Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 7113–7124
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

State-of-the-art eigensolvers for electronic structure calculations of large scale nano-systems ☆

Christof Vömel [a,*], Stanimire Z. Tomov [b], Osni A. Marques [a], A. Canning [a], Lin-Wang Wang [a], Jack J. Dongarra [b]

[a] Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, United States
[b] Computer Science Department, University of Tennessee, Knoxville, TN 37996-3450, United States

Received 25 May 2007; received in revised form 8 January 2008; accepted 10 January 2008
Available online 26 January 2008

Abstract

The band edge states determine optical and electronic properties of semiconductor nano-structures which can be computed from an interior eigenproblem. We study the reliability and performance of state-of-the-art iterative eigensolvers on large quantum dots and wires, focusing on variants of preconditioned CG, Lanczos, and Davidson methods. One Davidson variant, the GD + k (Olsen) method, is identified to be as reliable as the commonly used preconditioned CG while consistently being between two and three times faster.
Published by Elsevier Inc.

Keywords: Computational nano-technology; Electronic structure; Preconditioned conjugate gradients; Implicitly restarted Arnoldi; Davidson's method

1. Introduction

The computation of optical and electronic properties of large nano-structures such as quantum dots and wires is an important field of current research. As these particular electronic structure calculations can be formulated as eigenvalue problems, it is pressing to study what can be gained from recent progress on iterative eigensolvers in the numerical linear algebra community. This current paper addresses this need in comprehensive fashion by investigation of problems of realistic size and state-of-the-art algorithms. It supersedes previous work [3,42].

The connection of this application to eigenvalue problems stems from the Kohn–Sham approximations [13], through which one obtains an effective single-particle Schrödinger equation

☆ This work was supported by the US Department of Energy under LAB03-17 initiative, Contract Nos. DE-FG02-03ER25584 and DE-AC03-76SF00098.
* Corresponding author. Tel.: +41 44 632 6141.
E-mail addresses: cvoemel@lbl.gov (C. Vömel), tomov@cs.utk.edu (S.Z. Tomov), OAMarques@lbl.gov (O.A. Marques), OAMarques@lbl.gov (A. Canning), LWWang@lbl.gov (L.-W. Wang), dongarra@cs.utk.edu (J.J. Dongarra).

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2008.01.018

$$Hw_i := \left[-\frac{1}{2}\Delta + V_{\text{pot}}\right] w_i = \lambda_i w_i. \tag{1}$$

The Hamiltonian H is Hermitian indefinite and consists of the kinetic energy term, represented by the Laplacian Δ, and the potential V_{pot}. The set $\{w_i\}$ denotes the complex orthogonal wave-functions (eigenstates) and the real $\{\lambda_i\}$ their corresponding energies (eigenvalues).

In the context of the self-consistent field iteration for the ground state [26,27], a potentially large number of eigenstates of (1) needs to be computed, see for example [31] and also the recent review [30]. On the other hand, when the potential is given, on a real space grid, one often can restrict the computation to only a *small* number of interior eigenstates of (1) from which optical and electronic properties can be determined. The latter setting, the computation of band edge states, is the focus of this work. Section 2 outlines the computational challenges of this interior eigenvalue problem.

The parallel Energy SCAN (ESCAN) method [4], which solves (1) in a plane wave basis, gives the framework in which eigensolvers are compared. Section 3 gives a short overview of the algorithms and software packages studied here:

- our own implementation of the preconditioned conjugate gradient (PCG) method [45],
- our own implementation of the locally optimal block preconditioned conjugate gradient (LOBPCG), derived from [10],
- implicitly restarted Arnoldi/Lanczos (IRL), from P_ARPACK [14,15], and
- Generalized Davidson (GD + k) and Jacobi-Davidson (JD) implementations from PRIMME [17,36,37].

PCG is the current standard method in ESCAN and LOBPCG represents an 'all-band' generalization. Both methods work with subspaces of constant dimension. In contrast, IRL, GD + k and JD represent methods working with increasing subspaces and use restarts. Section 4 describes the test setup for our evaluation. Section 5 analyzes the usefulness of the algorithms, focusing on robustness and performance. Section 6 discusses our conclusions and possibilities for future work.

2. Computational challenges

2.1. Matrix-free computation

Plane wave computations use a special evaluation of the Hamiltonian H from (1) as it can be dense in both real and Fourier space. This reduces the cost of the matrix–vector product to $\mathcal{O}(N \log N)$ instead of $\mathcal{O}(N^2)$, N being the dimension of H.

The discrete Laplacian is directly computed in the plane wave basis because it is diagonal. The discretized potential is typically available only in real space. After a fast Fourier transform of a given plane wave vector into real space, the potential is evaluated; subsequently the result is re-transformed into the plane wave basis. Thus, since it is too large to store, H is never computed explicitly, instead it becomes only implicitly available through matrix–vector products. Hence, one needs to use iterative eigenvalue methods, and the choice of preconditioners is limited.

2.2. Degeneracies, gaps, and interior eigenpairs

There are three principal difficulties that make our eigenproblem challenging. First, physical symmetries in the system at hand result in eigenvalues of higher multiplicity, say three. Because of its physical relevance, the multiplicity of a degenerate eigenvalue needs to be reliably computed together with the associated eigenspace. Second, with an increasing number of electrons in the system, the relative separation of eigenvalues of interest from their neighbors decreases. Last, we need to solve an interior eigenproblem. We will consider states at the valence band maximum (VBM), the highest occupied state of the system, and states at the conduction band minimum (CBM), the lowest unoccupied state. These states at the 'band gap' can be separated by several thousand eigenvalues from the left end of the spectrum.

