PHYSICAL REVIEW B VOLUME 37, NUMBER 11 15 APRIL 1988-I

Diagonalization of large matrices in pseudopotential band-structure calculations: Dual-space formalism

José Luís Martins* and Marvin L. Cohen

Department of Physics, University of California, Berkeley, California 94720

and Materials and Chemical Science Division, Lawrence Berkeley Laboratory, Berkeley, California 94720

(Received 8 October 1987)

Combining iterative methods of calculating the eigenvectors of a Hermitian matrix with a matrix-multiplication technique using the fast-Fourier-transform algorithm, we present an efficient method of performing total-energy and band-structure calculations in crystals with the plane-wave local-pseudopotential formalism. The method can be viewed as a dual-space formalism where part of the calculations is performed in momentum space and another part in coordinate space. Significant savings in both computer time and memory are obtained. Results of calculations for molecular hydrogen with matrix sizes as large as 7200 are presented as an example.

INTRODUCTION

In the modern theory of the electronic structure of solids we must solve accurately a second-order partial differential equation in three-dimensional space, the Schrödinger equation. The existing schemes of solving this difficult numerical problem,[1] like the pseudopotential, orthogonalized-plane-wave, augmented-plane-wave, muffin-tin orbital, and linear combination of atomic orbitals methods and the methods derived from those, rely on the known properties of the solutions of the Schrödinger equation in the design of the computational procedure. For example the chemical concept of valency is reflected in the different treatments for the valence and core electrons, either by replacing the core electrons by a pseudopotential or by expanding the wave functions in different functional forms in the core and in the valence regions.

Most electronic structure methods expand the wave functions in a basis set, and therefore an important part of the numerical procedure is the diagonalization of Hermitian matrices. For complex crystal structures the matrix size can be very large and the matrix diagonalization may require extensive computing resources. Hence the search for new algorithms and methods of matrix diagonalization has been very active. As in the case of the electronic structure methods the most successful diagonalization procedures[2-4] are also those which exploit our knowledge of the properties of the solutions of the Schrödinger equation.

We implemented an efficient method to diagonalize the large matrices that occur in self-consistent local-pseudopotential band-structure calculations.[5] The method combines the iterative procedure of the direct inversion of the iterative subspace (DIIS) algorithm,[2] with the computation of the product of the Hamiltonian matrix by a wave vector through the use of fast Fourier transforms (FFT) implemented in the molecular-dynamics method of Car and Parrinello.[3] The resulting procedure is faster than traditional diagonalization methods (e.g., using the EISPACK subroutines) and since the Hamiltonian matrix is never explicitly calculated, it requires less computer memory. In practice the use of the FFT means that we calculate the kinetic-energy operator in momentum space and the potential-energy operator in coordinate space. Viewed in this perspective the momentum-space local-pseudopotential formalism[5] can be advantageously reformulated into a dual-space pseudopotential formalism. We will compare the results of calculations of the total energy of molecular hydrogen where matrices with sizes as large as 7200 were diagonalized with a VAX 11/785 minicomputer.

THEORY AND ALGORITHM

In the electronic structure theory of crystals we must solve the Schrödinger equation,

$$[-\tfrac{1}{2}\nabla^2+V(\mathbf{r})]\psi(\mathbf{r})=E\psi(\mathbf{r}) \ , \tag{1}$$

where $V(\mathbf{r})$ is an effective potential with the periodicity of the crystal lattice (e.g., a screened pseudopotential). Here we will use atomic units $e=\hbar=m_e=1$. In the plane-wave formalism[1,5] the eigenfunction is expanded in a sum of plane waves with crystal momentum $\mathbf{k}$,

$$\psi(\mathbf{r})=\sum_j a(\mathbf{G}_j)e^{i(\mathbf{k}+\mathbf{G}_j)\cdot\mathbf{r}}=a(\mathbf{r})e^{i\mathbf{k}\cdot\mathbf{r}} \ ,$$

where the sum is over all the reciprocal lattice vectors $\mathbf{G}_j$ and $a(\mathbf{r})$ has the periodicity of the crystal (Bloch's theorem). In the plane-wave basis the Schrödinger differential equation [Eq. (1)] is transformed into a matrix eigenvalue problem,

$$\sum_j H_{ij}a(\mathbf{G}_j)=Ea(\mathbf{G}_i) \ , \tag{2}$$

where the (**k**-dependent) Hermitian Hamiltonian matrix is

$$H_{ij}=\delta_{ij}\tfrac{1}{2}(\mathbf{k}+\mathbf{G}_i)^2+V(\mathbf{G}_i-\mathbf{G}_j) \ , \tag{3a}$$

and the Fourier transform of the potential is

$$V(\mathbf{G})=\int d^3r\, e^{i\mathbf{G}\cdot\mathbf{r}}V(\mathbf{r})\ . \tag{3b}$$

