Efficient Fock matrix diagonalization by a Krylov-space method

W. Thomas Pollard and Richard A. Friesner[a)]
Columbia University, New York, New York 10027

(Received 1 March 1993; accepted 23 July 1993)

The solution of the Hartree–Fock equations involves the iterative construction of the Fock matrix based on approximate molecular orbitals and the diagonalization of that Fock matrix to obtain new approximations to those orbitals. A significant portion of this work is unnecessary, however, because the occupied molecular orbitals, which are required to construct the Fock matrix, represent a small fraction of the total number of orbitals that are obtained in the diagonalization, and furthermore, typically change little in each iteration. In this paper we introduce a new method which significantly accelerates diagonalization of the Fock matrix by avoiding the unnecessary calculation of the virtual orbitals. Using the occupied orbitals from the previous iteration as an initial guess, accurate updated orbitals are obtained through a combination of diagonalization in the subspace spanned by the occupied orbitals and the mixing of virtual orbital character into the occupied orbitals using a single-vector Lanczos algorithm. Calculations are presented which demonstrate up to 15-fold acceleration of the Fock matrix diagonalizations in a typical problem of 430 orbitals.

I. INTRODUCTION

With the explosive growth in the power of computers over the past 10 years, *ab initio* electronic structure calculations of increasingly large molecules have become feasible. At present, both GAUSSIAN 92 and our own PSGVB program can carry out self-consistent field calculations using as many as 2000 basis functions; for a double-zeta plus polarization (DZP) basis set, this corresponds to molecules of 200 first-row atoms. Powerful and increasingly reliable density functional codes, e.g., DGAUSS and DEMON, have also been developed and are similarly capable of dealing with very large molecular structures.[1]

Assembly of the self-consistent (Fock) operators in both the Hartree–Fock and density functional methods scales asymptotically as N^2, where N is the size of the basis set, and the use of sophisticated cutoff strategies has allowed these limits to be approached in actual calculations. In contrast, diagonalization of the Fock matrix (necessary for most convergence schemes, in particular the DIIS approach of Pulay,[2] which is presently the most efficient available) using standard QR packages scales as N^3. Consequently, a point will be reached where the time for matrix diagonalization dominates the self-consistent field (SCF) iteration process. It is likely, in fact, that this point will be reached for molecules in the 100–200 atom range when numerical methods in the PSGVB program are applied to density functional calculations, work that is currently in progress.

This state of affairs provides a motivation to develop diagonalization methods that are more efficient than the existing QR algorithms, specifically tailored to the problem at hand. Given that one has a good initial guess for the occupied orbitals on every iteration, it seems wasteful not to optimally utilize this information. A reasonable hypothesis is that iterative refinement techniques would be able to start from this initial guess and yield adequately converged answers in less cpu time than would be required for a full QR diagonalization. An additional advantage, of equal if not greater importance for modern hardware, is the ease of parallelization of the multiple iterative refinements of the occupied orbitals, the most expensive part of which is completely independent for each orbital.

A variety of iterative diagonalization methods exist for finding a restricted number of eigenvectors of a matrix. The problem considered here, however, has a number of features not commonly associated with the use of such methods; (1) the matrix is not particularly sparse; (2) there are numerous near degeneracies in the occupied space; and (3) a relatively large number eigenvectors are required. These features are all inimical to the use of iterative methods and account for the use of the QR algorithm in existing SCF *ab initio* programs.

In the present paper we describe a new approach, based on the Lanczos algorithm[3] but containing a number of innovative modifications which allow the difficulties alluded to above to be surmounted. The essential feature of our proposed method is the use of a low-order Krylov-space expansion to refine initial guesses for the occupied molecular orbitals. The n-dimensional Krylov space for a given matrix $\mathbf{F}$ and initial vector $\mathbf{v}_0$ is that space spanned by the vectors $\mathbf{F}^j\mathbf{v}_0$, where $j=0,\ldots,(n-1)$. Thus, this space contains by construction those parts of the overall vector space of $\mathbf{F}$ that are most strongly coupled to $\mathbf{v}_0$. If $\mathbf{v}_0$ is already close to some eigenvector of $\mathbf{F}$, then diagonalization of $\mathbf{F}$ within a Krylov space built upon $\mathbf{v}_0$ would reasonably be expected to yield an improved estimate of this eigenvector.

There are two characteristics of the Hartree–Fock equations that make this a fruitful approach. First, there is typically a significant energy gap between the occupied and virtual orbitals, and the coupling of these two sets of orbitals by an unconverged Fock matrix can be expected to be smaller than the couplings among the occupied orbitals

a)To whom correspondence and reprint requests should be addressed.

 0021-9606/93/99(9)/6742/9/$6.00

themselves. This suggests that it would be productive to first diagonalize the Fock matrix in the space of the occupied orbitals and then allow for the weaker coupling of these orbitals to the virtual orbitals by diagonalizing, again, in a small Krylov space based on each of the occupied-space eigenvectors in turn. Second, because the number of occupied orbitals is typically somewhat less than half the total number of orbitals in the basis set, and the cost of matrix diagonalization scales as the cube of the matrix size, diagonalization in the occupied subspace is quite inexpensive compared to full diagonalization of the Fock matrix. In the problem considered below, for instance, only a fifth of the total number of calculated orbitals are occupied, so that diagonalization of **F** in the occupied subspace is roughly a hundred times faster than full diagonalization.

The new diagonalization algorithm is described in some detail in the next section below, and then calculations are presented which demonstrate the behavior of this method in a typical large-molecule calculation. Substantial timing improvements are obtained over the standard QR packages for a Hartree–Fock calculation on porphin at the DZP level, which requires a Fock matrix of 430 basis functions. We have automated the method in a robust fashion and, indeed, have now incorporated it as a standard component in PSGVB. Future publications will describe a more extensive set of tests on larger molecules, where the advantages are considerably more significant, as well as applications to density functional calculations.

