$$\begin{aligned}
\tilde{E} = & \sum_n f_n \langle \tilde{\Psi}_n | -\frac{1}{2} \nabla^2 | \tilde{\Psi}_n \rangle \\
& + \frac{1}{2} \iint \frac{(\tilde{n}(\mathbf{r}) + n_{\text{cmp}}(\mathbf{r}))(\tilde{n}(\mathbf{r}') + n_{\text{cmp}}(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\
& + \int \epsilon_{\text{xc}}(\tilde{n}(\mathbf{r})) \tilde{n}(\mathbf{r}) d\mathbf{r}.
\end{aligned} \tag{6}$$

The one-center atomic contributions to the energy are given by

$$\begin{aligned}
E^a = & \sum_n \sum_{\alpha\beta} f_n \langle \tilde{\Psi}_n | \tilde{p}_\alpha^a | \phi_\alpha^a | -\frac{1}{2} \nabla^2 | \phi_\beta^a | \tilde{p}_\beta^a | \tilde{\Psi}_n \rangle \\
& + \frac{1}{2} \iint \frac{(n^a(\mathbf{r}) - n_Z^a(\mathbf{r}))(n^a(\mathbf{r}') - n_Z^a(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\
& + \int \epsilon_{\text{xc}}(n^a(\mathbf{r}) + n_{\text{core}}^a(\mathbf{r}))(n^a(\mathbf{r}) + n_{\text{core}}^a(\mathbf{r})) d\mathbf{r},
\end{aligned} \tag{7}$$

$$\begin{aligned}
\tilde{E}^a = & \sum_n \sum_{\alpha\beta} f_n \langle \tilde{\Psi}_n | \tilde{p}_\alpha^a | \tilde{\phi}_\alpha^a | -\frac{1}{2} \nabla^2 | \tilde{\phi}_\beta^a | \tilde{p}_\beta^a | \tilde{\Psi}_n \rangle \\
& + \frac{1}{2} \iint \frac{(\tilde{n}^a(\mathbf{r}) + n_{\text{cmp}}^a(\mathbf{r}))(\tilde{n}^a(\mathbf{r}') + n_{\text{cmp}}^a(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\
& + \int \epsilon_{\text{xc}}(\tilde{n}^a(\mathbf{r})) \tilde{n}^a(\mathbf{r}) d\mathbf{r}.
\end{aligned} \tag{8}$$

Here the auxiliary electron densities $\tilde{n}(\mathbf{r})$, $n^a(\mathbf{r})$ and $\tilde{n}^a(\mathbf{r})$ are given by

$$\begin{aligned}
\tilde{n}(\mathbf{r}) &= \sum_n f_n |\tilde{\Psi}_n(\mathbf{r})|^2, \\
n^a(\mathbf{r}) &= \sum_n f_n \left| \sum_\alpha \langle \tilde{p}_\alpha^a | \tilde{\Psi}_n \rangle \phi_\alpha^a(\mathbf{r}) \right|^2 + n_{\text{core}}^a(\mathbf{r}), \\
\tilde{n}^a(\mathbf{r}) &= \sum_n f_n \left| \sum_\alpha \langle \tilde{p}_\alpha^a | \tilde{\Psi}_n \rangle \tilde{\phi}_\alpha^a(\mathbf{r}) \right|^2.
\end{aligned} \tag{9}$$

The compensation charge density,

$$\begin{aligned}
n_{\text{cmp}}(\mathbf{r}) &= \sum_a n_{\text{cmp}}^a(\mathbf{r} - \mathbf{R}^a), \\
n_{\text{cmp}}^a(\mathbf{r}) &= \sum_{lm} Q_{lm}^a g_l(r) Y_{lm}(\hat{\mathbf{r}}), \\
g_l(r) &= \frac{2^{l+2} r^l e^{-(r/\sigma^a)^2}}{\sqrt{\pi} (2l+1)!! (\sigma^a)^{2l+3}},
\end{aligned} \tag{10}$$

in Eqs. (6) and (8) is constructed such that it has the same multipole moments as the sum of the one-center atomic densities and nuclear charge (aka augmented charge densities):

$$Q_{lm}^a = \int_{\Omega_a} r^l (n^a(\mathbf{r}) - \tilde{n}^a(\mathbf{r}) - n_Z^a(\mathbf{r})) Y_{lm}^*(\hat{\mathbf{r}}) d\mathbf{r}.$$

The purpose of the compensation charge density is to aid the Coulomb energy calculation of the PAW transformed density. The PAW transformed density contains localized densities (augmented charge densities) on

different atoms whose long-range Coulomb interaction can be represented by a multipole expansion. Using the compensation charge density avoids the several levels of numerical grids present in most Vanderbilt pseudopotential implementations. However, it must be understood that if the PAW basis is incomplete then the introduction of the compensation charge density formally adds an error term to Eq. (5). Typically this error is expected to be small and goes to zero as the basis size increases [23,24].

Blöchl [23] has also suggested augmenting the total energy expression, Eq. (5), with additional auxiliary pseudopotential term:

$$E^{\text{PS}} = \int \left(\sum_a \bar{v}^a(\mathbf{r}) \right) \tilde{n}(\mathbf{r}) \, d\mathbf{r} - \sum_a \int \bar{v}^a(\mathbf{r}) \tilde{n}^a(\mathbf{r}) \, d\mathbf{r}.$$

The purpose of this term to remove the ghost state problem when the PAW basis is incomplete. In short, the ghost state problem is due to the fact that the PAW transformation with an incomplete basis is not guaranteed to have atomic eigenfunctions that are energetically increasing (for a given quantum number l) with an increasing number of nodal surfaces [28]. The pseudopotential $\bar{v}^a(\mathbf{r})$ has to be localized about the a th atom and its contribution to the total PAW energy vanishes if the basis is complete.

The minimization of the PAW energy Eq. (5) with respect to the smooth one-electron orbitals $\{\tilde{\Psi}_n\}$ can be performed by several iterative methods. All of these methods require the calculation of the gradient

$$\frac{\delta E}{\delta \langle \tilde{\Psi}_n |} = \left(\tilde{H} + \sum_a (H^a - \tilde{H}^a) \right) | \tilde{\Psi}_n \rangle$$

and the overlap matrix

$$\begin{aligned} S_{ij} &= \langle \Psi_i | \Psi_j \rangle \\ &= \langle \tilde{\Psi}_i | \tilde{\Psi}_j \rangle + \sum_a \sum_{\alpha\beta} \langle \tilde{\Psi}_i | \tilde{p}_\alpha^a \rangle (\langle \phi_\alpha^a | \phi_\beta^a \rangle - \langle \tilde{\phi}_\alpha^a | \tilde{\phi}_\beta^a \rangle) \langle \tilde{p}_\beta^a | \tilde{\Psi}_j \rangle. \end{aligned} \quad (11)$$

The smooth part of the gradient

$$\tilde{H} \tilde{\Psi}_n(\mathbf{r}) = \left[-\frac{1}{2} \nabla^2 + \int \frac{\tilde{n}(\mathbf{r}') + n_{\text{cmp}}(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \, d\mathbf{r}' + v_{\text{xc}}[\tilde{n}] + \sum_a \bar{v}^a(\mathbf{r}) \right] \tilde{\Psi}_n(\mathbf{r}), \quad (12)$$

and the overlap matrix can be evaluated using standard techniques found in pseudopotentials methods. For the most part, calculating the local part of the gradient is not much different than calculating a non-local pseudopotential

$$(H^a - \tilde{H}^a) | \tilde{\Psi}_n \rangle = \sum_{\alpha\beta} | \tilde{p}_\alpha^a \rangle (G_n^a)_{\alpha\beta} \langle \tilde{p}_\beta^a | \tilde{\Psi}_n \rangle, \quad (13)$$

since most of the one center atomic matrix $(G_n^a)_{\alpha\beta}$ can be precomputed. The exact form of $(G_n^a)_{\alpha\beta}$ is given in Appendix A.