According to the physicists' tradition we will be using the same symbol for a function, its Fourier transform, the corresponding operator and its matrix representation, and use the arguments and context to distinguish between the different mathematical representations of the same physical object.

For practical reasons the order of the reciprocal-lattice vectors is chosen such that $|\mathbf{k}+\mathbf{G}_i|\geq|\mathbf{k}+\mathbf{G}_j|$ if $i\geq j$, and the expansion is truncated at a size N determined by the convergence to the desired accuracy of the physical property (e.g., total energy) being studied. The condition $|\mathbf{k}+\mathbf{G}_i|\leq G_{\max}$ is often used to define the truncation, and the calculation is said to include plane waves up to an energy cutoff of $E_{\max}=\frac{1}{2}G_{\max}^2$.

The solution of the eigenvalue problem [Eq. (2)] with general-purpose subroutines (e.g., using the EISPACK library) requires a number of floating-point operations of the order to N^3 and a computer storage of N^2 words, and is unpractical for problems where N is larger than a few hundred. More efficient algorithms can be developed if we use the known properties of the Hamiltonian matrix [Eq. (3a)]. The algorithm need not work for an arbitrary matrix, but should be fast and accurate for the specific class of matrices it is designed to diagonalize. It should detect its own eventual failure to achieve the diagonalization when applied to a matrix outside that class.

In band-structure problems we only need the lowest eigenvalues of the Hamiltonian matrix, therefore iterative methods should be considered as an alternative. The simplest iterative method is the Lanczos method,[6] which is based on the convergence properties of the series

$$x^{(m+1)}=Hx^{(m)}\ .$$

This series converges to the eigenvector y of H with the largest eigenvalue in absolute value as long as $x^{(0)}$ and y are not orthogonal. The convergence is geometric with a convergence factor given by the ratio of the second largest to the largest eigenvalue in absolute value. This iteration defines a basis set $\{x^{(0)},x^{(1)},\ldots,x^{(l)}\}$ of an iterative subspace, the Krylov subspace.[6] If we define in this subspace the matrices $P_{ij}=x^{(i)T}Hx^{(j)}$ and $Q_{ij}=x^{(i)T}x^{(j)}$, then the eigenvalues of the equation $Pu=\lambda Qu$ converge to the lowest and largest eigenvalues of H with increasing dimension of the subspace. It is easily recognized that we have just performed a change of basis in which we calculate the Hamiltonian matrix elements. The series generated by the Lanczos procedure has an interesting physical interpretation in the recursive tight binding method.[7] We also notice that the matrices P and Q can be tridiagonalized with a trivial transformation.[4,6] In the numerical implementations of the Lanczos procedure instabilities related to the appearance of linear dependences in the iterative subspace can occur. Practical ways to avoid this problem are discussed in the literature.[6]

The properties of the Lanczos method provides the mathematical background to most of the modern iterative methods. The inverse iteration method[6,8] is defined by the modified series

$$x^{(m+1)}=(H-\mu)^{-1}x^{(m)}\ ,$$

and converges to the eigenvector y of H with the eigenvalue nearest to μ as long as $x^{(0)}$ and y are not orthogonal. The convergence properties of the inverse iteration are trivially derived from the convergence properties of the Lanczos series when we recognize that the eigenvector with eigenvalue E of H is an eigenvector of $(H-\mu)^{-1}$ with eigenvalue $(E-\mu)^{-1}$. The convergence will be very fast if μ is a good guess for the eigenvalue E since $(E-\mu)^{-1}$ will have a very large absolute value. The inverse iteration procedure requires a matrix inversion, exactly the operation we wanted to avoid in the first place, but its convergence properties can be used as a guide for methods where the iteration is performed with an *approximate* inverse matrix as is the case of the DIIS procedure.