II. THEORY

The Hartree–Fock equations, expressed in a general basis set, take the form of the generalized eigenvalue equation (the Roothaan equation),[4]

$$\mathbf{FC}=\mathbf{SCE}. \tag{1}$$

Here, **F** is the Fock matrix and **S** is the overlap matrix of the basis functions; the eigenvectors and eigenvalues, in matrix **C** and diagonal matrix **E**, respectively, represent the molecular orbitals and their energies. Although the Lanczos algorithm can be applied to this equation directly[3] it is more efficient to first transform the problem into an orthogonal basis set, so that **S** becomes a unit matrix. In fact, this is the usual first step in solving this equation, and it is natural to use the eigenvectors of **S** in the transformation. One particular scheme, canonical orthogonalization, employs the transformation matrix $\mathbf{X}=\mathbf{U}\boldsymbol{\Sigma}^{-1/2}$, where **U** and $\boldsymbol{\Sigma}$ are the eigenvectors and eigenvalues of the overlap matrix, i.e., $\mathbf{SU}=\mathbf{U}\boldsymbol{\Sigma}$. Equation (1) becomes

$$\mathbf{GV}=\mathbf{VE}, \tag{2}$$

where

$$\mathbf{G}=\mathbf{X}^T\mathbf{F}\mathbf{X},$$

$$\mathbf{V}=\mathbf{X}^{-1}\mathbf{C}. \tag{3}$$

At the nth stage of the iterative solution of Eq. (2), the occupied orbitals from the complete set, $\mathbf{V}_n$, define the electronic density matrix, which is needed to construct the Fock matrix, $\mathbf{F}_n$. The transformed Fock matrix, $\mathbf{G}_n$, is then diagonalized to yield a new (complete) set of orbitals, $\mathbf{V}_{n+1}$. That is,

$$\mathbf{G}_n\mathbf{V}_{n+1}=\mathbf{V}_{n+1}\mathbf{E}_{n+1}. \tag{4}$$

The occupied orbitals of this new set are then used to construct a better estimate of the Fock matrix, etc., until a self-consistent Fock matrix and orbital set are obtained. As self-consistency is approached, the orbitals $\mathbf{V}_{n+1}$ will be less and less different from the orbitals $\mathbf{V}_n$ that were used to construct $\mathbf{G}_n$. With the Lanczos refinement algorithm, we can obtain, at each stage, just the occupied orbitals of $\mathbf{V}_{n+1}$, using the $\mathbf{V}_n$ as an initial guess.

The application of our proposed Lanczos refinement method to a set of approximate eigenvectors of the Fock matrix is an iterative process involving four stages. First, the Fock matrix $\mathbf{G}_n$ is diagonalized in the subspace of the occupied (approximate) orbitals, $\mathbf{V}_n$, since their mutual couplings are expected to be significantly stronger than the coupling of the occupied and virtual orbitals. This is followed by the Lanczos refinement of each occupied eigenvector, in turn, to incorporate the couplings to the virtual orbitals. The refined eigenvectors are reorthogonalized and finally $\mathbf{G}_n$ is diagonalized again over the refined eigenvectors. The latter step is necessary to resolve secondary couplings among the occupied eigenvectors through the virtual orbitals. The convergence of the refined eigenvectors is checked, and any unconverged vectors are subjected to additional cycles of refinement and rediagonalization. Each of these stages is considered now, in turn.

A. Diagonalization in occupied subspace

The first step is to diagonalize the $N\times N$ matrix **G** in the space of the N_{occ} occupied orbitals. Writing $\mathbf{V}'$ to denote the matrix whose columns are the occupied orbitals of **V**, then $\mathbf{G}'=\mathbf{V}'^T\mathbf{G}\mathbf{V}'$ is the representation of **G** in the space of the occupied orbitals. Its eigenvalue equation is

$$\mathbf{G}'\mathbf{D}'=\mathbf{D}'\mathbf{E}. \tag{5}$$

The columns of the matrix $\mathbf{D}=\mathbf{V}'\mathbf{D}'$, are therefore approximate eigenvectors of **G** in which only couplings among the occupied-space vectors of $\mathbf{V}'$ have been considered. The vectors of **D** now become the object of Lanczos refinement to include the weaker couplings to the virtual orbitals. In the next section, we describe in detail the refinement of a given trial vector out of this set.

A final point is that in a typical problem, where N_{occ} is significantly smaller than N, the computational cost of diagonalization in the occupied subspace is dominated by the construction of the $\mathbf{G}'$ matrix and that the cost of the actual diagonalization of $\mathbf{G}'$ is negligible. The construction of the $\mathbf{G}'$ requires the multiplication of each initial (approximate) eigenvector by the full **G** matrix, so that the overall construction scales roughly as N^2N_{occ}; the diagonalization, in contrast, scales simply as N_{occ}^3.

B. Lanczos refinement of a trial eigenvector

Given a symmetric matrix **G** and an initial (normalized) vector $\mathbf{w}_0$, the Lanczos algorithm describes the con-

struction of an orthonormal basis set, $\{\mathbf{w}_j\}$, that spans the n-dimensional space defined by $n-1$ applications of $\mathbf{G}$ to $\mathbf{w}_0$ (the Krylov space); the special property of the Lanczos basis is that it gives $\mathbf{G}$ a tridiagonal representation. The basis vectors are defined by the recurrence relation[3]

$$\beta_{j+1}\mathbf{w}_{j+1}=\mathbf{G}\mathbf{w}_j-\alpha_j\mathbf{w}_j-\beta_j\mathbf{w}_{j-1}, \tag{6}$$

where

$$\alpha_j=\mathbf{w}_j^T\mathbf{G}\mathbf{w}_j=\mathbf{T}_{j,j},$$
$$\beta_{j+1}=\mathbf{w}_{j+1}^T\mathbf{G}\mathbf{w}_j=\mathbf{T}_{j+1,j}, \tag{7}$$

with the definition $\beta_0=0$; note that $\beta_1=0$ when the initial vector $\mathbf{w}_0$ is already an eigenvector of G. Here, $\mathbf{w}_j$ is the jth orthonormal basis vector and $\mathbf{T}_{i,j}$ is a matrix element of $\mathbf{G}$ in this basis. Using the recurrence relation, Eq. (6), the $\{\mathbf{w}_j\}$ are constructed iteratively; starting from a given vector $\mathbf{w}_j$, the next basis vector $\mathbf{w}_{j+1}$ is obtained by

$$\begin{aligned}
&(1)\ \mathbf{u}_{j+1}=\mathbf{G}\mathbf{w}_j,\\
&(2)\ \alpha_j=\mathbf{w}_j^T\mathbf{u}_{j+1},\\
&(3)\ \mathbf{u}'_{j+1}=\mathbf{u}_{j+1}-\alpha_j\mathbf{w}_j-\beta_j\mathbf{w}_{j-1},\\
&(4)\ \beta_{j+1}=(\mathbf{u}'^T_{j+1}\mathbf{u}'_{j+1})^{1/2},
\end{aligned} \tag{8}$$