3. Implementing free-space boundary conditions

In this section, we present an algorithm for calculating the Coulomb energy terms using free-space boundary conditions instead of periodic boundary conditions. This allows our program to accurately treat both charged and neutral isolated molecular and cluster systems. We have previously used this technique with our PSPW programs [29,30].

In the PAW approach one is faced with the task of calculating several Coulomb energies and gradients (e.g., see Eqs. (6), (12), and (A.3) that are of the general form

$$\begin{aligned}
E_{\text{Coulomb}} &= \frac{1}{2} \iint \rho_1(\mathbf{r}) g(\mathbf{r}, \mathbf{r}') \rho_2(\mathbf{r}') d\mathbf{r} d\mathbf{r}', \\
V_H(\mathbf{r}) &= \int g(\mathbf{r}, \mathbf{r}') \rho_2(\mathbf{r}') d\mathbf{r}', \\
g(\mathbf{r}, \mathbf{r}') &= \frac{1}{|\mathbf{r} - \mathbf{r}'|}.
\end{aligned} \tag{14}$$

When evaluated for a neutral system using periodic boundary conditions these terms can be written as

$$\begin{aligned}
E_{\text{Coulomb}} &= \frac{\Omega}{2} \sum_{\mathbf{G} \neq 0} \rho_1(-\mathbf{G}) \frac{4\pi}{|\mathbf{G}|^2} \rho_2(\mathbf{G}), \\
V_H(\mathbf{G}) &= \frac{4\pi}{|\mathbf{G}|^2} \rho_2(\mathbf{G}),
\end{aligned} \tag{15}$$

where Ω is the volume of the supercell enclosing the system. This treatment of the Coulomb energy terms effectively means that we are considering an infinite collection of replicated supercells. Such description is not entirely accurate even for isolated neutral cluster systems because of the residual (dipole or quadrupole) interaction between the unit cells.

For isolated neutral or charged cluster systems it is more appropriate to use free space boundary conditions. Free-space boundary conditions can be implemented, provided the density decays to zero at the edge of the supercell, by restricting the integration to just one isolated supercell, Ω .

$$\begin{aligned}
E_{\text{Coulomb}} &= \frac{1}{2} \int_{\Omega} \int_{\Omega} \rho_1(\mathbf{r}) g(\mathbf{r}, \mathbf{r}') \rho_2(\mathbf{r}') d\mathbf{r} d\mathbf{r}', \\
V_H(\mathbf{r}) &= \int g(\mathbf{r}, \mathbf{r}') \rho_2(\mathbf{r}') d\mathbf{r}', \\
g(\mathbf{r}, \mathbf{r}') &= \frac{1}{|\mathbf{r} - \mathbf{r}'|}.
\end{aligned} \tag{16}$$

This essentially defines a modified Coulomb interaction

$$g(\mathbf{r}, \mathbf{r}') = \begin{cases} \frac{1}{|\mathbf{r} - \mathbf{r}'|}, & \text{for } \mathbf{r}, \mathbf{r}' \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

Hockney and Eastwood showed that an interaction of the form (17) could still be used in conjunction with the Fast-Fourier Transform convolution theorem [31,32]. The interaction between neighboring supercells, in their algorithm, is removed by padding the density with an external region of zero density. In the specific case of a density defined in cubic supercell of length L , the density is extended to a cubic supercell of length $2L$, where the original density is defined as before on the $[0, L]^3$ domain and the remainder of the $[0, 2L]^3$ domain is set to zero. The extended grid is 8 times larger than the conventional grid. The Coulomb potential is calculated by convoluting the density with the Green's function kernel on the extended grid. The density on the extended grid is defined by expanding the conventional grid to the extended grid and putting zeros where the conventional grid is not defined. After the aperiodic convolution, the free-space potential is obtained by restricting the extended grid to the conventional grid.

In his original work Hockney suggested that the Coulomb kernel on the extended grid could be defined by

$$g(\mathbf{r}, \mathbf{r}') = \begin{cases} \frac{1}{|\mathbf{r} - \mathbf{r}'|}, & \text{otherwise,} \\ \frac{\text{const}}{h}, & \text{for } \mathbf{r} = \mathbf{r}', \end{cases} \tag{18}$$

where h^3 is the constant volume of subintervals, defined by the unit cell divided by the number of conventional FFT grid points [31]. Hockney suggested the constant at $|\mathbf{r}| = 0$ to be between 1 and 3. Barnett and Landman [33] in their implementation defined the constant to be

$$\frac{1}{h^2} \int_{h^3} \frac{1}{|\mathbf{r}|} d\mathbf{r} \approx \begin{cases} 2.380077, & \text{for SC lattice,} \\ 0.910123, & \text{for FCC lattice,} \\ 1.447944, & \text{for BCC lattice.} \end{cases} \quad (19)$$

Regardless of the choice of the constant, the singular nature of $g(\mathbf{r})$ in real-space can lead to significant numerical error. James addressed this problem somewhat by expanding the Coulomb kernel to higher orders in real space [34].

The convolution theorem suggests that defining $g(\mathbf{r})$ in reciprocal space will lead to a much higher accuracy. A straightforward definition in reciprocal space is

$$g(\mathbf{r}) = \sum_{\mathbf{G}} g_{\text{uniform}}(\mathbf{G}) e^{i\mathbf{G}\mathbf{r}}, \quad (20)$$

$$g_{\text{uniform}}(\mathbf{G}) = \frac{1}{h^3} \int_{\Omega'} \frac{e^{-i\mathbf{G}\mathbf{r}'/2}}{|\mathbf{r}'|} d\mathbf{r}',$$

where Ω' is the volume of the extended unit cell and h^3 is the volume of the unit cell divided by the number of conventional FFT grid points. The reciprocal space definition gains accuracy because the singularity at $\mathbf{r} = \mathbf{r}'$ in Eq. (15) is analytically integrated out.

Even when Eq. (20) is used to define the kernel, a slight error in the calculated electron–electron Coulomb energy will always remain present. This error is a result of the small discontinuity introduced in the definition of the extended density, when the extended density is forced to be zero in the extended region outside of Ω . However, this discontinuity is small, since we are interested in densities that decay to zero within Ω . Eq. (20) could be calculated numerically, however we have found that alternative definitions can be used with little loss of numerical accuracy. In our earlier work we suggested that the cutoff Coulomb kernel could be defined as [29]

$$g(\mathbf{r}) = \begin{cases} \sum_{\mathbf{G}} g_a(\mathbf{G}) e^{i\mathbf{G}\mathbf{r}}, & \text{for } |\mathbf{r}| \leq R_{\text{max}} - \delta, \\ \frac{1}{|\mathbf{r}|}, & \text{otherwise,} \end{cases}$$

$$g_a(\mathbf{G}) = \begin{cases} \frac{2\pi(R_{\text{max}})^2}{h^3}, & \text{for } \mathbf{G} = \mathbf{0}, \\ \frac{4\pi}{h^3|\mathbf{G}|^2} (1 - \cos(|\mathbf{G}|^2 R_{\text{max}})), & \text{otherwise,} \end{cases} \quad (21)$$

$$R_{\text{max}} = \begin{cases} L, & \text{for SC lattice,} \\ \frac{\sqrt{2}}{2}L, & \text{for FCC lattice,} \\ \frac{\sqrt{3}}{2}L, & \text{for BCC lattice,} \end{cases}$$

where δ is small constant.