The most important property of the matrix obtained from a plane-wave expansion is that if it is truncated at a smaller value of $G_{\max}$ its eigensolutions are still a reasonable approximation of the original Schrödinger equation. This is reflected in the matrix elements by the fact that the off-diagonal values given by the potential operator V are bounded, (the maximum of $|V(\mathbf{G}_i)|$ occurs for one of the smaller $\mathbf{G}_i$), whereas the diagonal elements given by the kinetic-energy operator [$V(\mathbf{0})$ adds a constant to the diagonal elements and can be ignored] are unbound and increase with row index i with the power $\frac{2}{3}$. Therefore for a sufficiently large value of i the matrix is diagonally dominant.

Matrices that are diagonally dominant have been the traditional candidates for the Jacobi and Gauss-Seidel iterative relaxation methods.[8] First we rewrite the eigenvalue problem as

$$x=D^{-1}(H-E)x+x\ ,$$

where D is a nonsingular matrix, and then we define the iteration procedure

$$y^{(m+1)}=D^{-1}(H-\lambda^{(m)})x^{(m)}+x^{(m)}\ , \tag{4a}$$

$$x^{(m+1)}=y^{(m+1)}\ , \tag{4b}$$

$$\lambda^{(m)}=\frac{x^{(m)T}Hx^{(m)}}{x^{(m)T}x^{(m)}}\ , \tag{4c}$$

with $x^{(0)}$ an arbitrary guess vector. If we define the residual vector $R^{(m)}=(H-\lambda^{(m)})x^{(m)}$, and the correction vectors $\delta x^{(m+1)}=y^{(m+1)}-x^{(m)}$, then Eq. (4a) is just a linear equation,

$$D\delta x^{(m+1)}=R^{(m)}\ . \tag{5}$$

The matrix D should be chosen such that the linear equation [Eq. (5)] is easier to solve than the original eigenvalue problem [Eq. (3)], and such that the iteration converges rapidly to an eigensolution of H. If we choose D equal to the diagonal part of H then we have the Jacobi relaxation method,[8] if we choose D to be the lower (or upper) triangular part of H then we obtain the Gauss-Seidel relaxation method.[8]

In the DIIS method[2] the relaxation step (called the

Newton step in Ref. 2) is obtained with the choice of the diagonal elements of H augmented with the off-diagonal elements H_{ij} such that $i,j \le M$ and $M \ll N$. It requires a diagonalization of a matrix of size M and converges rapidly if this matrix gives a coarse solution to the original Schrödinger equation. The power of the DIIS algorithm is in the use of a convergence accelerator replacing Eq. (4b),

$$x^{(m+1)}=c_0x^{(0)}+\sum_j c_j\delta x^{(j)} ,$$

where the coefficients of the expansion are chosen such that they minimize the residual $R^{(m+1)}$. These coefficients can be obtained by diagonalizing the Hamiltonian in the iterative subspace $\{x^{(0)},\delta x^{(1)},\dots,\delta x^{(l)}\}$, an easy operation given the fact that $l \lesssim 10$. The time consuming steps in the DIIS method are the following: (i) the calculation of the products $H\delta x^{(j)}$ needed for both the calculation of the Hamiltonian matrix in the iterative subspace and for the calculation of the residuals, requiring $O(N^2)$ floating-point operations, and (ii) the diagonalization of a small matrix [Eq. (5)], requiring $O(M^3)$ floating-point operations. The method requires the storage of the N^2+N elements of the Hamiltonian matrix in the computer memory and may access each of them several times.