and finally,

$$(5)\ \mathbf{w}_{j+1}=\mathbf{u}'_{j+1}/\beta_{j+1}.$$

We denote by $\mathbf{T}_{(N_{\mathrm{kry}})}$ the $N_{\mathrm{kry}}\times N_{\mathrm{kry}}$ (tridiagonal) matrix that is generated by $N_{\mathrm{kry}}-1$ iterations of Eq. (8), and by $\mathbf{W}_{(N_{\mathrm{kry}})}$, the $N\times N_{\mathrm{kry}}$ matrix whose columns are the Lanczos basis vectors $\mathbf{w}_0,\ldots,\mathbf{w}_{N_{\mathrm{kry}}-1}$. Writing the eigenvalue equation for $\mathbf{T}_{(N_{\mathrm{kry}})}$ as

$$\mathbf{T}_{(N_{\mathrm{kry}})}\mathbf{Y}'=\mathbf{Y}'\mathbf{E}, \tag{9}$$

the columns of $\mathbf{Y}=\mathbf{W}\mathbf{Y}'$ are the Krylov-space eigenvectors expressed in the full basis set of $\mathbf{G}$. As N_{kry} approaches N, the eigenvalues of $\mathbf{T}_{(N_{\mathrm{kry}})}$ will converge on those of $\mathbf{G}$. It is well known that different eigenvalues converge at different rates and that, typically, the extreme eigenvalues converge most quickly.[3] In our application, since $\mathbf{w}_0$ is assumed to be close to some particular eigenvector of $\mathbf{G}$, this eigenvector is expected to converge especially quickly with increasing N_{kry}. $\mathbf{T}_{(N_{\mathrm{kry}})}$ is diagonalized for every N_{kry} and its eigenvectors are checked for convergence to some arbitrary level of accuracy. Once the eigenvector that correlates most strongly with $\mathbf{w}_0$ has satisfied the convergence criterion, it is accepted and iteration is halted.

One advantage of the Lanczos algorithm is the convenience with which convergence of the approximate eigenvectors can be estimated. A natural criterion to use is β^2, where β is the value of β_1 that would be obtained if the approximate eigenvector were used as another initial vector $\mathbf{w}_0$ in Eqs. (6), (7), and (8). That is, for some approximate (normalized) eigenvector $\mathbf{y}_i$,

$$\beta^2=\|(\mathbf{G}-\mathbf{y}_i^T\mathbf{G}\mathbf{y}_i)\mathbf{y}_i\|^2, \tag{10}$$

which vanishes when $\mathbf{y}_i$ is an eigenvector of $\mathbf{G}$. When $\mathbf{y}_i$ is close to some eigenvector $\mathbf{z}$ of $\mathbf{G}$, it can be shown (see Appendix) that β^2 is proportional to both the error in the eigenvector, $1-|\mathbf{z}^T\mathbf{y}_i|$, and to a factor, $\lambda^2_{(i)}$, that is different for every eigenvector of $\mathbf{G}$. In the porphin calculation presented below, the quantity $\lambda^2_{(i)}$ is found to be on the order or unity or greater (up to two orders of magnitude greater). In our implementation, we conservatively assume that β^2 is a direct measure of the absolute error. The calculation of β^2 is simplified when $\mathbf{y}_i$ is the ith eigenvector of $\mathbf{T}_{(N_{\mathrm{kry}})}$, in which case

$$\beta^2=(\beta_{N_{\mathrm{kry}}}\mathbf{Y}'_{N_{\mathrm{kry}}-1,i})^2, \tag{11}$$

$$=(\beta_{N_{\mathrm{kry}}}\mathbf{w}^T_{N_{\mathrm{kry}}-1}\mathbf{y}_i)^2, \tag{12}$$

where $\beta_{N_{\mathrm{kry}}}=\mathbf{w}^T_{N_{\mathrm{kry}}+1}\mathbf{G}\mathbf{w}_{N_{\mathrm{kry}}}$ was calculated in the last iteration of Eq. (8) and $\mathbf{w}_{N_{\mathrm{kry}}-1}$ is the highest-order Krylov-space basis vector. Thus, no extra work, in particular no additional multiplication with $\mathbf{G}$, is required to estimate the convergence of the eigenvectors of $\mathbf{T}_{(N_{\mathrm{kry}})}$.

In practice, the convergence of the eigenvector being refined is initially rapid, but eventually reaches a point at which further expansion of the Krylov space no longer improves the refined eigenvector. For most examples considered, β^2 actually increases significantly, or fluctuates, with N_{kry} after the initial period of rapid convergence. Presumably this behavior is related to the well-known characteristic of the Lanczos algorithm that orthonormality of the Lanczos basis vectors is lost once some of the eigenvectors of $\mathbf{T}_{(N_{\mathrm{kry}})}$ have converged.[3] Because of this, however, it is necessary to set a maximum size on the Krylov space used for refinement. As the Krylov space is increased in size, the best eigenvector obtained, i.e., the one with the lowest β^2 value, is always kept if the convergence criterion is not met before the Krylov-space size limit is reached. Eigenvectors which have not converged are marked and resubmitted to Lanczos refinement after rediagonalizing the Fock matrix in the space of the refined eigenvectors.

C. Reorthogonalization and final diagonalization

Once each of the approximate eigenvectors has been refined, the Fock matrix must be diagonalized again in the space of the refined eigenvectors. This step serves two purposes. First, although the exact eigenvectors of $\mathbf{G}$ are naturally orthonormal, the approximate eigenvectors that result from the Lanczos refinement will typically not be. Therefore, they must be reorthogonalized, which can be done straightforwardly and inexpensively using the standard Gram–Schmidt procedure.[5] However, the results of a Gram–Schmidt orthogonalization depend upon the order in which the vectors are treated, and this introduces an undesirable source of ambiguity in the results. Furthermore, the eigenvalue obtained for each eigenvector in the refinement stage is generally no longer correct for the reorthogonalized eigenvectors. As it is, the cost of computing the diagonal matrix elements of $\mathbf{G}$ for these eigenvectors is almost as expensive as a complete rediagonalization of the

Fock matrix in the occupied subspace, since no further multiplications by **G** are required to compute the off-diagonal matrix elements. Thus, diagonalizing the Fock matrix using the reorthogonalized eigenvectors as a basis inexpensively removes the dependence of the final eigenvectors on the details of the Gram–Schmidt orthogonalization and ensures that the final approximate eigenvectors and eigenvalues are self-consistent.