Other forms have been suggested and could also be used [6,35,36]. The Fourier-represented kernels improve the integration accuracy by removing the singularity at $\mathbf{r} = \mathbf{r}'$ in a trapezoidal integration [37]. A disadvantage of the kernel defined by (22) is that only regular shaped cells can be used. To extend this method to irregular shaped cells, a short and long range decomposition based on a Gaussian function can be used

$$g(\mathbf{r}) = g_{\text{short-range}}(\mathbf{r}) + g_{\text{long-range}}(\mathbf{r}),$$

$$g_{\text{short-range}}(\mathbf{r}) = \sum_{\mathbf{G}} g_{\text{short-range}}(\mathbf{G}) e^{i\mathbf{G}\mathbf{r}},$$

$$g_{\text{short-range}}(\mathbf{G}) = \begin{cases} \frac{4\pi}{h^3|\mathbf{G}|^2} (1 - e^{-|\mathbf{G}|^2/4\epsilon^2}), & \text{for } |\mathbf{G}| \neq \mathbf{0}, \\ \frac{\pi}{h^3\epsilon^2}, & \text{for } |\mathbf{G}| = \mathbf{0}, \end{cases}$$

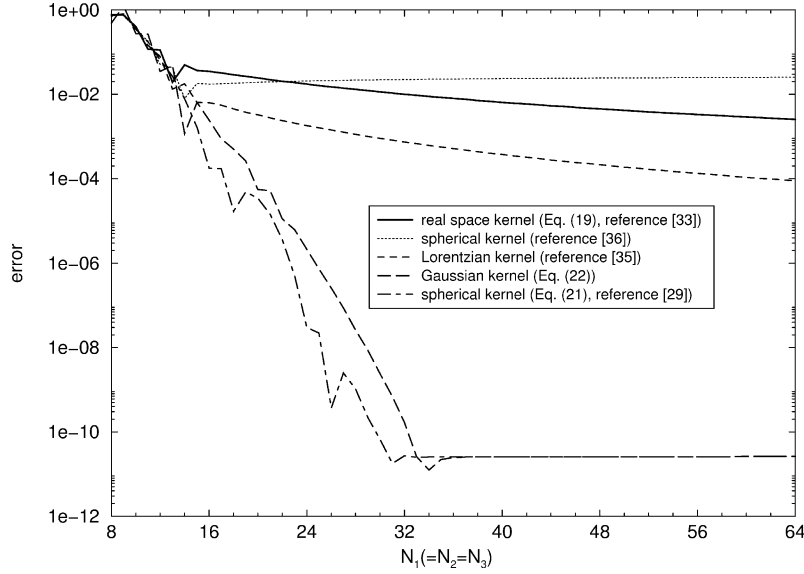


Fig. 1. Accuracy for several cutoff Coulomb kernels is shown. The test density used in these calculations is composed of three normalized Gaussian functions located at (8.0, 8.0, 10.0), (12.0, 12.0, 12.0), and (8.0, 13.0, 10.0) on the $\Omega = [0.0, 20.0]^3$ domain. The normalized Gaussians have decay rates of 0.4, 0.4 and 0.5, respectively. The free-space electron–electron Coulomb energy for this test density is $E = 1.4043497402$.

$$g_{\text{long-range}}(\mathbf{r}) = \begin{cases} \frac{\text{erf}(\epsilon|\mathbf{r}|)}{h^3}, & \text{for } |\mathbf{r}| \neq 0, \\ \frac{2\epsilon}{\sqrt{\pi}}, & \text{for } |\mathbf{r}| = 0. \end{cases} \quad (22)$$

We have found this kernel to give very high accuracy, even for highly non-cubic supercells. Marx and Hutter [6] recently proposed the use of this kernel as well. Other kernel definitions are possible. Kawai [35], in his PSPW programs uses a short and long range decomposition based on a Lorentzian function. The accuracy for many of the cutoff Coulomb kernels discussed in this section is shown in Fig. 1.

4. Parallel implementation and efficiencies

Many groups have shown that plane-wave programs can be made to run in parallel relatively easily, and several parallel implementations of plane-wave programs have been reported to have an excellent scaling to a large number of processors [6,38–40].

There are several ways to parallelize a plane-wave program. For many solid-state calculations the computation can be distributed over the Brillouin zone sampling space. However, this approach is limited by the size of the Brillouin zone sampling space, which is usually very small. Another approach is to distribute the one-electron orbitals across processors [40]. This approach has been shown to scale quite well. Unfortunately, this method will not work for jobs with large cutoff energy requirements (i.e. using large numbers of plane-waves to describe the one-electron orbitals) on parallel computers that have nodes with a small amount of memory. This is because an entire one-electron orbital must be stored on each task. Finally, another straightforward parallelization approach is to do a spatial decomposition of the one-electron orbitals [6,38,39]. This approach is the most versatile, and easily implemented. However, a parallel three-dimensional Fast Fourier Transform (FFT) must be used.

Our parallel PAW program is written in Fortran 90 and MPI, and is parallelized using a spatial decomposition approach, i.e. distributing the plane-wave basis set across tasks. Here we discuss the parallel decomposition for a Γ -point program. Purely Γ -point programs are faster and require less memory than full band-structure programs, since restricting the calculation to just the Γ -point allows one to use half the plane-waves of a full band-structure calculation. The parallel FFT subroutine was implemented using a slab decomposition of the three-dimensional FFT block as shown in Fig. 2. Our parallel FFT subroutine is built from one-dimensional FFT subroutines and a parallel matrix transpose. The program package FFTPACK [41] is used to perform the one-dimensional FFT's.

Special care must be taken in the implementation of Γ -point programs, because real-to-complex three-dimensional FFT's have specific placement requirements for data in complex (reciprocal) space. The real-space data for a $n_1 \times n_2 \times n_3$ FFT block must be stored in a $A[1:n_1+2, 1:n_2, 1:n_3]$ real array and reciprocal-space is stored in a $B[1:\frac{n_1}{2}+1, 1:n_2, 1:n_3]$ complex array. The constraints required by the real-to-complex three-dimensional FFT for the real array and complex array are shown in Eq. (23). The extra space is defined to be zero in real-space, and the extra space is taken up by the real-space symmetry constraint imposed in complex space, $B_k = B_{-k}^*$

$$\left\{ \begin{aligned} &A[i_1, i_2, i_3] \equiv 0: \forall i_1 = n_1 + 1, n_1 + 2; i_2 = 1:n_2; i_3 = 1:n_3 \}, \\ &B[1, n_2 - k_2, n_3 - k_3] \equiv B[1, k_2, k_3]: \forall k_2 = 1:\frac{n_2}{2} - 1; k_3 = 1:\frac{n_3}{2} - 1 \}. \end{aligned} \right. \quad (23)$$

The spatial decomposition of the plane-waves in real-space is blocked into yz -plane slabs, matching the decomposition of the three-dimensional FFT block. Storing reciprocal-space is complicated by the fact that only plane-waves up to a certain cutoff energy are used in calculations. Matching the decomposition of the three-dimensional FFT block to the needed reciprocal-space can lead to a load imbalance in our PAW program. This is because in reciprocal-space only a hemisphere, of radius E_{cut} , contained within the three-dimensional FFT block is needed and saved in the program, as shown in Fig. 3. Balancing the reciprocal-space distribution is important because the majority of computations done in the program are performed in this space. To reduce this load imbalance we averaged out the space stored by each processor. This averaging requires a small amount of additional overhead communication between processors, but the additional overhead cost is minimal because only a small amount of data is shifted between a few of the tasks.