The efficiency of the DIIS algorithm that we just described briefly can be improved in local-pseudopotential calculations through the use of FFT to calculate the product of the Hamiltonian by a wave vector.[3] This matrix multiplication requires only $O(N\log_2 N)$ floating-point operations, and the storage of a real array of size $O(N)$. The small matrix diagonalization [Eq. (5)] still requires $O(M^3)$ floating-point operations but only the M^2+N matrix elements of D need to be stored. Inspection of Eqs. (1) and (2) show that while the action of the kinetic-energy operator is easy to calculate in momentum (reciprocal or **G** space) the action of the potential-energy operator is easy to calculate in the coordinate space (direct or **r** space). We will calculate the effect of each of these operators in the "easy" space and Fourier-transform the result to the space where we want to do our "bookkeeping." In our case the bookkeeping of the potential and the wave vectors is done in the momentum space where the calculation of the effect of the kinetic-energy operator is trivial. To obtain the action of the potential operator on a wave vector, we Fourier-transform both to coordinate space obtaining the periodic functions $a(\mathbf{r})=e^{-i\mathbf{k}\cdot\mathbf{r}}\psi(\mathbf{r})$ and $V(\mathbf{r})$, in a uniform grid. Then $Va(\mathbf{r})$ is obtained in the same grid by trivial multiplication, and an inverse Fourier transform is used to transform the result back into momentum space. This operation can also be described as the calculation of the convolution appearing in Eq. (2) through Fourier-transform methods. This procedure has three very interesting properties: (i) with a judicious choice of the Fourier-transform procedure it is an exact algorithm (in practice this means that the results obtained through the two Fourier transformations are identical, within computer roundoff accuracy, to the straightforward calculation of the matrix product); (ii) it is asymptotically faster than the direct matrix multiplication; and (iii) last but not least, since the Hamiltonian matrix elements H_{ij} are never explicitly calculated with the double Fourier-transform procedure, they do not need to be stored in the computer memory.

Both the potential $V(\mathbf{r})$ and the function $a(\mathbf{r})$ are periodic functions in coordinate space represented in the computer by their Fourier components at the reciprocal lattice vectors, $V(\mathbf{G}_i)$ and $a(\mathbf{G}_i)$. The expansion of ψ is truncated at $G_{\max}$, and inspection of Eq. (2) shows that the expansion of V can be truncated at $2G_{\max}$. The result of the convolution will also be truncated at $G_{\max}$. All these functions are therefore bandwidth limited in (spatial) frequency. Before using the discrete Fourier-transform methods[9] (DFT) we must define periodic functions in momentum space by convoluting with the periodic function

$$S_{N_1N_2N_3}(G)=\sum_{m_1,m_2,m_3}\delta(\mathbf{G}-(m_1N_1\mathbf{b}_1+m_2N_2\mathbf{b}_2+m_3N_3\mathbf{b}_3)) ,$$

where $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ are the reciprocal-lattice basis vectors. We can calculate the three dimensional $N_1\times N_2\times N_3$ DFT of the periodically extended functions $\psi'=S*\psi$ and $V'=S*V$ obtaining the values of V' and a' in a $N_1\times N_2\times N_3$ grid in coordinate space. In the grid points $\mathbf{r}_{n_1n_2n_3}=n_1\mathbf{t}_1/N_1+n_2\mathbf{t}_2/N_2+n_3\mathbf{t}_3/N_3$, where $\mathbf{t}_1$, $\mathbf{t}_2$, and $\mathbf{t}_3$ are the primitive translation vectors of the crystal ($\mathbf{b}_i\cdot\mathbf{t}_j=2\pi\delta_{ij}$), the unprimed function are identical to the primed functions, $a(\mathbf{r}_{n_1n_2n_3})=a'(\mathbf{r}_{n_1n_2n_3})$ and $V(\mathbf{r}_{n_1n_2n_3})=V'(\mathbf{r}_{n_1n_2n_3})$, and therefore their product is also identical. The application of the inverse DFT gives the periodic function $V'*a'$ from which we want to extract the values of $Va(\mathbf{G}_i)$. If we define G_{DFT} as the radius of the largest sphere that fits inside the rhombus with edges $N_1\mathbf{b}_1$, $N_2\mathbf{b}_2$, and $N_3\mathbf{b}_3$, then recovering the values of $Va(\mathbf{G}_i)$ for $|\mathbf{G}_i| \le G_{\max}$ is only a well-defined operation if $G_{\max}\le G_{\mathrm{DFT}}$. Moreover, since $V'*a'$ has a bandwidth of $3G_{\max}$ for all the reciprocal-lattice vectors satisfying $|\mathbf{G}_i| < 2G_{\mathrm{DFT}}-3G_{\max}$, we have that $V'*\psi'(\mathbf{G}_i)=V\psi(\mathbf{G}_i)$, whereas for all the other reciprocal-lattice vectors the equality is only approximately satisfied. Therefore if we want to obtain the exact product we must choose $G_{\mathrm{DFT}}\ge 2G_{\max}$. The choice $G_{\mathrm{DFT}}\ge G_{\max}$ is, however, physically sound as we will discuss later.