3. Iterative Hermitian eigensolvers

Based on a short description of the underlying concepts in Section 3.1, we give in Section 3.2 an overview of the methods studied in this paper. Our emphasis is on highlighting differences and similarities between the various approaches. For alternative reviews which also influenced this current work, we refer to [1,36,37,47] and also the monographs [2,7,24,28,40].

3.1. Fundamental concepts

For a given wave function w, define its Rayleigh quotient ρ (the energy functional) and the residual r as

$$\rho(w) := \frac{w^* H w}{w^* w}, \quad r(w) := Hw - \rho(w)w. \tag{2}$$

It can be shown that there always is an eigenvalue of H within distance $\|r(w)\|_2$ to $\rho(w)$, see [24]. Furthermore, as the residual points in the same direction as $\nabla\rho$, the eigenvectors are exactly the stationary points of ρ. Last, ρ is bounded for $w \neq 0$

$$\lambda_{\min}(H) = \min_{w\neq 0} \rho(w), \quad \lambda_{\max}(H) = \max_{w\neq 0} \rho(w), \tag{3}$$

where $\lambda_{\min}(H)$ and $\lambda_{\max}(H)$ denote the smallest and largest eigenvalue of H, respectively.

For $V \in \mathcal{C}^{n\times k}$, $V^*V = I$, define the matrix Rayleigh Quotient $\varrho(V) := V^*HV$. For each eigenpair (μ_i, y_i) of ϱ, the Rayleigh–Ritz (also called subspace diagonalization) procedure considers the Ritz value μ_i and Ritz vector $w_i := Vy_i$ as an approximation to an eigenpair of H. As the Ritz values only depend on $\mathcal{V}$, the space spanned by the columns of V, application of (3) to ϱ yields

$$\mu_{\min}(\mathcal{V}) := \lambda_{\min}(\varrho) = \min_{\mathcal{V}\ni w\neq 0} \rho(w), \quad \mu_{\max}(\mathcal{V}) := \lambda_{\max}(\varrho) = \max_{\mathcal{V}\ni w\neq 0} \rho(w). \tag{4}$$

One way to compute an interior eigenpair closest to a prescribed reference energy e_{ref} is to apply the Rayleigh Quotient $\rho(w) = \rho(H, w)$ to a spectral transformation of H. The 'shift-and-invert' Rayleigh quotient $\rho((H - e_{\text{ref}}I)^{-1}, w)$ is not perfectly suitable for our purposes: as the Hamiltonian H is only implicitly available through matrix–vector products, an additional interior linear solver would be needed. Instead, we consider the folded spectrum [45] Rayleigh quotient $\rho((H - e_{\text{ref}}I)^2, w)$ and Rayleigh–Ritz matrix $\varrho((H - e_{\text{ref}}I)^2, V)$. In this case, the goal becomes the computation of the smallest positive eigenvalue(s) and associated eigenvector(s) of the transformed operator.[1] In practice, we choose two reference energies: e_{v} closer to the VBM, and e_{c} closer to the CBM.

3.2. Iterative eigensolvers

Each of the methods in Table 1 constructs a sequence of matrices $\{V_i\}$, usually with orthonormalized columns, from which subspace diagonalization computes eigenpair approximations. The distinguishing feature of each method is the way in which V_{i+1} is constructed with respect to V_i. First, a new orthonormalized direction can be included to augment the subspace. Second, to reduce the subspace dimension, one or more directions can be removed. Last, if multiple eigenpairs are desired and an eigenpair converges before the others, the converged pair is 'locked', i.e. it is no longer used as an iterate.

3.2.1. PCG and LOBPCG

The methods in this section work with Rayleigh–Ritz projections on subspaces of constant size. Our preconditioned conjugate gradient (PCG) method [45] calculates the smallest eigenpair(s) of the folded spectrum operator. Given the iterate w_i^{PCG} and a descent direction d_i^{PCG}, one computes

$$w_{i+1}^{\text{PCG}} = \arg\min\{\rho((H - e_{\text{ref}}I)^2, w) : w \in \text{span}(w_i^{\text{PCG}}, d_i^{\text{PCG}})\}, \tag{5}$$

[1] For alternative spectral transformations for the interior eigenproblem, see for example [23,33,41].

Table 1
Overview of methods studied in this paper

Algorithm	Operator	Subspace	Implementation	References	Description
PCG	Folded	Constant	Self	[45]	3.2.1
LOBPCG	Folded	Constant	Self	[10]	3.2.1
IRL	Un/folded	Increasing	P_ARPACK	[14,15]	3.2.2
GD + k (Olsen)	Un/folded	Increasing	PRIMME	[17,36,37]	3.2.3
JDQMR	Un/folded	Increasing	PRIMME	[17,36,37]	3.2.3

Algorithms include preconditioned conjugate gradients (PCG), locally optimal block preconditioned conjugate gradients (LOBPCG), implicitly restarted Arnoldi/Lanczos (IRL), Generalized Davidson (GD) with restart (the parameter k is explained in Section 3.2.3), and Jacobi-Davidson with QMR as inner solver (JDQMR). Some algorithms require a spectral transformation with the folded operator, others can use both the unfolded and folded operator: ARPACK's IRL allows the computation of eigenstates of smallest magnitude [14], and PRIMME's GD + k (Olsen) and JDQMR both can select states closest to one side of the reference shift. Following [1], the algorithms can be classified as using a subspace of constant or increasing size. Some algorithms are implemented by us, others are obtained from packages. The last column gives the section in which a description is given.

where the descent direction is computed using the preconditioned residual as

$$\hat{d}_i^{\mathrm{PCG}} = -P\, r(w_i^{\mathrm{PCG}}) + \beta_i d_{i-1}^{\mathrm{PCG}} \tag{6}$$

and the parameter β_i is chosen according to Nocedal and Wright [20].