In our current implementation, the computationally most demanding part of the PAW program is the generation of the exchange-correlation matrix elements for the one-center atomic contributions. These calculations are not performed on the FFT grid and are not automatically parallel with our spatial grid parallel distribution. Rather,

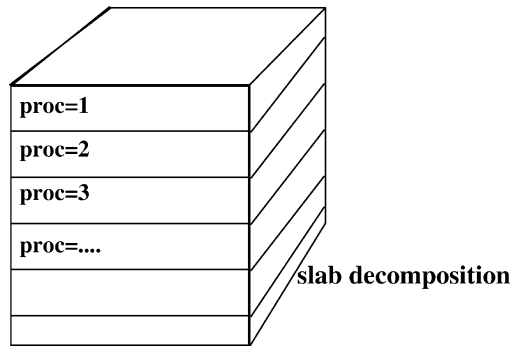


Fig. 2. Slab decomposition of the FFT grid.

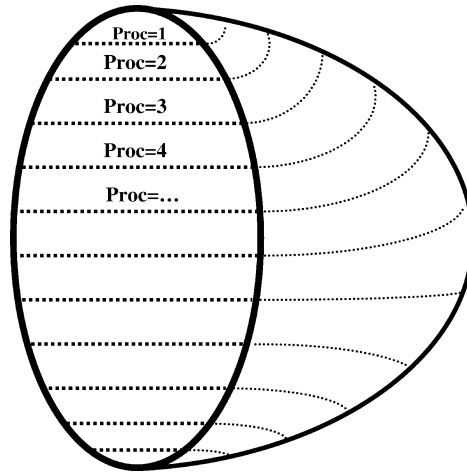


Fig. 3. The unbalanced reciprocal-space decomposition.

calculations are performed on localized grids about each atom. These localized grids are represented in spherical coordinates, and our parallel strategy is to decompose the radial grid across processors. This distribution has minimal overhead cost and is expected to scale perfectly to a large number of processors.

The overall performance of our program is shown in Table 1. Calculations were performed on a Linux cluster composed of dual processor 500 MHz Pentium nodes connected via a Gigaset cLAN high-speed network. The program was compiled using Portland Group Fortran 90 compiler, version 3.1 [42], and used the MPI/Pro library, version 1.5 [43], for message passing. The performance of the program is quite good with an overall parallel efficiency to 64 processors being perfect. However, not every part the program scales perfectly. For illustrative purposes, we have broken up the total electronic update step time into four parts: t_1 being dominated by FFT calculations, t_2 being dominated by Lagrange multiplier or orthonormality calculations, t_3 being the time elapsed generating exchange-correlation matrix elements for the one-center atomic contributions (Eq. (A.4)), and t_4 being dominated by applying the projector functions (Eq. (13)). The efficiency of t_1 shows that Fast Fourier Transforms are the most inefficient part of the program. However, these inefficiencies are damped out due to the fact that these parts of the program make up less than 10% of the overall computation. The most time consuming part of the calculation is the generation of the exchange-correlation matrix elements for the one-center atomic contributions seen in t_3 , and the next largest parts of the calculations are the Lagrange multipliers and applying the projector functions, seen in t_2 and t_4 .

A limitation of our parallelization strategy is that the FFT block is divided along the third dimension, or slab decomposed. As a consequence, the PAW program cannot take advantage of more processors than the size of the third dimension (e.g., a $64 \times 64 \times 64$ FFT block can use at most 64 processors). In terms of just the parallel three-dimensional FFT performance this is not much of a limitation. It is difficult to get parallel performance beyond that of slab decomposed a FFT [44], because the data communicated in performing any type of parallel three-dimensional FFT is on the same order as the number of floating point operations. However, a finer scaled decompositions can still be of utility since other parts of the PAW program are expected to scale well to a large number of tasks (see Table 1). For example, if we use the performance numbers reported Table 1, then we can estimate the parallel efficiencies to be 90% for 128 tasks, 73% for 256 tasks, and 53% for 512 tasks. These estimated parallel efficiencies are based on the assumptions that the time spent in the FFT routines remains constant and the time spent in the other routines scales perfectly.

Table 1

Timing in seconds (parallel efficiencies) for $\text{Fe}^{3+}(\text{H}_2\text{O})_6$ cluster calculations. Determined from calculations on Linux cluster made up of dual processor 500 MHz Pentium nodes connected via a Gigaset high-speed network. (t1 = FFT calculations, t2 = orthonormality calculations, t3 = generation of exchange-correlation matrix elements, t4 = projector function calculations)

Tasks	Total update step	t1	t2	t3	t4
8	143.14	10.31	23.97	86.31	22.55
16	68.75 (104%)	7.21 (71%)	11.60 (103%)	38.14 (113%)	11.80 (95%)
32	32.85 (109%)	2.75 (94%)	5.57 (108%)	18.74 (115%)	5.79 (97%)
64	17.58 (102%)	2.35 (55%)	2.55 (118%)	9.83 (110%)	2.85 (99%)

Table 2

Timing in seconds for complex-to-complex FFT's. Determined from calculations on Linux cluster made up of dual processor 500 MHz Pentium nodes connected via a Gigaset high-speed network

Grid	Tasks	Hilbert	Slab	Grid	Tasks	Hilbert	Slab
64 ³	1	9.10E−01	9.51E−01	128 ³	1	7.44E+00	7.98E+00
	2	6.71E−01	5.90E−01		2	5.46E+00	5.48E+00
	4	3.37E−01	3.17E−01		4	2.74E+00	2.64E+00
	8	1.97E−01	1.66E−01		8	1.65E+00	1.39E+00
	16	1.01E−01	1.03E−01		16	8.66E−01	7.45E−01
	32	7.16E−02	4.10E−02		32	5.87E−01	4.16E−01
	64	5.48E−02	2.52E−02		64	3.02E−01	2.35E−01
	80	4.98E−02	–		80	–	–
	96	3.78E−02	–		96	–	–
	128	1.49E−02	–		128	1.62E−01	9.86E−02
	144	1.78E−02	–		144	2.37E−01	–
	160	1.84E−02	–		160	2.85E−01	–

A massively parallel PAW program will be more straightforward to develop if the spatial decomposition used by the parallel FFT is used to represent real-space and reciprocal-space. However, using a finer scaled decomposed parallel FFT instead of a slab decomposed FFT will only be viable if its performance is at least as good as the slab decomposed FFT. In Table 2 we report timings for a complex to complex FFT spatially decomposed along the second and third dimension, and for a slab decomposed complex to complex FFT. To generate the two-dimensional spatial decomposition we block-mapped the tasks to a two-dimensional Hilbert curve spanning the grid of the second and third dimensions (see Fig. 4). The two-dimensional Hilbert parallel FFT was built from one-dimensional FFTs and a parallel block rotation, and the program package FFTPACK was used to perform the one-dimensional FFTs. The Hilbert decomposition reduces the latency of the parallel block rotation to be proportional to the square root of the number of tasks used. Table 2 shows the timings of the two different parallel FFTs to be about the same. Unfortunately, no improvements are seen with using a large number of tasks in the two-dimensional Hilbert parallel FFT. However, the timings do not become worse. Consequently, using a two-dimensional Hilbert parallel FFT is a viable strategy for a massively parallel implementation of the PAW program.