A more important reason for rediagonalizing is that the Lanczos refinement reintroduces couplings among the occupied orbitals. This is especially problematic in the case of near-degenerate sets of eigenvectors, in which even small couplings can cause significant mixing. Furthermore, when nearby approximate eigenvectors are significantly coupled, the β^2 test for convergence can be highly inaccurate, because the factor $\lambda^2_{(i)}$ will be very small for near-degenerate eigenvectors, causing the actual error in the refinement to be significantly underestimated. Rather than try to identify and treat these situations explicitly, it is equally effective, and simpler, to resolve them by a complete rediagonalization in the occupied subspace.

Writing $\mathbf{V}''$ to denote the matrix whose columns are the Lanczos-refined eigenvectors, $\mathbf{G}''=\mathbf{V}''^T\mathbf{G}\mathbf{V}''$ is the representation of **G** in this new space and

$$\mathbf{G}''\mathbf{D}''=\mathbf{D}''\mathbf{E}. \tag{13}$$

The matrix $\mathbf{V}_{\text{final}}=\mathbf{V}''\mathbf{D}''$, then, holds the final refined eigenvectors generated by a single cycle of this algorithm, and the diagonal elements of **E** are the final eigenvalues.

III. CALCULATIONS

As a challenging sample problem to test the performance of our proposed method, we chose a calculation of the closed-shell ground state of the molecule porphin. This molecule, which contains 28 heavy atoms and 14 hydrogen atoms, represents an intermediate-size problem by current standards; a 6-31G** basis set of 430 atomic orbitals was used to obtain 81 occupied molecular orbitals. Eight SCF iterations were required for convergence of the Fock matrix and the occupied molecular orbitals to self-consistency; the iterative Lanczos refinement algorithm was used to obtain the occupied molecular orbitals for the new Fock matrix at each stage in this sequence, using the eigenvectors from the previous Fock matrix as initial guesses. To evaluate the ability to control the error in the calculated orbitals, each matrix was diagonalized using four different target error levels ($\beta^2_{\max}$).

Table I displays results illustrating the convergence of individual eigenvectors in one particular calculation. The Fock matrix in this example was from the second SCF iteration and the eigenvectors were calculated to an error level of 10^{-8}; the maximum size of the Krylov space used before rediagonalizing was eight. This case was the most demanding of all examples considered because the Fock matrix was the most poorly converged and the target error level was fairly strict, yet all 81 occupied eigenvectors were obtained at the specified error level in less than half of the time that would have been required for a QR diagonalization of the full Fock matrix.

The main point to take from Table I is that the amount of work necessary to converge a given eigenvector varied considerably, with those eigenvectors near the energy gap requiring much more work than those far from the gap. The 24 lowest-energy eigenvectors converged quite quickly, requiring only one cycle of Lanczos refinement, while five of the eight highest-energy eigenvectors needed four cycles to satisfy the specified error level. To some extent, this behavior reflected the quality of the initial guess used for Lanczos refinement (reflected in the third column of Table I). However, there was also a real difference in the rate of convergence and eigenvectors farther from the energy gap generally converged more rapidly than those higher in energy; for instance, eigenvectors 54 and 55, which have fairly large initial errors, converged in just two cycles, while eigenvectors 79 and 81 had much smaller initial errors, but took the most effort to converge.

Similarly, the final accuracy attained for each eigenvector was significantly lower for those farther from the energy gap. In particular, the first 24 eigenvectors significantly overstepped the error goal of 10^{-8}, uniformly reaching levels below 10^{-10}; the four lowest eigenvectors were all below 10^{-12}! To some extent, this was another manifestation of the more rapid rate of convergence of the lower-energy eigenvectors. To an equal degree this also reflects the greater discrepancy between the error estimator β^2 and the actual error for the lower-energy eigenvectors. As shown in the Appendix, β^2 is proportional to the error and the factor $\lambda^2_{(i)}$, which is roughly the average of the square of the energy difference between the refined (approximate) eigenvector and the other (exact) eigenvectors with which it is still mixed. If the residual couplings of the occupied eigenvectors are largely to the virtual eigenvectors, then $\lambda^2_{(i)}$ will be on the order of the square of the distance between the eigenvectors and the energy gap. Empirically, it was observed that this factor is in the range of 400–600 for eigenvectors 1–4, around 200 for eigenvectors 5–24, and between 1 and 10 for most of the higher-energy eigenvectors.

Tables II, III, and IV present overall results obtained with the increasingly well-converged Fock matrices from later SCF iterations. The numbers given in Tables II and III are averages over all 81 eigenvectors; the behavior of the individual eigenvectors in all of these cases followed the general patterns discussed above. Note that these two tables also include the average error of the initial eigenvectors, before any treatment; this number is the most direct indication of the degree of convergence of the Fock matrix. For the calculation summarized in Table I, for instance, the average error in the initial eigenvalues is 0.035, which is larger than the average separation among the eigenvalues. Comparing these initial errors with the errors after the initial diagonalization, we see that the error typically drops 2–3 orders of magnitude in this stage, making it the most significant stage in the overall refinement process. In fact, the initial approximate eigenvalues could not be included in Table I because it was not consistently possible to assign the approximate eigenvectors to the exact eigenvectors until after the initial diagonalization.

TABLE I. Intermediate results in the iterative refinement of the 81 occupied orbitals of porphin. The absolute error in the approximate eigenvalues is shown after each stage in the refinement process. The Fock matrix was taken from the second SCF iteration. The maximum size of the Krylov space in each refinement cycle was 8 and the target error level was 10^{-8}.