A comparison of the computing times between the straight matrix multiplication and the FFT method depends on the computer and on the algorithms used and how they access the memory or vectorize, but an estimation of the number of multiplications needed for each method will be a good guide of their respective performance. The matrix multiplication requires N^2 multiplications whereas the two three-dimensional Fourier transforms (the Fourier-transform of V needs to be done only once) require $6N'\log_2 N'$ multiplications[9] where $N'=N_1N_2N_3$. For the exact multiplication N' is larger than the matrix size N due to the "wasted space." The "padding" of the momentum space gives a factor of 2^3 which is to be multiplied by the wasted space when we inscribe a sphere inside a rhombus. For the case of a cube this last factor is $6/\pi \simeq 2$. Getting all the factors together we obtain the critical matrix size N_c where the FFT method becomes more efficient than straightforward matrix multiplication, $100 N_c \log_2 N_c \simeq N_c^2$, giving a value of $N_c \simeq 700$.

DUAL-SPACE FORMALISM

The procedure of evaluating the kinetic energy in momentum space and the potential energy in real space can be thought of as a dual-space formalism for pseudopotential calculations in contrast with the momentum-space formalism[5] where all the calculations are carried on in momentum space. In this context the truncation of the expansion of the wave function at $G_{\max}$ and of the potential at $2G_{\max}$, is equivalent to neglecting details of the wave function with linear size smaller than $2\pi/G_{\max}$, and details of the potential with linear size smaller than $\pi/G_{\max}$. It is reasonable to assume that we do not need more details in the description of the potential than in the description of the wave functions and that we should calculate everything with the same kind of detail, that is truncate all expansions in plane waves at $G_{\max}$. Consistently with this requirement, the choice $G_{\mathrm{DFT}} \geq G_{\max}$ represents the functions in coordinate space with the same level of detail. In the dual-space formalism we calculate in coordinate space the product of the potential-energy operator by an eigenvector, the modulo square of the wave function $|\psi|^2$, and the exchange and correlation potential, while we calculate in momentum space the effect of the kinetic-energy operator, the total electronic charge density, and the Hartree potential. The bookkeeping is still done in the momentum space. With this choice of $G_{\mathrm{DFT}} \geq G_{\max}$ the break-even point for the dual-space formalism is given by $12 N_c \log_2 N_c \simeq N_c^2$, that is $N_c \simeq 100$.

In contrast with the choice of $G_{\mathrm{DFT}} \geq 2G_{\max}$, the results of the calculation with $G_{\mathrm{DFT}} \geq G_{\max}$ are not identical within computer accuracy to the results of the traditional momentum-space calculation, but are a different approximate solution of the original Schrödinger equation with an accuracy determined by the choice of $G_{\max}$. The momentum-space formalism and the dual-space formalism results will both converge to the exact solution with increasing $G_{\max}$ with similar rates of convergence.

TEST FOR MOLECULAR HYDROGEN

The method was tested in calculations for molecular hydrogen with the $Pa3$ structure (8 atoms per unit cell) with a cutoff energy of $E_{\max} = 36$ Ry or $E_{\max} = 64$ Ry, and for several lattice constants a. The total energy was calculated with a single special k point, and the eight lowest-energy eigenvectors were calculated at this k point. It took four iterations to achieve self-consistency, and the size M of the small matrix of the DIIS algorithm was determined by an energy cutoff of 7 Ry.

The total computing times for five different combinations of lattice constant and energy cutoff giving different matrix sizes N are presented in Table I for four different diagonalization procedures: EISPACK subroutines, DIIS method, DIIS method plus the matrix multiplication using the FFT ($G_{\mathrm{DFT}} \geq 2G_{\max}$), and the dual-space method ($G_{\mathrm{DFT}} \geq G_{\max}$). Our estimates of the critical matrix sized of $N_c \simeq 700$ for the DIIS plus FFT method and $N_c \simeq 100$ for the dual-space method are verified in practice. We notice that the relative performance of the different method may depend on the computer being used and on the efficiency of the FFT algorithm.