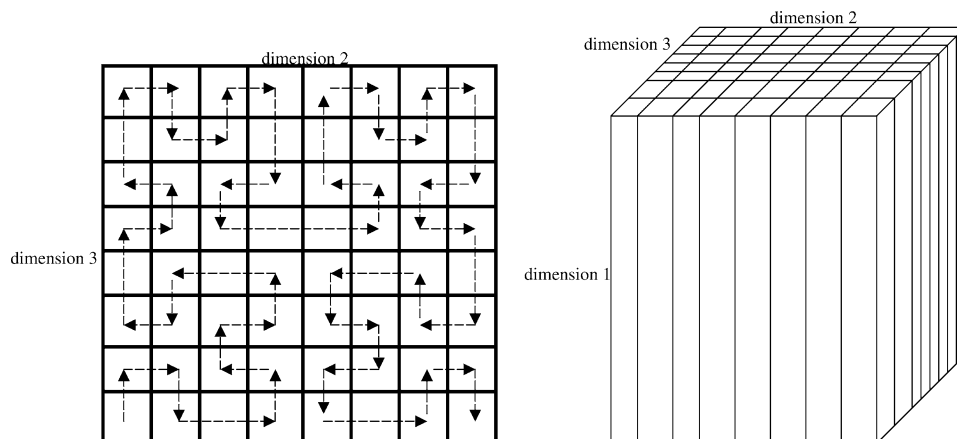


Fig. 4. The two-dimensional Hilbert decomposition of the FFT grid.

5. First-row transition-metal monoxides and dioxides

In this section we compare results from our free-space PAW program and the Gaussian basis set program NWChem [45] for a series of first-row transition metal monoxides and dioxides, ScO, TiO, VO, CrO, MnO, FeO, TiO₂, VO₂, CrO₂, and FeO₂. These small molecules, which are well characterized experimentally, are a stringent test of the reliability of an ab initio method because they have a large number of low lying states, many with high spin multiplicity.

The parallel PAW electronic structure calculations used a PAW basis consisting of a $1s^2 2s^2 2p^6$ core with 3s, 3p, 3d, 4s and 4p valence orbitals for the transition metals, and a $1s^2$ core with 2s, 2p, 3d, 3s, and 3p valence orbitals for the oxygen atom. Semi-core states, 3s and 3p, were included among the valence orbitals for the transition metals because these states extend into the bonding region, and their omission can cause ghost-state problems. The smooth part of the density and one-electron orbitals were expanded using a plane-wave basis set with a kinetic energy cutoff of 60 Ry. The lattice constant was chosen to be sufficiently large to ensure that the density decays to zero by the supercell edges. These calculations used a $26 \times 26 \times 26$ (a.u.) simple cubic supercell. Also, the Coulomb interactions were solved with free-space boundary conditions using the algorithm described in Section 3. Here only the local spin density approximation (LSDA) parameterized by Vosko et al. [46] was used.

The PAW basis used in these calculations have several parameters which are given in Table 3. σ^a is the width of the compensation charge density, and r_k^a , λ^a , R^a , and V_0^a are parameters used to generate the smooth atomic orbitals and projector functions. The smooth atomic orbitals were generated using a Hamann pseudopotential procedure, omitting the norm-conservation step. Here r_k^a and λ^a are the cutoff function parameters used for Hamann pseudopotentials, and R^a is the radius where the atomic orbitals and smooth atomic orbitals are forced to match. To improve the scattering properties of the PAW basis a short-range potential, with coefficient V_0^a , is added to the generation of the projector functions. A detailed discussion of these parameters and generation of PAW basis in general is given in Refs. [23,24].

The Gaussian based electronic structure calculations were done using the NWChem program package [45]. As with the PAW calculations, the local density approximation (LDA) parameterized by Vosko et al. [46] was used for the exchange-correlation functional. The Kohn–Sham orbitals were expanded using the DZVP2 DFT orbital basis set [47], without the use of a Coulomb fitting basis. The exchange-correlation energy and potential were evaluated on a numerical grid. The integration scheme that we chose, partitions the density into atomic contributions [48] with the density integrated using an Euler–McLaurin radial quadrature and a Lebedev angular quadrature [49].

Table 3

Parameters used in the construction of PAW basis set. Numbers in round brackets in the second column denote the energy of the scattering states

Element	Basis	R^a	r_k^a	λ^a	σ^a	V_0^a
H	1s, 2p, 3d	0.90	0.35	6.00	0.07	0
O	2s, 2p, 3d, 3s, 3p	1.8	0.71	6.00	0.15	0
Sc	3s, 3p, 3d, 4s, 4p, 4d (0.1)	2.00	1.18	6.00	0.39	−3.5
Ti	3s, 3p, 3d, 4s, 4p, 4d (0.1)	2.00	1.18	6.00	0.28	−3.5
V	3s, 3p, 3d, 4s, 4p, 4d (0.1)	2.00	1.18	6.00	0.28	−3.5
Cr	3s, 3p, 3d, 4s, 4p, 4d (0.1)	1.80	1.18	6.00	0.38	−3.5
Mn	3s, 3p, 3d, 4s, 4p, 4d (0.1)	2.00	1.18	6.00	0.34	−3.5
Fe	3s, 3p, 3d, 4s, 4p, 4d (0.1)	2.00	1.18	6.00	0.34	−3.5

Table 4

Structural parameters for transition metals monoxides

System (state)	Method	R_e (Å)	ω_e (cm ^{−1})	$\omega_e X_e$ (cm ^{−1})	D_e (eV)
⁵⁴ Sc ¹⁶ O ($X^2\Sigma^+$)	PAW	1.658	991	3.3	9.09
	NWChem	1.673	990	3.5	8.79
	Exp	1.6682	964.95	4.2	7.02
⁴⁸ Ti ¹⁶ O ($X^3\Delta_r$)	PAW	1.620	1020	3.6	9.03
	NWChem	1.615	1017	3.6	8.92
	Exp	1.6202	1009.02	4.498	6.93
⁵¹ V ¹⁶ O ($X^4\Sigma^-$)	PAW	1.585	992	3.7	8.36
	NWChem	1.574	992	3.7	8.30
	Exp	1.5893	1011.3	4.86	6.46
⁵² Cr ¹⁶ O ($X^5\Pi$)	PAW	1.585	960	5.0	5.68
	NWChem	1.600	910	4.2	6.05
	Exp	1.615	898.4	6.7	4.46
⁵⁵ Mn ¹⁶ O ($X^6\Sigma$)	PAW	1.620	939	4.2	6.51
	NWChem	1.613	944	4.3	6.44
	Exp	1.76	839.6	4.7	3.75
⁵⁶ Fe ¹⁶ O ($X^5\Delta$)	PAW	1.599	935	4.1	6.65
	NWChem	1.591	947	4.3	6.55
	Exp	1.57	965	—	4.26