There are various ways to extend PCG to a subspace method, see for example the references in [1,9]. In our 'band-by-band' inner–outer iteration, an inner loop performs a number of steps of the inner minimization (5) on each eigenstate, and the outer loop consists of an orthonormalization of the computed states with subsequent Rayleigh–Ritz.

In variation of (5), the locally optimal preconditioned conjugate gradient (LOPCG) method [10] works with a three-dimensional space, that is

$$w_{i+1}^{\mathrm{LOP}} = \arg\min \left\{ \rho((H - e_{\mathrm{ref}} I)^2, w) : w \in \mathrm{span}(w_{i-1}^{\mathrm{LOP}}, w_i^{\mathrm{LOP}}, P\, r(w_i^{\mathrm{LOP}})) \right\}. \tag{7}$$

In the block version, LOBPCG [10], (7) is generalized to use the Rayleigh–Ritz approximations from a set of iterates W_{i-1}, W_i, R_i. Its practical advantage compared to other methods is the absence of parameters to be set or tuned.

3.2.2. Implicitly restarted Arnoldi/Lanczos

This and the subsequent sections describe restarted projection methods using subspaces of variable size. The Lanczos algorithm [24] works with the Krylov space sequence $\mathcal{K}_i(H, v) := \mathrm{span}(v, Hv, \ldots, H^{i-1}v)$, generated from a suitable start vector v. It can also use the folded operator. Its characteristic property is the computation of an orthogonal basis V_i of $\mathcal{K}_i$ by a three-term recurrence. Within this basis, the matrix Rayleigh quotient ϱ_i is a real symmetric tridiagonal whose Ritz pairs *simultaneously* approximate different eigenpairs of the original matrix [24].

The implicit restart (IR) procedure [35] in ARPACK [14] and P_ARPACK [15] deflates unwanted Ritz pairs through the implicitly shifted QR algorithm [24]. The relevance of the QR algorithm as an ideal deflation technique in the tridiagonal case lies in the fact that it transforms the basis $V_m, m = k + p$, containing k wanted and p unwanted Ritz approximations into an orthogonal $V_m^+ := (V_k^+, V_p^+)$ such that (1) the columns of V_k^+, a basis of $\mathcal{K}_k(H, v_1^+)$, satisfy the three-term Lanczos recurrence, that (2) the matrix $\varrho_k^+ := (V_k^+)^* H V_k^+$ is again tridiagonal, and that (3) the eigenvalues of ϱ_k^+ are exactly the k Ritz values whose corresponding vectors one desires to keep in the subspace at restart.[2]

[2] ARPACK actually does not explicitly take advantage of the Hermitian structure of the Hamiltonian. Strictly speaking, the implementation uses a generalization of Lanczos, the Arnoldi algorithm. Nevertheless, in exact arithmetic, the procedure is equivalent to Implicitly Restarted Lanczos (IRL) in our Hermitian context.

3.2.3. Generalized Davidson and Jacobi-Davidson

This section focuses on two variants of Davidson's eigenvalue method [6] that are available, as default methods, in the PRIMME package [17,36,37].

The Generalized Davidson (GD) method [5,18,19] successively augments the projection subspace by directions of the form

$$(\widetilde{H} - \mu_i I)w_{i+1}^{\mathrm{GD}} = -r(w_i) \tag{8}$$

with a preconditioner $\widetilde{H}$ approximating the original H and $(\widetilde{H} - \mu_i I)$ being nonsingular. The method can restart with several vectors without any difficulty. In the GD + k method [19,36,38], one not only includes Ritz vectors from the current subspace like in a thick-restart, but also k additional ones from the previous iteration. This is somewhat similar to (7) of LOPCG. As a variation of Olsen's method [22] due to [39], the GD + k (Olsen) approach modifies the right-hand side of (8) by adding a multiple of w_i, see [17].

The Jacobi-Davidson approach [8,32,33] considers (8) in the subspace $w_i^{\perp}$. For $w_i, w_i^* w_i = 1$, define the projector $P_i := (I - w_i w_i^*)$ and solve the (generally indefinite) linear system

$$[P_i(\widetilde{H} - \mu_i I)P_i]w_{i+1}^{\mathrm{JD}} = -r(w_i), w_{i+1}^{\mathrm{JD}} \perp w_i \tag{9}$$

by QMR as an inner solver (see [29] for an overview of QMR and references). Expansion of (9) shows that this is an implicit formulation of the Olsen approach [33].

4. Test configuration

This section gives the background information for the test results presented in Section 5. Our experiments were performed on an IBM Power 5 with 8 processors per compute node, each with a peak performance of 7.6 GFlops. Table 2 gives a description of the nano-structures considered for this comparison. These systems have been selected as representatives for nano-structures of different sizes and levels of model accuracy.

4.1. Memory and important parameter settings

Table 3 gives an overview of the memory requirements. Our test cases were chosen such that memory was not a constraining factor in the comparison. IRL, GD + k (Olsen), and JDQMR could be run with *reasonably* large subspaces. However, to make the comparison fair, it was important to not let the subspace size greatly exceed the fixed amount of memory used in PCG and LOBPCG, and the restarts where chosen accordingly. Table 3 also gives a rough indication of what memory costs one might expect to incur in other applications.

For GD + k (Olsen), we used $k = 1$ similar to [36], that is one extra vector for maintaining the CG-like recurrence. This results in a minimum restart size $m + 2$ in Table 3. (We note that in exact arithmetic, LOBPCG with a given block size b is theoretically equivalent to a Generalized Davidson method with restart size b, with $k = b$ extra vectors retained from the previous step, and with maximum space size $3b$ [37].)