No.	Energy	Init. Diag.	Cycle 1	Cycle 2	Cycle 3	Cycle 4
1	−15.6699	0.192×10^{-5}	0.140×10^{-12}			
2	−15.6626	0.193×10^{-5}	0.141×10^{-12}			
3	−15.2201	0.181×10^{-5}	0.387×10^{-12}			
4	−15.2181	0.197×10^{-5}	0.499×10^{-12}			
5	−11.2246	0.241×10^{-6}	0.231×10^{-10}			
6	−11.2212	0.223×10^{-6}	0.192×10^{-10}			
7	−11.1924	0.277×10^{-6}	0.222×10^{-10}			
⋮	⋮	⋮	⋮			
22	−11.1265	0.323×10^{-5}	0.706×10^{-12}			
23	−11.1253	0.333×10^{-5}	0.119×10^{-11}			
24	−11.1233	0.102×10^{-5}	0.911×10^{-12}			
25	−1.2301	0.152×10^{-2}	0.290×10^{-6}	0.106×10^{-9}		
26	−1.2280	0.147×10^{-2}	0.250×10^{-6}	0.727×10^{-10}		
27	−1.1618	0.196×10^{-3}	0.207×10^{-7}	0.982×10^{-10}		
28	−1.1557	0.312×10^{-3}	0.732×10^{-7}	0.349×10^{-9}		
⋮	⋮	⋮	⋮	⋮		
66	−0.4720	0.603×10^{-2}	0.201×10^{-5}	0.818×10^{-9}		
67	−0.4698	0.385×10^{-1}	0.450×10^{-5}	0.185×10^{-8}	0.120×10^{-8}	
68	−0.4659	0.428×10^{-2}	0.107×10^{-5}	0.142×10^{-8}		
69	−0.4603	0.446×10^{-1}	0.338×10^{-5}	0.241×10^{-8}		
70	−0.4560	0.116×10^{-1}	0.427×10^{-5}	0.364×10^{-8}	0.251×10^{-8}	
71	−0.4215	0.778×10^{-2}	0.687×10^{-5}	0.513×10^{-8}	0.152×10^{-8}	
72	−0.4168	0.740×10^{-2}	0.706×10^{-5}	0.765×10^{-8}	0.280×10^{-8}	
73	−0.3991	0.231×10^{-2}	0.135×10^{-5}	0.305×10^{-8}		
74	−0.3347	0.187×10^{-1}	0.855×10^{-5}	0.122×10^{-6}	0.649×10^{-8}	0.511×10^{-8}
75	−0.3308	0.238×10^{-1}	0.128×10^{-4}	0.869×10^{-7}	0.484×10^{-8}	0.395×10^{-8}
76	−0.3169	0.523×10^{-2}	0.340×10^{-5}	0.212×10^{-7}	0.364×10^{-8}	
77	−0.3039	0.184×10^{-1}	0.306×10^{-4}	0.497×10^{-7}	0.396×10^{-8}	
78	−0.2856	0.823×10^{-2}	0.128×10^{-4}	0.402×10^{-7}	0.792×10^{-8}	
79	−0.2792	0.893×10^{-2}	0.725×10^{-4}	0.515×10^{-6}	0.291×10^{-7}	0.189×10^{-8}
80	−0.1929	0.129×10^{-1}	0.449×10^{-4}	0.284×10^{-6}	0.357×10^{-7}	0.123×10^{-7}
81	−0.1772	0.469×10^{-2}	0.156×10^{-4}	0.400×10^{-6}	0.519×10^{-7}	0.128×10^{-7}

TABLE II. The overall rms errors in the approximate eigenvalues before treatment, after the initial diagonalization in the subspace of the occupied orbitals, and after each cycle of Lanczos refinement and rediagonalization. The results are given for Fock matrices from a sequence of SCF iterations and for increasingly stringent target error levels. The maximum size of the Krylov space in each refinement cycle was 8.

Iter.	$\beta^2_{\max}$	Input	Init. Diag.	Cycle 1	Cycle 2	Cycle 3	Cycle 4
	10^{-5}	0.349×10^{-1}	0.469×10^{-3}	0.208×10^{-5}	0.261×10^{-6}		
2	10^{-6}			0.208×10^{-5}	0.287×10^{-7}	0.281×10^{-7}	
	10^{-7}			0.208×10^{-5}	0.890×10^{-8}	0.255×10^{-8}	
	10^{-8}			0.208×10^{-5}	0.980×10^{-8}	0.704×10^{-9}	0.294×10^{-9}
	10^{-5}	0.105×10^{-1}	0.797×10^{-4}	0.408×10^{-6}	0.268×10^{-6}		
3	10^{-6}			0.368×10^{-6}	0.282×10^{-7}		
	10^{-7}			0.368×10^{-6}	0.341×10^{-8}	0.311×10^{-8}	
	10^{-8}			0.368×10^{-6}	0.252×10^{-8}	0.284×10^{-9}	0.264×10^{-9}
	10^{-5}	0.266×10^{-2}	0.423×10^{-5}	0.249×10^{-6}			
4	10^{-6}			0.252×10^{-7}	0.232×10^{-7}		
	10^{-7}			0.165×10^{-7}	0.266×10^{-8}		
	10^{-8}			0.164×10^{-7}	0.365×10^{-9}	0.312×10^{-9}	
	10^{-5}	0.233×10^{-3}	0.144×10^{-6}	0.144×10^{-6}			
6	10^{-6}			0.457×10^{-7}			
	10^{-7}			0.273×10^{-8}			
	10^{-8}			0.682×10^{-9}	0.326×10^{-9}		
	10^{-5}	0.287×10^{-4}	0.240×10^{-7}	0.240×10^{-7}			
8	10^{-6}			0.240×10^{-7}			
	10^{-7}			0.390×10^{-8}			
	10^{-8}			0.278×10^{-9}			

TABLE III. The overall rms errors in the approximate eigenvectors before treatment, after the initial diagonalization in the subspace of the occupied orbitals, and after each cycle of Lanczos refinement and rediagonalization. The error is defined as the difference from unity of the overlap of the exact and approximate eigenvectors. The results are given for Fock matrices from a sequence of SCF iterations and for increasingly stringent target error levels. The maximum size of the Krylov space in each refinement cycle was 8.