The structural parameters for the ScO, TiO, VO, CrO, MnO, and FeO transition metal monoxides obtained using our free-space PAW program and the Gaussian basis set program NWChem are shown Table 4. To obtain the structural parameters we performed a series of energy calculations with inter-nuclear distances between 1.500–2.100 Å at 0.025 Å intervals and fit the binding energy curves to a Morse potential using the Levenberg–Marquardt

Table 5
Structural parameters for transition metals dioxides

System (state)	Method	R_{MO} (Å)	θ_{OMO} (deg)	D_e (eV)
TiO ₂ (¹ A ₁)	PAW	1.653	110	1.71
	NWChem	1.643	108.9	1.69
VO ₂ (² A ₁)	PAW	1.613	109.3	1.59
	NWChem	1.602	108.3	1.57
CrO ₂ (³ A ₁)	PAW	1.585	120.5	1.33
	NWChem	1.590	120.3	1.34
FeO ₂ (³ A ₁)	PAW	1.572	132.1	1.30
	NWChem	1.569	132.3	1.27

method [50,51]. Total energy calculations of the ground-state atoms were used to generate binding energies. The agreement in structural parameters was quite good between the two methods and with experiment. The worst case difference was for the CrO molecule with absolute differences in distance, frequency, and binding energy being 0.015 Å, 50 cm⁻¹, and 0.37 eV between the PAW/LDA/60 Ry PAW and NWChem/LDA/DZVP2 results.

Table 5 shows the calculated structural parameters for the transition metal dioxides, TiO₂, VO₂, CrO₂, and FeO₂. Again the agreement between the two methods was quite good. The worst case difference was for the TiO₂ molecule with absolute differences in distance, angle, and atomization energy being 0.01 Å, 1.1°, and 0.02 eV between the PAW/LDA/60 Ry PAW and NWChem/LDA/DZVP2 results.

6. Aqueous iron complexes

Iron (II) is involved in a variety of environmentally important redox roles, and is linked to global cycles of oxygen, carbon, and sulfur. One such environmentally important reduction process is that surface functional groups of many Fe(III)-oxides are known to dissolve in highly reducing conditions to aqueous Fe(II). Aqueous Fe(II) is thought to be an important reducing agent for several environmental pollutants, and understanding its formation along with its interaction with pollutants in highly reducing environments are important research areas in environmental chemistry. Clearly to study these kind of processes with an ab initio method, we need to have a method that can move smoothly between molecular systems and periodic systems (such as slabs or bulk solids). Gaussian-based methods have different basis set requirements for gas and solid phase applications, complicating the transferability of these methods. Plane-wave methods, on the other hand, do not have such problems and have been shown to work quite well on some of the oxyhydroxide minerals of interest, and have recently been applied to aqueous metal complexes. It is therefore of interest to test whether the PAW methods are capable of describing aqueous iron complexes. Here we compare the calculations of the one-electron oxidation energies of two iron aqueous complexes, Fe²⁺(H₂O) and Fe²⁺(H₂O)₆. The free-space PAW calculations and the Gaussian basis set calculations were setup as in the previous section.

The geometries of aqueous iron complexes (see Table 6), Fe²⁺(H₂O), Fe³⁺(H₂O), Fe²⁺(H₂O)₆, and Fe³⁺(H₂O)₆, were consistently optimized with our PAW program and the NWChem program. The one-electron oxidation energies were obtained by taking their total energy differences. Table 7 shows that results for the oxidation of the aqueous iron complexes are nearly identical between the two ab initio methods.

Table 6
Geometries for Fe–water compounds. Bond distances are in angstrom and bond angles are in degrees

	Parameter	PAW/	NWChem	Exp
H ₂ O	<i>r</i> (OH)	0.974	0.975	0.957
	θ (HOH)	104.7	105.7	104.5
⁵ X Fe ²⁺ (H ₂ O)	<i>r</i> (FeO)	1.86	1.86	
	<i>r</i> (OH)	1.00	1.01	
	θ (FeOH)	126	126	
⁶ X Fe ³⁺ (H ₂ O)	<i>r</i> (FeO)	1.94	2.07	
	<i>r</i> (OH)	1.06	1.07	
	θ (FeOH)	126	126	
⁵ X Fe ²⁺ (H ₂ O) ₆	<i>Av r</i> (FeO)	2.08	2.07	
	<i>Av r</i> (OH)	0.98	0.98	
	<i>Av θ</i> (FeOH)	126	126	
⁶ X Fe ³⁺ (H ₂ O) ₆	<i>r</i> (FeO)	2.01	2.01	
	<i>r</i> (OH)	0.99	0.99	
	θ (FeOH)	126	126	

Table 7
Calculated ionization energies for Fe–water compounds. Zero point energies are not included

	Method	ΔE (eV)
Fe ²⁺ (H ₂ O)	PAW	25.5
	NWChem	26.1
Fe ²⁺ (H ₂ O) ₆	PAW	15.9
	NWChem	16.2

7. Conclusion

In this paper, our newly developed parallel free-space PAW program is described. It is shown to be efficient to a large number of processors, even for modest system sizes. At this state in our development our PAW program is approximately 2 times slower than PSPW codes. This added cost comes from additional calculations needed to evaluate the exchange-correlation potential on the radial grid around each atom. A unique aspect of our PAW program is the use of free-space boundary conditions. Free-space boundary conditions greatly facilitate the transferability of the PAW program, allowing the calculation of charged and highly polar molecules.

Results obtained with our free-space PAW program were shown to agree quite well with standard Gaussian basis set methods. Using both methods we calculated the structural parameters of a series of transition metal monoxides and dioxides, ScO, TiO, VO, CrO, MnO, FeO, TiO₂, VO₂, CrO₂, FeO₂, and the one-electron oxidation energies of two iron aqueous complexes, Fe²⁺(H₂O) and Fe²⁺(H₂O)₆. These calculations demonstrate that the PAW approach

can provide accuracy similar to more traditional Gaussian based first principles methods in non-trivial transition metal complexes.

Acknowledgements

A significant portion of this research was supported by Laboratory Directed Research and Development funds from Pacific Northwest National Laboratory (PNNL). Additional support also came from the Office of Naval Research through research Grant No. N00014-97-1-0751, the Department of Energy Office of Basic Energy Sciences, Engineering and Geosciences Division contract No. 18328, and the Department of Energy Environmental Management Sciences Program (EMSP) contract No. 30944. The Pacific Northwest National Laboratory is operated by the Battelle Memorial Institute. Some of the calculations were performed on the IBM SP computer in the Molecular Science Computing Facility in the Environmental Molecular Sciences Laboratory (EMSL) at PNNL. EMSL operations are supported by the DOE's Office of Biological and Environmental Research. We are also grateful to the National Energy Research Supercomputing Center for a generous grant of computer time. Finally the authors would like to thank E. Apra for giving us help on carrying out the Gaussian basis set calculations with NWChem.