Table 2
Overview of the physical systems considered in this comparison: QD1 and QD2 are quantum dots, QW is a quantum wire

System	V_{pot}	Atoms	Cutoff (Ryd.)	N (K)	$e_{\mathrm{v}}, e_{\mathrm{c}}$(eV)
QD1	EPM (nl)	534 Cd, 527Se	6.88	141	−4.8, −3.8
QW	EPM (l)	34,624 In, 11,115 As, 20,885 P	5.0	2266	−5.4, −5.1
QD2	CP (nl)	675 Cd, 652 Se	35	2717	−0.4, 0.6

The atomic potential V_{pot} is either computed using the empirical pseudopotential method (EPM) [46] or charge patching (CP) [44]. Both quantum dot calculations use a nonlocal (nl) potential. For the quantum wire, only a local (l) potential is used. The quantum dots consist of cadmium (Cd) and selenium (Se) atoms and are embedded inside a larger box of vacuum. The quantum wire consists of indium arsenide (InAs) that is embedded in bulk indium phosphide (InP). The energy cutoff denotes the limit of Fourier space components considered, leading to a discrete Hamiltonian of the stated dimension N. Note that the charge patching potential requires a significantly larger energy cutoff, despite the relatively small number of atoms. The last column of the table specifies the reference energies, in eV, used for the folded spectrum approach, e_{v} denotes the shift for the valence band and e_{c} the one for the conduction band.

Table 3
Number of vectors (wave-functions) of dimension N for approximation of m eigenpairs in each algorithm

Algorithm	Minimum required	Maximum used
PCG	$m+1$	$m+1$
LOBPCG	$3m$	$3m$
IRL	$m+1$	$2m$–$5m$
GD + k (Olsen)	$m+2$	$\approx 2m$–$3m$
JDQMR	$m+1$	$\approx 2m$–$3m$

The minimum and maximum number of vectors used refer to the typical subspace size immediately after and before a restart. Note that PCG and LOBPCG reuse the same fixed memory throughout the computation. IRL, GD + k (Olsen), and JDQMR restart with the minimum number of vectors once the maximum is reached. Note that for methods with variable subspace size, we assume that at least five eigenpairs are to be computed, guaranteeing a minimum subspace size of at least 10.

PCG was used with 200 inner iterations for each state, then followed by outer subspace diagonalizations. The (block) iterations for LOBPCG have been limited to maximally 20,000. JDQMR from PRIMME implements an automatic accuracy control of the inner linear solver [17,21,37] so that the user need not intervene here. The maximum number of matrix vector (MV) products for PRIMME was set to 100,000 for QD1 and QW, and to 300,000 for QD2.

4.2. Choice of the preconditioner

All methods from Section 3.2, with the notable exception of Lanczos, rely on preconditioning, albeit in slightly different ways.

As the Hamiltonian H is not explicitly available in our setting, we cannot use preconditioners from matrix factorizations such as incomplete LU (ILU) [34]. We do not use multigrid because it is less suitable for plane wave calculations than for mesh eigenvalue problems [12,1]. Instead, we use a standard diagonal preconditioner from the approximation of the diagonal of the Hamiltonian.

For the folded spectrum, we use a diagonal preconditioner constructed from the (folded) sum of the Laplacian and an averaging potential [45]. With the original, unfolded Hamiltonian H, we employ a diagonal rational preconditioner that approximates the dominant diagonal of H, as suggested in [25]. All methods use the same preconditioner, folded and unfolded respectively, except for P_ARPACK which cannot use a preconditioner.

4.3. Convergence criteria, residual control

An eigenpair (ρ, w) of H is declared converged when $\|r(H,w)\|_2 \leqslant \tau$, for a given tolerance τ.

There is a potential issue regarding the stopping criterion when using the folded spectrum approach: in practice, one cares about the residual norm being small with respect to the original Hamiltonian H, and not with respect to the folded operator $(H - e_{\mathrm{ref}} I)^2$ that was merely used to facilitate the computation.

The problem is easily addressed as long as one can adapt the folded spectrum eigensolver to use the 'unfolded' stopping criterion. However, in 'black box' software that only provides a reverse-communication handle to the operator used in the computation, more work is needed for the folded case.

A first possibility is to embed the folded spectrum computation into an outer loop in which the folded spectrum accuracy is gradually increased until the required accuracy with respect to H is reached.

An alternative is the control of the unfolded residual through a stopping criterion based on the folded operator.

For an eigenvalue approximation $\hat{\lambda}$, define the folded residual

$$r_f := r((H - e_{\mathrm{ref}} I)^2, w) = (H - e_{\mathrm{ref}} I)^2 w - (\hat{\lambda} - e_{\mathrm{ref}})^2 w \tag{10}$$

and the corresponding unfolded one

$$r_u := r(H, w) = (H - e_{\mathrm{ref}} I) w - (\hat{\lambda} - e_{\mathrm{ref}}) w. \tag{11}$$

It follows that $r_f = (H + (\hat{\lambda} - 2e_{\text{ref}})I)r_u$ and thus

$$\|r_u\|_2 \leqslant \|(H + (\hat{\lambda} - 2e_{\text{ref}})I)^{-1}\|_2 \|r_f\|_2. \tag{12}$$

Consider the folded spectrum computation of the CBM λ_c using the reference shift $e_{\text{ref}} = e_c$, and further assume that the unfolded residual norm should be reduced to the accuracy level τ. In this case, for $e_c < \hat{\lambda}$,[3]

$$\|(H + (\hat{\lambda} - 2e_c)I)^{-1}\| = \frac{1}{\lambda_c - e_c + \hat{\lambda} - e_c} \leqslant \frac{1}{\lambda_c - e_c}, \tag{13}$$

and hence $\|r_f\|_2 \leqslant \tau * (\lambda_c - e_c)$ guarantees that $\|r_u\|_2 \leqslant \tau$ by (12).