Iter.	$\beta^2_{\max}$	Input	Init. Diag.	Cycle 1	Cycle 2	Cycle 3	Cycle 4
	10^{-5}	0.360×10^{-1}	0.251×10^{-2}	0.122×10^{-5}	0.211×10^{-6}		
2	10^{-6}			0.122×10^{-5}	0.194×10^{-7}	0.189×10^{-7}	
	10^{-7}			0.122×10^{-5}	0.829×10^{-8}	0.234×10^{-8}	
	10^{-8}			0.122×10^{-5}	0.902×10^{-8}	0.878×10^{-9}	0.285×10^{-9}
	10^{-5}	0.272×10^{-1}	0.139×10^{-2}	0.310×10^{-6}	0.206×10^{-6}		
3	10^{-6}			0.304×10^{-6}	0.219×10^{-7}		
	10^{-7}			0.304×10^{-6}	0.262×10^{-8}	0.235×10^{-8}	
	10^{-8}			0.304×10^{-6}	0.227×10^{-8}	0.223×10^{-9}	0.209×10^{-9}
	10^{-5}	0.998×10^{-2}	0.416×10^{-5}	0.165×10^{-6}			
4	10^{-6}			0.169×10^{-7}	0.155×10^{-7}		
	10^{-7}			0.152×10^{-7}	0.202×10^{-8}		
	10^{-8}			0.152×10^{-7}	0.339×10^{-9}	0.219×10^{-9}	
	10^{-5}	0.483×10^{-2}	0.176×10^{-6}	0.176×10^{-6}			
6	10^{-6}			0.375×10^{-7}			
	10^{-7}			0.227×10^{-8}			
	10^{-8}			0.719×10^{-9}	0.244×10^{-9}		
	10^{-5}	0.225×10^{-3}	0.430×10^{-7}	0.430×10^{-7}			
8	10^{-6}			0.430×10^{-7}			
	10^{-7}			0.358×10^{-8}			
	10^{-8}			0.171×10^{-9}			

The results in Tables II and III demonstrate that the new algorithm was able to obtain converged results to any specified error level in every case that was examined. Comparing the final error for each calculation with its target error level, $\beta^2_{\max}$, it is also apparent that there is a fairly constant factor by which the error actually achieved exceeded the target level, reflecting the spread in the $\lambda^2_{(i)}$ values and the rates of convergence of the individual eigen-

TABLE IV. The time required for the initial diagonalization in the subspace of the occupied orbitals and for each cycle of Lanczos refinement and rediagonalization. The time is given as a fraction of the time required for full diagonalization of the Fock matrix; the number of eigenvectors treated in each stage is noted in parentheses. The results are given for Fock matrices from a sequence of SCF iterations and for increasingly stringent target error levels. The maximum size of the Krylov space in each refinement cycle was 8.

Iter.	$\beta^2_{\max}$	Init. Diag.	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Total
	10^{-5}	0.04 (81)	0.20 (81)	0.04 (34)			0.29
2	10^{-6}		0.21 (81)	0.08 (54)	0.00 (1)		0.34
	10^{-7}		0.21 (81)	0.11 (57)	0.01 (8)		0.39
	10^{-8}		0.22 (81)	0.13 (57)	0.03 (17)	0.00 (5)	0.45
	10^{-5}	0.04 (81)	0.16 (81)	0.00 (6)			0.21
3	10^{-6}		0.20 (81)	0.03 (26)			0.28
	10^{-7}		0.21 (81)	0.08 (54)	0.00 (2)		0.34
	10^{-8}		0.21 (81)	0.11 (57)	0.02 (9)	0.00 (2)	0.40
	10^{-5}	0.04 (81)	0.08 (61)				0.13
4	10^{-6}		0.13 (81)	0.00 (1)			0.18
	10^{-7}		0.19 (81)	0.01 (11)			0.25
	10^{-8}		0.21 (81)	0.06 (50)	0.00 (1)		0.33
	10^{-5}	0.04 (81)	0.02 (0)				0.06
6	10^{-6}		0.03 (19)				0.08
	10^{-7}		0.09 (59)				0.13
	10^{-8}		0.14 (81)	0.00 (3)			0.19
	10^{-5}	0.04 (81)	0.02 (0)				0.06
8	10^{-6}		0.02 (0)				0.06
	10^{-7}		0.02 (10)				0.07
	10^{-8}		0.07 (54)				0.11

TABLE V. The total diagonalization time is given as a function of the maximum size Krylov space size in each refinement cycle; times are expressed a fraction of the time required for full QR diagonalization of the Fock matrix. Results are given for Fock matrices from a sequence of SCF iterations and for increasingly stringent error levels. Asterisks mark those calculations in which some eigenvectors could not be refined to the specified error level.

Iter.	$\beta^2_{\max}$	4	5	6	8	10	12	14	16	18
	10^{-5}	0.27	0.27	0.26	0.29	0.32	0.37	0.42	0.46	0.51
2	10^{-6}	0.42*	0.40*	0.31	0.34	0.38	0.43	0.49	0.55	0.62
	10^{-7}	0.64*	0.50*	0.38	0.39	0.43	0.47	0.53	0.60	0.67
	10^{-8}	1.02*	0.69*	0.44	0.45	0.48	0.53	0.58	0.65	0.74
	10^{-5}	0.20	0.21	0.20	0.21	0.21	0.21	0.22	0.23	0.23
3	10^{-6}	0.28	0.29	0.28	0.28	0.32	0.35	0.39	0.43	0.49
	10^{-7}	0.44*	0.35	0.33	0.34	0.38	0.43	0.48	0.54	0.61
	10^{-8}	0.94*	0.45	0.40	0.41	0.43	0.50	0.53	0.60	0.69
	10^{-5}	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13
4	10^{-6}	0.18	0.18	0.18	0.18	0.18	0.19	0.18	0.18	0.18
	10^{-7}	0.26	0.25	0.25	0.25	0.25	0.26	0.25	0.25	0.26
	10^{-8}	0.39*	0.30	0.31	0.33	0.30	0.32	0.32	0.33	0.34
	10^{-5}	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
6	10^{-6}	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08
	10^{-7}	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13
	10^{-8}	0.20	0.19	0.18	0.19	0.19	0.19	0.19	0.19	0.19
	10^{-5}	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
8	10^{-6}	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
	10^{-7}	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07	0.07
	10^{-8}	0.12	0.12	0.11	0.11	0.11	0.12	0.11	0.11	0.11

vectors. The exceptions were those calculations in which the errors of most of the eigenvectors already exceeded the error criterion following the initial diagonalization.