Appendix A. One-center atomic Hamiltonian matrix elements

In this appendix the matrix elements contained in the one-center atomic Hamiltonian matrix are presented

$$\begin{aligned}
 (G_n^a)_{\alpha\beta} = & (t^a)_{\alpha\beta} + (v_Z^a)_{\alpha\beta} + (v_{\text{core}}^a)_{\alpha\beta} + (\bar{v}^a)_{\alpha\beta} \\
 & + \sum_{lm} \mathcal{G}_{l_\alpha m_\alpha, l_\beta m_\beta}^{lm} Q_{lm}^a (v_{\text{comp}}^a)_{\alpha\beta}^l \\
 & + \sum_{lm} \mathcal{G}_{l_\beta m_\beta, l_\alpha m_\alpha}^{lm} (v_q^a)^{lm} ((n^a)_{\alpha\beta}^l - (\tilde{n}^a)_{\alpha\beta}^l) \\
 & + \sum_{lm} \sum_{l_\mu m_\mu} \sum_{l_\nu m_\nu} \mathcal{G}_{l_\alpha m_\alpha, l_\beta m_\beta}^{lm} (v_H^a)_{\alpha\beta\mu\nu}^l \mathcal{G}_{l_\nu m_\nu, l_\mu m_\mu}^{lm} \left(\sum_{n'} f_{n'} \langle \tilde{\Psi}_{n'} | \tilde{p}_\mu^a | \tilde{p}_\nu^a | \tilde{\Psi}_{n'} \rangle \right) \\
 & + \sum_{lm} \mathcal{G}_{l_\alpha m_\alpha, l_\beta m_\beta}^{lm} (v_{\text{xc}}^a)_{\alpha\beta}^{lm},
 \end{aligned} \tag{A.1}$$

where

$$\mathcal{G}_{l_i, m_i; l_j, m_j}^{l, m} = \int Y_{lm}^*(\hat{\mathbf{r}}) Y_{l_i, m_i}(\hat{\mathbf{r}}) Y_{l_j, m_j}^*(\hat{\mathbf{r}}) d\hat{\mathbf{r}} \tag{A.2}$$

is the Gaunt coefficient. Most of the matrix elements in (A.1) can be precomputed:

$$\begin{aligned}
 (n^a)_{\alpha\beta}^l &= \int_0^{r_c^a} \phi_{n_\alpha l_\alpha}^a(r) \phi_{n_\beta l_\beta}^a(r) r^l dr, \\
 (\tilde{n}^a)_{\alpha\beta}^l &= \int_0^{r_c^a} \tilde{\phi}_{n_\alpha l_\alpha}^a(r) \tilde{\phi}_{n_\beta l_\beta}^a(r) r^l dr,
 \end{aligned}$$

$$\begin{aligned}
(t^a)_{\alpha\beta} &= \frac{\delta_{m_\alpha m_\beta} \delta_{l_\alpha l_\beta}}{2} \int_0^{r_c^a} \left\{ (\phi_{n_\alpha l_\alpha}^a)' (\phi_{n_\beta l_\beta}^a)' - (\tilde{\phi}_{n_\alpha l_\alpha}^a)' (\tilde{\phi}_{n_\beta l_\beta}^a)' \right. \\
&\quad \left. + l_\alpha (l_\alpha + 1) \frac{\phi_{n_\alpha l_\alpha}^a \phi_{n_\beta l_\beta}^a - \tilde{\phi}_{n_\alpha l_\alpha}^a \tilde{\phi}_{n_\beta l_\beta}^a}{r^2} \right\} dr, \\
(v_Z^a)_{\alpha\beta} &= -|Z^a| \delta_{m_\alpha m_\beta} \delta_{l_\alpha l_\beta} \int_0^{r_c^a} \frac{\phi_{n_\alpha l_\alpha}^a \phi_{n_\beta l_\beta}^a}{r} dr, \\
(v_{\text{core}}^a)_{\alpha\beta} &= \delta_{m_\alpha m_\beta} \delta_{l_\alpha l_\beta} 4\pi \int_0^{r_c^a} \int_0^{r_c^a} \frac{1}{r_{>}} \phi_{n_\alpha l_\alpha}^a(r) \phi_{n_\beta l_\beta}^a(r) n_{\text{core}}(r') r'^2 dr dr', \\
(\bar{v}^a)_{\alpha\beta} &= -\delta_{m_\alpha m_\beta} \delta_{l_\alpha l_\beta} \int_0^{r_c^a} \bar{v}^a(r) \tilde{\phi}_{n_\alpha l_\alpha}^a \tilde{\phi}_{n_\beta l_\beta}^a dr, \\
(v_{\text{comp}}^a)_{\alpha\beta}^l &= -\frac{4\pi}{2l+1} \int_0^{r_c^a} \int_0^{r_c^a} \tilde{\phi}_{n_\alpha l_\alpha}^a(r) \tilde{\phi}_{n_\beta l_\beta}^a(r) \frac{r_{\leq}^l}{r_{>}^{l+1}} g_l^a(r') r'^2 dr' dr, \\
(v_H^a)_{\alpha\beta\mu\nu}^l &= \frac{4\pi}{2l+1} \int_0^{r_c^a} \int_0^{r_c^a} \frac{r_{\leq}^l}{r_{>}^{l+1}} \{ \phi_{n_\alpha l_\alpha}^a(r) \phi_{n_\beta l_\beta}^a(r) \phi_{n_\mu l_\mu}^a(r') \phi_{n_\nu l_\nu}^a(r') \\
&\quad - \tilde{\phi}_{n_\alpha l_\alpha}^a(r) \tilde{\phi}_{n_\beta l_\beta}^a(r) \tilde{\phi}_{n_\mu l_\mu}^a(r') \tilde{\phi}_{n_\nu l_\nu}^a(r') \} dr dr'
\end{aligned}$$

However, the multipole potential coefficients and the exchange correlation matrix elements must be recalculated at each step in the self-consistent process.

$$\begin{aligned}
(v_q^a)^{lm} &= \iint \frac{g_l^a(r) Y_{lm}(\hat{\mathbf{r}}) (\tilde{n}(\mathbf{r}') + n_{\text{cmp}}(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\
&\quad - \iint_{\Omega_a \Omega_a} \frac{g_l^a(r) Y_{lm}(\hat{\mathbf{r}}) (\tilde{n}^a(\mathbf{r}') + n_{\text{cmp}}^a(\mathbf{r}'))}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}',
\end{aligned} \tag{A.3}$$

$$(v_{\text{xc}}^a)_{\alpha\beta}^{lm} = \int_0^{r_c^a} [\phi_{n_\alpha l_\alpha}^a(r) v_{\text{xc}}^{lm}(r) \phi_{n_\beta l_\beta}^a(r) - \tilde{\phi}_{n_\alpha l_\alpha}^a(r) \tilde{v}_{\text{xc}}^{lm}(r) \tilde{\phi}_{n_\beta l_\beta}^a(r)] dr, \tag{A.4}$$

$$\begin{aligned}
v_{\text{xc}}^{lm}(r) &= \int v_{\text{xc}}[n^a + n_c^a] Y_{lm}^*(\hat{\mathbf{r}}) d\hat{\mathbf{r}}, \\
\tilde{v}_{\text{xc}}^{lm}(r) &= \int v_{\text{xc}}[\tilde{n}^a] Y_{lm}^*(\hat{\mathbf{r}}) d\hat{\mathbf{r}}.
\end{aligned}$$

References

- [1] D.R. Salahub, in: K.P. Lawley (Ed.), *Ab Initio Methods in Quantum Chemistry-II*, John Wiley & Sons, New York, 1987, p. 447.