For the folded spectrum computation of the VBM λ_c with reference shift e_v, one obtains the analogous stopping criterion $\|r_f\|_2 \leqslant \tau * (e_v - \lambda_v)$ provided that $\hat{\lambda} < e_v$.

5. Evaluation results

Table 4 shows the computed eigenvalues for valence and conduction bands. The valence band eigenpairs are in general more difficult to compute due to stronger clustering. Since the quantum dot QD1 is relatively small in size, we could afford to compute more eigenpairs than for the other two systems. In general, the eigenvalue clustering is stronger in the valence band. For this reason, we expect these computations to be more difficult than the corresponding conduction band calculations.

5.1. Robustness

As the degeneracy of eigenpairs corresponds to physical symmetries in the system, it is important to correctly determine the multiplicity of eigenvalues. Whenever a Hermitian matrix can be factored, the number of eigenvalues in a given interval can be found by using Sylvester's theorem [24]. However, when the Hamiltonian is not explicitly available as in this current study, no iterative method is theoretically guaranteed to reliably determine multiplicities in all cases. However, PCG commonly does not miss states in practice.

In order to compute the correct eigenvalues, practitioners employ a number of heuristics including various deflation variants, locking, and blocking, see for example [11,16]. ARPACK and PRIMME provide their own deflation routines, our implementations of PCG and LOBPCG lock converged vectors and continue iterating with the unconverged ones.

The general benefits of blocking are well recognized but blocking by itself does not guarantee reliability, see the model problem in [1] for an example. Furthermore, even with an available blocked matrix–vector product, the actual performance might still depend on the particular problem, the implementation, and the computer [11].

Since not all algorithms in our comparison are available in blocked form (ARPACK does not provide a blocked version), results shown in this paper are computed without blocked matrix–vector product. This means that the timing results of block methods shown in Section 5.2 might be very different for another implementation. For this reason, we report the number of matrix vector (MV) products for each method which is independent of the block size. Furthermore, we note the finding of [37] that at least for Davidson-type methods, 'when seeking many eigenvalues, we [the authors] have never found the use of a block size $b > 1$ beneficial'.

We here focus on investigating the benefits of increasing the subspace dimension versus enforcing higher accuracy.

Table 5 summarizes our findings on the reliability of the algorithms. The GD + k (Olsen) method from PRIMME turned out to be reliable for both folded and unfolded computations. PCG and LOBPCG were both reliable for finding the states closest to the band gap in all cases. In one case, the QD2 CBM, both algorithms failed to make approximate eigenpairs five and six converge to the desired accuracy. ARPACK's IRL turned out to converge very slowly for most folded spectrum computations. In unfolded computations with

[3] We assume that the VBM of H is sufficiently far away from e_c.

Table 4
Computed eigenvalues in the valence and conduction bands of the test systems

Index	QD1		QW		QD2	
	Valence	Conduction	Valence	Conduction	Valence	Conduction
(e_{ref})	−4.8	−3.8	−5.4	−5.1	−0.4	0.6
1	−5.39076	−3.10118	−5.73241	−4.89017	−0.72398	1.35724
2	−5.39076	−2.83770	−5.73241	−4.71187	−0.72398	1.64617
3	−5.40313	−2.81099	−5.73423	−4.68034	−0.72398	1.64617
4	−5.44361	−2.81099	−5.74245	−4.68034	−0.72946	1.64617
5	−5.44361	−2.56043	−5.74360	−4.55008	−0.72946	1.92364
6	−5.48316	−2.52280			−0.72946	1.92364
7	−5.49335	−2.52280				
8	−5.51804	−2.51159				
9	−5.51804	−2.51159				
10	−5.52054	−2.37371				
Next	−5.52054	−2.28992	−5.75502	−4.55008	−0.77562	1.92364

Results have been computed with residual norm accuracy 10^{-6} with respect to the operator H. The energies are sorted by increasing distance from their respective folded spectrum reference shift, given in the leading row, with their index numbering correspondingly. This means that the valence band is ordered from largest to smallest, and the conduction band from smallest to largest. The computed band gap can be found as the difference of the two numbers in the first row. The trailing row shows the next eigenvalue (which is not computed in the tests), to give a feeling for the difficulty of each calculation.

Table 5
Reliability of each algorithm to compute the eigenvalues with their correct multiplicities as shown in Table 4 up to the specified tolerance 10^{-6}

Algorithm	QD1		QW		QD2	
	VBM	CBM	VBM	CBM	VBM	CBM
PCG (F)	√	√	√	√	√	‡*
LOBPCG (F)	√	√	√	√	‡	‡*
IRL (F)	‡	†	‡	‡	‡	‡
GD + k (F)	√	√	√	√	√	√
JDQMR (F)	√	√	√	√	√	‡*
IRL (U)	†	√	‡	√	†	†
GD + k (U)	√	√	√	√	‡	‡
JDQMR (U)	√	√	√	√	‡	‡

(F) stands for folded spectrum computation, (U) stands for unfolded, that is the natural Hamiltonian. √ denotes success, † denotes convergence but some missed eigenvalues, and ‡ denotes failure to converge in specified maximum number of iterations. Note that some of these failures can be corrected. The asterisk indicates that the convergence problems of PCG, LOBPCG, and JDQMR with the QD2 CBM can be avoided by including an additional, extremely close by state in the computation.

the original operator, IRL sometimes showed mis-convergence (that is, convergence to the wrong eigenvalues, resulting in relevant eigenpairs being missed).