Table IV summarizes the computer time required for each stage of these calculations, as well as the total time to diagonalize each Fock matrix to each of the target error levels. Together with Tables II and III, it shows that the effort required to diagonalize the Fock matrix is significantly reduced as the quality of the initial approximate eigenvectors is improved and as the target error level, $\beta^2_{\max}$, is relaxed. In Table IV, we see that even in the worst case, the diagonalization of the second Fock matrix to an error level of 10^{-8}, the total time required is only 0.45, or 45% of the time required for full diagonalization. Relaxing the error level to 10^{-5} reduces this time to 0.29, a significant savings in time if greater accuracy is not required at this stage; as Tables II and III show, even the exact eigenvectors for the second Fock matrix are poor approximations to the eigenvectors of the next (third) Fock matrix, with an average error of 10^{-2}. Diagonalization times are uniformly reduced for the later Fock matrices. In the best cases, the occupied eigenvectors were obtained in a time of just 0.06. This represents the point at which no Lanczos refinement at all was required after the initial diagonalization, which itself requires a time of 0.04; the additional time of 0.02 is that which required to multiply each approximate eigenvector into the Fock matrix once, in order to calculate β^2 and test for convergence.

The only adjustable parameter which affects the efficiency of the method is the size limit imposed on the Krylov space used for Lanczos refinement. In Table V we present results showing the dependence of the overall diagonalization times on this parameter. For the poorly converged Fock matrices in iterations 2 and 3, diagonalization times depended quite strongly on the Krylov space limit. When the limit was relaxed to 18, for instance, the time required for diagonalization of the second Fock matrix to an accuracy of 10^{-8} went up to 0.74. On the other hand, for well-converged Fock matrices there was almost no dependence on this parameter.

This behavior reflects the fact that some eigenvectors are unable to converge to the desired error level in a single Lanczos refinement step, i.e., they reach an impasse at which increasing the size of the Krylov space no longer improves the approximate eigenvector. Once a refinement reaches this stage, the additional work to increase the size of the Krylov space is simply wasted, since no progress can be made until couplings among the refined eigenvectors are resolved by explicit rediagonalization of the Fock matrix over all the vectors. For the more well-converged Fock matrices, when all or most of the eigenvectors can reach the target error level in a single Lanczos refinement cycle, it does not matter where the size limit is set. For the other cases, it is more efficient to truncate the Lanczos refinement early than to let it proceed too long, and so better results are obtained for lower limits on the size of the Krylov space used in each cycle. However, the limit can also be too low; the starred entries in Table V, for Krylov-space limits of 4 and 5, represent calculations in which it was *impossible* to converge all of the eigenvectors to the desired accuracy. The optimum Krylov-space limit was found to be in the range of 6–8; the timings for these values were essentially the same, although fewer refinement cycles were generally required when the limit was 8. Therefore, this value seems to represent a good compromise between

safety and efficiency, and the calculations in Tables I, II, III, and IV were all performed with this limit.

IV. CONCLUSIONS

We have described a new algorithm which allows one to rapidly obtain a large block of the eigenvectors of a symmetric matrix when reasonable initial guesses for those eigenvectors are available. The method has been introduced and demonstrated in the context of the diagonalization of the Fock matrix in self-consistent-field (Hartree–Fock) calculations, in which the occupied orbitals obtained from one Fock matrix can be used as good initial guesses for diagonalizing the new Fock matrix constructed from those orbitals.

Our algorithm is based on the refinement of each approximate eigenvector by diagonalizing the Fock matrix in a small Krylov space built on the trial vector using the Lanczos algorithm. The size of the Krylov space is increased in steps until the eigenvector under refinement is converged to some specified accuracy, estimated by the convergence estimator β^2, or a maximum Krylov-space size is reached. This becomes significantly more efficient when it is preceded and followed by the diagonalization of the Fock matrix in the complete subspace of the approximate eigenvectors (occupied orbitals), which is typically very inexpensive compared to full diagonalization of the Fock matrix. The prior diagonalization treats the stronger couplings among the occupied orbitals exactly, allowing the weaker coupling of the resulting eigenvectors to the virtual orbitals to be handled using small Krylov-space diagonalizations; the subsequent diagonalization in the space of the refined eigenvectors resolves secondary couplings introduced among the occupied orbitals by their mixing with the virtual orbitals; this step is particularly critical for the treatment of near-degenerate sets of eigenvectors. The proposed algorithm is most efficient when the Krylov-space refinements are limited to a fairly small size; approximate eigenvectors that are not converged to the desired accuracy in a single cycle of Lanczos refinement and rediagonalization are then submitted to additional cycles of refinement and diagonalization until converged. In this form, the method is fast, stable, and reliable.

The algorithm has a number of features that make it computationally attractive. First, it consists predominantly of matrix-vector multiplications, which means that, unlike the traditional QR diagonalization algorithm, it can be implemented quite efficiently on modern parallel- and vector-processing computer architectures. The Krylov-space refinement method itself is naturally parallelizable at a higher level, since it treats each approximate eigenvector independently. Second, because the the matrix to be diagonalized enters only in the context of matrix–vector and matrix–matrix multiplication, it is not modified in any way. Thus, much less computer storage is required than in diagonalization methods which explicitly transform the original matrix; less storage is also required for the eigenvectors themselves, since only the relevant eigenvectors are calculated. These advantages are increased tremendously when the matrix to be diagonalized is sparse, or otherwise structured, since this makes multiplication with the matrix significantly cheaper and computer storage for the full matrix need never be allocated at all. Finally, the ability to specify the accuracy of the eigenvectors allows additional economies, since in many iterative problems, such as SCF calculations, the eigenvectors obtained in the early iterations need not be as precise as those required for the final solution. Practical experience suggests that error levels of 10^{-4} or 10^{-5} are completely adequate for the first few SCF cycles; for the porphin calculation this would allow an overall savings in diagonalization time of more than 80%.

These ideas were successfully demonstrated in a calculation of the molecular orbitals of the molecule porphin. In this system, 81 occupied molecular orbitals were obtained from a Fock matrix constructed in a 430-orbital basis set. The new method allowed the accurate calculation of these orbitals to specified accuracies between 10^{-5}–10^{-8} in one-half to one-fifteenth of the time required for complete diagonalization, depending on the quality of the initial guesses, i.e., on the level of convergence of the Fock matrix, and on the specified accuracy level. For poorly converged Fock matrices, the time required for the diagonalization depended strongly on the maximum Krylov-space size allowed per refinement cycle, and a Krylov-space size limit of 6–8 was found to be optimal. For well-converged Fock matrices, the diagonalization time approached an asymptotic lower limit representing a 15-fold speedup over full QR diagonalization of the Fock matrix; this limit reflects the time for a single diagonalization in the occupied subspace and the testing of each eigenvector for convergence.