- [2] D.J. Singh, *Planewaves, Pseudopotentials and the LAPW Method*, Kluwer Academic Publishers, 1994.
- [3] M.C. Payne et al., *Rev. Mod. Phys.* 64 (1992) 1045.
- [4] J. Zhu, X.W. Wang, S.G. Louie, *Phys. Rev. B* 45 (1992) 8887.
- [5] R. Car, M. Parrinello, *Phys. Rev. Lett.* 55 (1985) 247.
- [6] D. Marx, J. Hutter, in: J. Grotendorst (Ed.), *Modern Methods and Algorithms of Quantum Chemistry*, NIC Series, Vol. 1, Forschungszentrum Julich, 2000.
- [7] M. Sung, R. Kawai, J.H. Weare, *Phys. Rev. Lett.* 73 (1994) 3552.
- [8] S.G. Louie, S. Froyen, M.L. Cohen, *Phys. Rev. B* 26 (1982) 1738.
- [9] D. Porezag, M.R. Pederson, A.Y. Liu, *Phys. Stat. Sol.* 217 (2000) 219.
- [10] P.J.D. Lindan, N.M. Harrison, M.J. Gillan, J.A. White, *Phys. Rev. B* 55 (1997) 15 919.
- [11] P.J.D. Lindan, N.M. Harrison, M.J. Gillan, *Phys. Rev. Lett.* 80 (1998) 762.
- [12] S.P. Lewis, A.M. Rappe, *Phys. Rev. Lett.* 77 (1996) 5241.
- [13] A.M. Rappe, K.M. Rabe, E. Kaxiras, J.D. Joannopoulos, *Phys. Rev. B* 41 (1990) 1227.
- [14] J.S. Lin, A. Qteish, M.C. Payne, V. Heine, *Phys. Rev. B* 47 (1993) 4174.
- [15] D. Vanderbilt, *Phys. Rev. B* 41 (1990) 7892.
- [16] K. Laasonen et al., *Phys. Rev. B* 47 (1993) 10 142.
- [17] C. Massobrio, A. Pasquarello, R. Car, *Phys. Rev. Lett.* 75 (1995) 2104.
- [18] C. Massobrio, A. Pasquarello, R. Car, *Phys. Rev. B* 54 (1996) 8913.
- [19] C. Massobrio, A. Pasquarello, R. Car, *Chem. Phys. Lett.* 238 (1995) 215.
- [20] V. Musolino, A. Selloni, R. Car, *Phys. Rev. Lett.* 83 (1999) 3242.
- [21] A. Palma, A. Pasquarello, G. Ciccotti, R. Car, *J. Chem. Phys.* 108 (1998) 9933.
- [22] A. Pasquarello et al., *Phys. Rev. Lett.* 69 (1992) 1982.
- [23] P.E. Blöchl, *Phys. Rev. B* 50 (1994) 17 953.
- [24] M. Valiev, J.H. Weare, *J. Phys. Chem. A* 103 (1999) 10 588.
- [25] N.A.W. Holzwarth et al., *Phys. Rev. B* 55 (1997) 2005.
- [26] N.A.W. Holzwarth, G.E. Matthews, A.R. Tackett, R.B. Dunning, *Phys. Rev. B* 57 (1997) 11 827.
- [27] G. Kresse, D. Joubert, *Phys. Rev. B* 59 (1999) 1758.
- [28] X. Gonze, P. Käckell, M. Scheffler, *Phys. Rev. B* 41 (1990) 12 264.
- [29] E.J. Bylaska, P.R. Taylor, R. Kawai, J.H. Weare, *J. Phys. Chem.* 100 (1996) 6966.
- [30] M.I. Lubin, E.J. Bylaska, J.H. Weare, *Chem. Phys. Lett.* 322 (2000) 447.
- [31] R.W. Hockney, *Methods Comput. Phys.* 9 (1970) 135.
- [32] R.W. Hockney, J.W. Eastwood, *Computer Simulations Using Particles*, McGraw-Hill, New York, 1981.
- [33] R.N. Barnett, U. Landman, *Phys. Rev. B* 48 (1993) 2081.
- [34] R.A. James, *J. Comput. Phys.* 35 (1977) 71.
- [35] R. Kawai, Private communication, 1996.
- [36] M.R. Jarvis, I.D. White, R.W. Godby, M.C. Payne, *Phys. Rev. B* 56 (1997) 14 972.
- [37] G.J. Martna, M.E. Tuckerman, *J. Chem. Phys.* 110 (1999) 2810.
- [38] L.J. Clarke, I. Stich, M.C. Payne, *Comput. Phys. Comm.* 72 (1992) 14.
- [39] J.S. Nelson, S.J. Plimpton, M.P. Sears, *Phys. Rev. B* 47 (1993) 1765.
- [40] J. Wiggs, H. Jonsson, *Comput. Phys. Comm.* 81 (1994) 1.
- [41] P.N. Swartztrauber, *FFTPACK*, Version 4: A package of Fortran subprograms for the Fast Fourier Transform of periodic and other symmetric sequences, National Center for Atmospheric Research, Boulder, CO, 1985.
- [42] PPGI IA-32, *Compilers and tools release 3.1*, The Portland Group, Inc., 1999.
- [43] *MPI/Pro Software*, Version 1.5 for the Linux Operating System, MPI Software Technology, Inc., 1999.
- [44] P.D. Haynes, M. Cote, *Comput. Phys. Comm.* 130 (2000) 130.
- [45] R.J. Harrison, J.A. Nichols, T.P. Straatsma, M. Dupuis, E.J. Bylaska, G.I. Fann, T.L. Windus, E. Apra, J. Anchell, D. Bernholdt, P. Borowski, T. Clark, D. Clerc, H. Dachsel, B. de Jong, M. Deegan, K. Dyall, D. Elwood, H. Fruchtl, E. Glendenning, M. Gutowski, A. Hess, J. Jaffe, B. Johnson, J. Ju, R. Kendall, R. Kobayashi, R. Kutteh, Z. Lin, R. Littlefield, X. Long, B. Meng, J. Nieplocha, S. Niu, M. Rosing, G. Sandrone, M. Stave, H. Taylor, G. Thomas, J. van Lenthe, K. Wolinski, A. Wong, Z. Zhang, *NWChem*, A Computational Chemistry Package for Parallel Computers, Version 4.0, Pacific Northwest National Laboratory, Richland, WA, 2000.
- [46] S.H. Vosko, L. Wilk, M. Nusair, *Can. J. Phys.* 58 (1980) 1200.
- [47] N.D. Godbout, D.R. Salahub, J. Andzelm, E. Wimmer, *Can. J. Chem.* 70 (1992) 560.
- [48] A.D. Becke, *J. Chem. Phys.* 88 (1988) 2547.
- [49] C.W. Murray, N.C. Handy, G.L. Laming, *Mol. Phys.* 78 (1993) 997.
- [50] D.W. Marquardt, *SIAM J.* 11 (1963) 431.
- [51] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B. P. Flannery, *Numerical Recipes in Fortran. The Art of Scientific Computing*, Cambridge University Press, 1992.