A remedy for some convergence failures, for example feasible for PCG and LOBPCG in the QD2 CBM calculation, is to increase the number of eigenvalues to be computed. While incurring a slightly higher cost, the orthogonalization procedure can help reduce errors in the eigenpairs of interest stemming from irrelevant yet close-by neighboring eigenpairs. For ARPACK, we found that an increase of the maximum Krylov subspace dimension was sometimes helpful. As an example, the unfolded QW VBM calculation was successful when increasing the size to 50, that is ten times the number of wanted eigenpairs.

What is the right remedy for correcting the mis-convergence problem? In our experiments, increasing the subspace dimensions from Table 3 by a moderate number of additional vectors for methods with a variable subspace size was not successful. However, increasing the tolerance to force eigenpairs to converge further before deflating or locking them can be helpful.

5.2. Matrix–vector products and wall clock time

Table 6 states the run time results when computing ten eigenpairs of the quantum dot QD1. For both VBM and CBM, the folded GD + k (Olsen) is the fastest method. Compared to the default PCG method in ESCAN, it is five times as fast for the VBM and three times as fast for the CBM. The unfolded P_ARPACK comes close for the CBM but misses some eigenvalues for the VBM. GD + k (Olsen) is also by far the best with respect to matrix vector multiplications. The orthonormalization cost is highest for the large basis used in our implementation of LOBPCG. Even though it takes significantly fewer matrix vector multiplications than PCG, its runtime is longer. This could potentially be improved by a more efficient implementation but will not change the number of matrix vector multiplications which is about three times higher than for the fastest method, GD + k (Olsen).

The larger amount of matrix–vector (MV) products of the PCG and LOBPCG methods compared to GD + k (Olsen) indicates that, with the given preconditioner, the local subspace-iteration-like of the conjugate gradient algorithms is here less effective than the Generalized Davidson approach. This is not uncommon: in [36], similar findings on some examples from test matrix collections are reported.

Table 7 lists the run time results when computing five eigenpairs of the quantum wire. For the VBM, the folded GD + k (Olsen) is the fastest method, again being about five times faster than standard PCG. The unfolded P_ARPACK is fastest for the CBM. Folded GD + k (Olsen) is about 25% slower than unfolded P_ARPACK but still 2.5 times faster than PCG. For the VBM, neither folded nor unfolded P_ARPACK converges in the specified maximum number of iterations and basis size.

Table 8 lists the run time results when computing six eigenpairs of the quantum dot QD2. Computationally, the nonlocal matrix vector product is significantly more expensive than for the QW. PCG and LOBPCG show

Table 6
QD1, nonlocal EPM potential, 1061 atoms, $n = 141$ K. IBM SP5, $p = 32$, 10 eigenpairs

Algorithm (QD1)	VBM		CBM	
	MV products	Runtime (s)	MV products	Runtime (s)
PCG (F)	23,810	228	7066	63
LOBPCG (F)	16,862	255	6880	92
IRL (F)	‡	‡	†	†
GD + k (F)	4762	46	2176	21
JDQMR (F)	11,259	109	6212	52
IRL (U)	†	†	2434	24
GD + k (U)	15,449	173	12813	144
JDQMR (U)	11,259	99	6383	59

(F) stands for folded spectrum computation, (U) stands for unfolded, that is natural Hamiltonian. † denotes convergence but some missed eigenvalues, and ‡ denotes failure to converge in specified maximum number of iterations.

Table 7
QW, local EPM potential, 66,624 atoms, $n = 2.27$M. IBM SP5, $p = 64$, 5 eigenpairs

Algorithm (QW)	VBM		CBM	
	MV products	Runtime (s)	MV products	Runtime (s)
PCG (F)	149,726	7278	21,931	1072
LOBPCG (F)	56,207	3690	20,337	1377
IRL (F)	‡	‡	‡	‡
GD + k (F)	26,326	1424	8504	418
JDQMR (F)	81,364	3877	28076	1392
IRL (U)	‡	‡	8232	344
GD + k (U)	39,828	2452	20,669	1269
JDQMR (U)	56,441	2746	20,392	1036

(F) stands for folded spectrum computation, (U) stands for unfolded, that is natural Hamiltonian. ‡ denotes a failure to converge in specified maximum number of iterations.

Table 8
QD2, nonlocal CP potential, 1327 atoms, $n = 2.7$ M. IBM SP5, $p = 64$, 6 eigenpairs

Algorithm (QD2)	VBM		CBM	
	MV products	Runtime [s]	MV products	Runtime [s]
PCG (F)	101,904	15,671	‡	‡
LOBPCG (F)	‡	‡	‡	‡
IRL (F)	‡	‡	‡	‡
GD + k (F)	54,362	8758	62,334	10,211
JDQMR (F)	254,810	39,111	‡	‡

(F) stands for folded spectrum computation, (U) stands for unfolded, that is natural Hamiltonian. ‡ denotes a failure to converge in specified maximum number of iterations. To get a feeling for the performance of GD + k in the CBM calculation, we note that it was more than two times faster than the unconverged PCG reaching the maximum number of iterations. The unfolded calculations are not reported since none of them succeeded.

convergence problems in the CBM since, as a more difficult test, only CBM states number five and six, but not seven with nearly identical energy, are computed, see also the rightmost column of Table 4. GD + k (Olsen) is both fast and reliable for both VBM and CBM. Even though the time of PCG for the CBM is not shown in Table 6 because it did not converge, we mention that GD + k was more than two times faster than the unconverged PCG reaching the maximum number of iterations.

Overall, on the considered test systems, the CG-based methods are slower than the Davidson methods, and than Lanczos when it converges. Leaving aside the difference in the computation of the next iterate, one possible reason is that the CG methods simultaneously approximate all eigenpairs from one subspace of fixed size. In contrast, IRL, GD + k, and JDQMR do not decrease the size of the subspace when an eigenpair has converged. In that respect, these methods are more flexible in the generation and use of their subspace to approximate later-converging eigenpairs.