It is anticipated that this method will be generally useful in Hartree–Fock electronic structure calculations, especially as the size of these system increases and diagonalization becomes the rate-limiting step in the convergence of the Fock matrix. Moreover, for sufficiently large molecules, the Fock matrix will begin to display usable sparsity, in which case our diagonalization algorithm scales asymptotically as N^2, rather than N^3. The method will be similarly applicable in density-functional calculations of electronic structure. In addition, it should be useful in the broader range of calculations in which a series of similar matrices must be diagonalized and only a subset of the eigenvectors required. Applications in these more general contexts will be explored in subsequent papers.

ACKNOWLEDGMENTS

R.A.F. is a Camille and Henry Dreyfus Teacher–Scholar and a recipient of a Research Career Development Award from the National Institutes of Health. W.T.P. acknowledges the support of a National Science Foundation Postdoctoral Research Fellowship in Chemistry. This work was supported in part by a grant from the National Science Foundation to R.A.F.

APPENDIX

Let $\mathbf{r}$ be an N vector that is meant to be an approximation to $\mathbf{z}_i$, the ith eigenvector of the $N \times N$ symmetric

matrix **A**, and let λ_i be the eigenvalue associated with $\mathbf{z}_i$. Then, we characterize the error in the approximate eigenvector by the quantity ϵ, defined as the difference in the overlap of **r** and $\mathbf{z}_i$ from unity. We expand **r** in the basis of the eigenvectors of **A**,

$$\mathbf{r}=\sum_j^N r_j\mathbf{z}_j, \tag{A1}$$

where

$$r_j=\mathbf{z}_j^T\mathbf{r}, \tag{A2}$$

and, in particular,

$$r_i=1-\epsilon. \tag{A3}$$

Then, since **r** is considered to be normalized,

$$\mathbf{r}^T\mathbf{r}=1-2\epsilon+\sum_{j\neq i}^N r_j^2 \tag{A4}$$

or

$$2\epsilon=\sum_{j\neq i}^N r_j^2=R\sum_{j\neq i}^N \bar{r}_j^2, \tag{A5}$$

neglecting terms of order ϵ^2. In the final equation, the expansion coefficients are separated into a scale factor $R^{1/2}$, and the normalized coefficients $\bar{r}_j$. If **r** is normalized then R will be unity, since the $\mathbf{z}_j$ are orthonormal.

Another measure of the error is to compare λ_i, the eigenvalue corresponding to $\mathbf{z}_i$, and the diagonal matrix element of **A** with respect to **r**, which is written

$$\alpha=\mathbf{r}^T\mathbf{A}\mathbf{r}=\lambda_i(1-2\epsilon)+\sum_{j\neq i}^N \lambda_j r_j^2. \tag{A6}$$

The error in this approximate eigenvalue is

$$\Delta\lambda=\alpha-\lambda_i=R\sum_{j\neq i}^N \lambda_j\bar{r}_j^2-2\lambda_i\epsilon=2\epsilon\bar{\lambda}_{(i)}, \tag{A7}$$

where

$$\bar{\lambda}_{(i)}=\sum_{j\neq i}^N (\lambda_j-\lambda_i)\bar{r}_j^2\Big/\sum_{j\neq i}^N \bar{r}_j^2. \tag{A8}$$

In this way $\Delta\lambda$, the eigenvalue error, is seen to be proportional to ϵ, the eigenvector error, and also to the quantity $\bar{\lambda}_{(i)}$, which reflects the distribution of the extraneous components of **r**, i.e., its projection on other eigenvectors of A besides $\mathbf{z}_i$. More precisely, $\bar{\lambda}_{(i)}$ can be described as the weighted average of the difference between the eigenvalue of interest and the eigenvalues of those eigenvectors to which approximate eigenvector is still coupled.

The error measure β^2 used in the Lanczos refinement is related to the eigenvector and eigenvalue errors in a simple way. Starting with the definition of β^2, and again neglecting terms of order ϵ^2, we find that

$$\begin{aligned}\beta^2&=\|\mathbf{A}\mathbf{r}-\alpha\mathbf{r}\|^2\\&=\mathbf{r}^T\mathbf{A}^2\mathbf{r}-\alpha^2\\&=R\sum_{j\neq i}^N \lambda_j^2\bar{r}_j^2+\lambda_i^2(1-2\epsilon)-\alpha^2\\&=2\epsilon\bar{\lambda}_{(i)}^2\end{aligned} \tag{A9}$$

where

$$\bar{\lambda}_{(i)}^2=\sum_{j\neq i}^N (\lambda_j-\lambda_i)^2\bar{r}_j^2\Big/\sum_{j\neq i}^N \bar{r}_j^2. \tag{A10}$$

Thus, β^2 is proportional to the eigenvector error, ϵ, and to the quantity, $\bar{\lambda}_{(i)}^2$, which is another measure of the distribution of extraneous components in **r**; in this case, it is the weighted average of the squared differences between the eigenvalue of $\mathbf{z}_i$ and the eigenvalues of those other eigenvectors to which **r** is still coupled.

In the context of eigenvector refinement, β^2 is useful because it can be calculated without knowledge of the exact eigenvectors. However, its use as an estimator of ϵ and $\Delta\lambda$ is complicated by the uncertainty in the quantities $\bar{\lambda}_{(i)}$ and $\bar{\lambda}_{(i)}^2$, which will differ for each eigenvector and change throughout the course of the refinement. Nonetheless, experience suggests that these quantities do not change as dramatically as ϵ, which decreases by many orders of magnitude over the course of a refinement.

[1] R. A. Friesner, Annu. Rev. Phys. Chem. **42**, 341 (1991), and references therein.
[2] P. Pulay, Chem. Phys. Lett. **73**, 393 (1980).
[3] J. K. Cullum and R. A. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations* (Birkhaeuser, Boston, 1985), Vol. 1.
[4] A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry. An Introduction to Advanced Electronic Structure Theory* (McGraw–Hill, New York, 1989).
[5] G. Strang, *Linear Algebra and Its Applications* (Academic, New York, 1980).