The savings in computing time are evident from Table I. For the calculation with a matrix size of 7200 where the EISPACK routine would take more than a year,[10] and the DIIS method would take three days,[10] the dual-space method takes only 2 h. The storage of a 7200 by 7200 matrix requires 415 Mbytes of computer memory while the storage space required for *all* the variables in our calculation with the dual-space formalism is less than 10 Mbytes. It is also apparent from Table I that for very

TABLE I. The performance of the different methods described in the text is compared for several calculations with different matrix sizes N, for the test case of $Pa3$ hydrogen (8 atoms per unit cell) with lattice constant a (in Bohr radii) and energy cutoff $E_{\max}$ (in Ry). The computing time is given in minutes of VAX 11/785 CPU times for the full self-consistent total-energy calculation. The percentage of CPU time spent in the EISPACK subroutines and in the iterative diagonalization subroutines is given in parentheses below the computing times.

	Computing time in minutes					
N	EISPACK	DIIS	DIIS + FFT	Dual	a	$E_{\max}$
305	35	6.2	8.2	1.6	4.35	36.0
	(94,0)	(1,62)	(1,75)	(4,66)		
597		32	30	7.9	5.5	36.0
		(2,71)	(2,80)	(8,74)		
2968			98	39	7.0	64.0
			(6,71)	(16,68)		
3031			167	104	9.4	36.0
			(40,45)	(64,30)		
7200				126	9.4	64.0
				(53,38)		

large matrices the computing times starts depending strongly on the small matrix size M. Increasing N by increasing the energy cutoff does not change the small matrix size M and causes only a moderate increase in computer time. Increasing N by increasing the unit cell size implies also an increase in the small matrix size M and may increase significantly the total computing time. The diagonalization of the small matrix is thus the factor that will ultimately limit the application of the method to more complicated systems.

CONCLUSIONS

The combination of the iterative diagonalization DIIS algorithm with a matrix multiplication using FFT techniques allowed us to perform total-energy calculations in the local-pseudopotential plane-wave formalism for very large systems. The method saves both computer time and memory for problems with Hamiltonian matrix sizes larger than a few hundred and can be used for problems with matrix sizes of several thousand.

ACKNOWLEDGMENTS

One of the authors (J.L.M.) acknowledges the support of the Miller Institute for Basic Research in Science. We thank Dr. R. Car and Dr. D. M. Wood for the very helpful discussions about the molecular-dynamics and the DIIS methods. We thank Dr. S. Froyen for helpful discussions about fast diagonalization methods and for pointing out some of the properties of the FFT. This work was supported by National Science Foundation Grant No. DMR8319024 and by the Director, Office of Energy Research, Office of Basic Energy Sciences, Materials Science Division of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

---

*Present address: Department of Chemical Engineering and Materials Science, University of Minnesota, Minneapolis, Minnesota 55455.

[1] V. Heine, in *Solid State Physics*, edited by H. Ehrenreich, F. Seitz, and D. Turnbull (Academic, New York, 1970), Vol. 24, p. 1; J. M. Ziman, in *Solid State Physics*, edited by H. Ehrenreich, F. Seitz, and D. Turnbull (Academic, New York, 1971), Vol. 26, p. 1.

[2] D. M. Wood and A. Zunger, J. Phys. A **18**, 1343 (1985).

[3] R. Car and M. Parrinello, Phys. Rev. Lett. **55**, 2471 (1985).

[4] K. Maschke, Philos. Mag. B **51**, 397 (1985).

[5] J. Ihm, A. Zunger, and M. L. Cohen, J. Phys. C **12**, 4409 (1979).

[6] B. N. Parlett, *The Symmetrical Eigenvalue Problem* (Prentice-Hall, Englewood Cliffs, 1980); C. C. Paige, in *Large Scale Matrix Problems*, edited by A. Björk, R. J. Plemmons, and H. Schneider (North-Holland, Amsterdam, 1981).

[7] R. Haydock, in *Solid State Phyics*, edited by H. Ehrenreich, F. Seitz, and D. Turnbull (Academic, New York, 1980), Vol. 35, p. 216.

[8] W. H. Press, B. P. Flannery, S. T. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1986).

[9] For a discussion of the discrete Fourier transform theory and the fast Fourier transform algorithms, see Ref. 8 and references therein.

[10] Values extrapolated using an N^3 scaling for EISPACK and an N^2 scaling for DIIS.