6. Summary, conclusions and future work

This paper described a study of modern iterative eigensolvers for the computation of interior eigenstates close to the band gap of large semiconductor nano-structures. Degeneracies and eigenvalue clustering make this problem a hard one to solve, and additional difficulties arise from the matrix-free plane wave formulation.

We performed a robustness and timing evaluation for valence and conduction band calculations. The GD + k (Olsen) method from the PRIMME package turned out to be very reliable and at the same time up to five times faster than the commonly used PCG method. PCG still has its place in calculations needing extraordinary amounts of memory. LOBPCG, with the preconditioner used here, and without blocked MV product, was not competitive. Furthermore, the higher subspace diagonalization cost, associated with a larger basis size, had a negative impact on the overall performance. The unfolded IRL from P_ARPACK was very fast for some calculations but in general unreliable when used with memory comparable to the other methods. Improved filtering methods might help to correct this problem [47].

We also discussed strategies to address convergence failures and missed eigenvalues. ARPACK's IRL method, in our opinion, would benefit from providing a block version. This might reduce the probability of missed eigenvalues. In addition, this would be useful for starting a calculation with pre-converged states, for example from a previous computation or with good initial guesses for several wave-functions at the same time. All other algorithms studied here can work with multiple start vectors.

We showed that spectral transformations may complicate the matter of converging to a required accuracy with respect to the original Hamiltonian H. For this reason, we suggest that algorithm developers allow for more flexible stopping criteria even though those might incur costs of additional MV products. In the case of the folded operator, it is enough to control the folded residual norm as we demonstrated. Nevertheless, for more general spectral transformations, it might still become useful to determine convergence based on the residual norm defined with respect to the original problem.

One aspect of future work is the exploration of alternative spectral transformations for the interior eigenproblem such as harmonic Ritz values [2,23]. A second possible direction of future research concerns better

preconditioners. We recently developed a bulk-based preconditioner to improve the convergence of PCG for band edge calculations [43]. It will be interesting to explore the combination of this new preconditioner with the best algorithm in this study, GD + k (Olsen). This new method promises to be a faster work horse for electronic nano-structure calculations of the type and setting studied in this paper.

Acknowledgments

We thank A. Knyazev and A. Stathopoulos for discussions, and C. Yang for comments on a previous version of this paper. J. Langou provided the folded spectrum ESCAN interface of P_ARPACK.

References

[1] P. Arbenz, U.L. Hetmaniuk, R.B. Lehoucq, R.S. Tuminaro, Comparison of eigensolvers for large-scale 3D modal analysis using AMG-preconditioned iterative methods, Int. J. Numer. Methods Eng. 64 (2) (2005) 204–236.
[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, Henk van der Vorst, Templates for the Solution of Algebraic Eigenvalue Problems – A Practical Guide, SIAM, Philadelphia, 2000.
[3] A. Canning, J. Dongarra, J. Langou, O. Marques, S. Tomov, C. Vömel, L.-W. Wang, Performance evaluation of eigensolvers in nanostructure computations, in: Proceedings of the IEEE/ACM HPCNano05 and HPCNano06 Workshop, Seattle, WA, 2006.
[4] A. Canning, L.-W. Wang, A. Williamson, A. Zunger, Parallel empirical pseudopotential electronic structure calculations for million atom systems, J. Comput. Phys. 160 (2000) 29–41.
[5] M. Crouzeix, B. Philippe, M. Sadkane, The Davidson method, SIAM J. Sci. Comput. 15 (1) (1994) 62–76.
[6] E.R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices, J. Comput. Phys. 17 (1975) 87–94.
[7] G.H. Golub, C. van Loan, Matrix Computations, third ed., The John Hopkins University Press, Baltimore, Maryland, 1996.
[8] M.E. Hochstenbach, Y. Notay, The Jacobi-Davidson method, GAMM Mitteilungen 29 (2006) 368–382.
[9] A.V. Knyazev, Preconditioned eigensolvers – an oxymoron? Electron. Trans. Numer. Anal. 7 (1998) 104–123.
[10] A.V. Knyazev, Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method, SIAM J. Sci. Comput. 23 (2) (2001) 517–541.
[11] A.V. Knyazev, M.E. Argentati, I. Lashuk, E.E. Ovtchinnikov, Block locally optimal preconditioned eigenvalue solvers (BLOPEX) in Hypre and PETSc, SIAM J. Sci. Comput. 29 (5) (2007) 2224–2239.
[12] A.V. Knyazev, K. Neymeyr, Efficient solution of symmetric eigenvalue problems using multigrid preconditioners in the locally optimal block preconditioned conjugate gradient method, Electron. Trans. Numer. Anal. 15 (2003) 38–55.
[13] W. Kohn, L.S. Sham, Self-consistent equations including exchange and correlation effects, Phys. Rev. 140A (1965) 1133–1140.
[14] R.B. Lehoucq, D.C. Sorensen, C. Yang, ARPACK User's Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods, SIAM, Philadelphia, 1998.
[15] K.J. Maschhoff, D.C. Sorensen, P_ARPACK: an efficient portable large scale eigenvalue package for distributed memory parallel architectures, in: Applied Parallel Computing, Industrial Computation and Optimization, Third International Workshop, PARA 96, Lyngby, Denmark, August 18–21, 1996, Proceedings, Springer, LNCS # 1184, 1996, pp. 478–486.
[16] J.R. McCombs, A. Stathopoulos, Iterative validation of eigensolvers: a scheme for improving the reliability of Hermitian eigenvalue solvers, SIAM J. Sci. Comput. 28 (6) (2006) 2337–2358.
[17] J.R. McCombs, A. Stathopoulos. PRIMME: preconditioned iterative multimethod eigensolver: methods and software description. Technical Report WM-CS-2006-08, Computer Science Department, The College of William and Mary, Williamsburg, VA, 2006.
[18] R.B. Morgan, D.S. Scott, Generalizations of Davidson's method for computing eigenvalues of sparse symmetric matrices, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 817–825.
[19] C.R. Murray, S.C. Racine, E.R. Davidson, Improved algorithms for the lowest few eigenvalues and associated eigenvectors of large matrices, J. Comput. Phys. 103 (1992) 382–389.
[20] J. Nocedal, S.J. Wright, Numerical Optimization, first ed., Springer, New York, 1999.
[21] Y. Notay, Convergence analysis of inexact Rayleigh quotient iteration, SIAM J. Matrix Anal. Appl. 24 (2003) 627–644.
[22] J. Olsen, P. Jørgensen, J. Simons, Passing the one-billion limit in full configuration interaction (FCI) calculations, Chem. Phys. Lett. 169 (6) (1990) 463–472.
[23] C.C. Paige, B.N. Parlett, H.A. van der Vorst, Approximate solutions and eigenvalue bounds from Krylov subspaces, Numer. Linear Algebr. Appl. 2 (2) (1995) 115–133.
[24] B.N. Parlett, The Symmetric Eigenvalue Problem, SIAM Press, Philadelphia, PA, 1998.
[25] M.C. Payne, M.P. Teter, D.C. Allan, T.A. Arias, J.D. Joannopoulos, Iterative minimization techniques for ab initio total-energy calculations: molecular dynamics and conjugate gradients, Rev. Mod. Phys. 64 (4) (1992) 1045–1097.
[26] P. Pulay, Convergence acceleration of iterative sequences. The case of SCF iteration, Chem. Phys. Lett. 73 (2) (1980) 393–398.
[27] P. Pulay, Improved SCF convergence acceleration, J. Comput. Chem. 3 (4) (1982) 556–560.
[28] Y. Saad, Numerical Methods for Large Eigenvalue Problems, Manchester University Press, 1993.
[29] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., SIAM, Philadelphia, 2003.

[30] Y. Saad, J.R. Chelikowsky, S.M. Shontz, Numerical Methods for Electronic Structure Calculations of Materials. Technical Report umsi-2006-15, University of Minnesota, Department of Computer Science and Engineering, March 2006.
[31] Y. Saad, A. Stathopoulos, J.R. Chelikowsky, K. Wu, S. Ögüt, Solution of large eigenvalue problems in electronic structure calculations, BIT 36 (3) (1996) 1–16.
[32] G.L.G. Sleijpen, A.G.L. Booten, D.R. Fokkema, H.A. Van der Vorst, Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems, BIT 36 (3) (1996) 595–633.
[33] G.L.G. Sleijpen, H.A. Van der Vorst, A Jacobi-Davidson iteration method for linear eigenvalue problems, SIAM J. Matrix Anal. Appl. 17 (2) (1996) 401–425.
[34] G.L.G. Sleijpen, F.W. Wubs, Exploiting multilevel preconditioning techniques in eigenvalue computations, SIAM J. Sci. Comput. 25 (4) (2003) 249–1272.
[35] D.C. Sorensen, Implicit application of polynomial filters in a *k*-step Arnoldi method, SIAM J. Matrix Anal. Appl. 13 (1) (1992) 357–385.
[36] A. Stathopoulos, Nearly optimal preconditioned methods for Hermitian eigenproblems under limited memory. Part I: seeking one eigenvalue, SIAM J. Sci. Comput. 29 (2) (2007) 481–514.
[37] A. Stathopoulos, J.R. McCombs, Nearly optimal preconditioned methods for hermitian eigenproblems under limited memory. Part II: seeking many eigenvalues, SIAM J. Sci. Comput. 29 (5) (2007) 2162–2188.
[38] A. Stathopoulos, Y. Saad, Restarting techniques for the (Jacobi-)Davidson symmetric eigenvalue methods, Electron. Trans. Numer. Anal. 7 (1998) 163–181.
[39] A. Stathopoulos, Y. Saad, C.F. Fischer, Robust preconditioning of large sparse symmetric eigenvalue problems, 1995.
[40] G.W. Stewart, Matrix Algorithms II: Eigensystems, SIAM, Philadelphia, 2001.
[41] A.R. Tackett, M. Di Ventra, Targeting specific eigenvectors and eigenvalues of a given Hamiltonian using arbitrary selection criteria, Phys. Rev. B 66 (2002) 245104.
[42] S. Tomov, J. Langou, J. Dongarra, A. Canning, L.-W. Wang, Conjugate-gradient eigenvalue solvers in computing electronic properties of nanostructure architectures, Int. J. Comput. Sci. Eng. 3–4 (2004) 205–212.
[43] C. Vömel, S. Tomov, L.-W. Wang, O.A. Marques, J.J. Dongarra, The use of bulk states to accelerate the band edge state calculation of a semiconductor quantum dot, J. Comput. Phys. 223 (2007) 774–782.
[44] L.-W. Wang, Charge-density patching method for unconventional semiconductor binary systems, Phys. Rev. Lett. 88 (2002) 256402.
[45] L.-W. Wang, A. Zunger, Solving Schrödinger's equation around a desired energy: application to silicon quantum dots, J. Chem. Phys. 100 (1994) 394–2397.
[46] L.-W. Wang, A. Zunger, Pseudopotential calculations of nanoscale CdSe quantum dots, Phys. Rev. B 53 (1996) 9579.
[47] C. Yang, Solving large-scale eigenvalue problems in SCIDAC applications, J. Phys.: Conf. Ser. 16 (2005) 425–434.