Pseudospectral Hartree–Fock calculations on glycine

Murco N. Ringnalda, Youngdo Won, and Richard A. Friesner
Department of Chemistry, University of Texas at Austin, Austin, Texas 78712

(Received 14 April 1989; accepted 20 September 1989)

The pseudospectral method for Hartree–Fock calculations is applied to the glycine molecule, a test case with 100 basis functions. Several algorithmic improvements are reported, including a Newton–Raphson convergence scheme, Fock matrix updating, a multigrid technique, and optional recalculation of integrals. The pseudospectral method is shown to accurately reproduce the Roothaan–Hall relative and total energies for three conformations of glycine. Timing results show the pseudospectral code to be substantially faster than conventional Hartree–Fock codes.

I. INTRODUCTION

In a series of previous papers,[1–4] we have developed a new algorithm for solving the self-consistent equations of electronic structure theory. The algorithm is a synthesis of conventional quantum chemical techniques with a modified version of the pseudospectral method, a numerical approach widely used in hydrodynamic simulations[5] and, more recently, a variety of chemical physics applications.[6] Both a numerical grid representation of the molecular orbitals and a standard quantum chemical basis set are employed in the calculation, which is designed to yield the Roothaan–Hall (RH) result[7] for that basis set. The basic advantages of the method are N^3 (rather than N^4) scaling with basis set size and elimination of the explicit evaluation of two electron integrals, thus in principle providing order of magnitude reductions in central processing unit (CPU) and input/output (I/O) time for *ab initio* electronic structure calculations on large molecules.

In Ref. 4, a key problem, that of automatic grid generation, was solved in a satisfactory manner. Systematic convergence of spectroscopic constants and the total energy to the corresponding RH values was demonstrated for Hartree–Fock (HF) calculations on the water molecule. Most critically, the grid generation algorithm is easily applicable to an arbitrary molecule.

In this paper, we apply the method to a medium sized organic molecule, glycine. A 6-31G** basis set[8] is used, so there are 100 basis functions in all. Thus, the calculation is moderately demanding computationally by the standards of current *ab initio* technology. By contrast, a three-dimensional molecule of such size and complexity is far beyond the range of any alternative numerical approaches to solving the HF equations, at least as judged by the published literature.

This paper has a number of objectives. First, we introduce a great deal of additional numerical technology vital to efficient implementation of a pseudospectral HF algorithm for large molecules. This includes multigrid capabilities, recalculation of integrals (avoiding any disk storage demands that scale by more than N^2), an N^3 Newton–Raphson algorithm, and various numerical improvements in all aspects of the computations. The process of optimizing the approach along these lines is at present in its infancy; the substantial gains achieved here are encouraging with regard to the ultimately achievable efficiency.

Secondly, we demonstrate that the method can produce results which are for all practical purposes equivalent to those of conventional HF codes. That is, the difference in total energy between the pseudospectral and RH results at any geometry is less than 0.033 kcal/mol; eigenvectors and eigenvalues are also extremely close to RH values. A subsequent paper will present an analytic gradient formulation which indicates that adequate accuracy can also be obtained for local energy differences.

The specific test chosen is computation of three points on the torsional potential energy surface obtained by rotation around the C–C bond of glycine. This is a severe test for a numerical method, because it involves significantly changing the integration grid for conformations which differ by a small amount of energy. The success and stability of the calculations for glycine, which has several nearly degenerate orbitals whose energy gaps change significantly as a function of torsional angle, suggests that the method will work in most cases of interest. This, of course, needs to be confirmed by calculations for a large number of molecules.

Finally, we present an overall timing comparison between our method and several available HF programs, for the Cray X-MP. Such comparisons can be misleading, partly because there are tradeoffs in CPU and I/O which are machine dependent, or because a particular program has been highly optimized for a specific machine, or because one algorithm is in a much more mature phase of development than another. Nevertheless, it is a useful exercise at this point to make some sort of comparison of the pseudospectral algorithm with its main competitor. In fact, we achieve significant reductions in CPU time even in comparison to the program GRADSCF, which has been specifically optimized for Cray computers.

The algorithms described here also are relevant to extension of the method to more complex electronic structure calculations. For example, the Newton–Raphson scheme utilizes an iterative method which, with minimal modification, is directly applicable to coupled-perturbed HF calculations (e.g., to obtain second derivatives), or to certain types of multiconfiguration self-consistent field (MCSCF) or configuration interaction calculations.

0021-9606/90/021163-11$03.00

II. NUMERICAL METHODS

A. Overview

The formal equations used in our pseudospectral formulation of the HF approximation have already been given elsewhere[1–4] and we will not repeat this here. Rather, we will in this subsection outline the principal new features that have been added to our code and then describe these in detail in the following subsections.

The program can be divided into routines which perform the following major tasks:

(i) Definition of the molecular geometry and basis set. This is no different from analogous routines in a conventional HF code.

(ii) Generation of the molecular grid. This is essentially identical in form to what was presented in Ref. 4, although significant changes in the actual grids employed have been made.

(iii) Generation of the kinetic energy, overlap, nuclear attraction, and three-center, one-electron potential integrals. The numerical details have been improved over the procedures described in Refs. 1–3, but the basic elements of the calculation have not been altered. We have not yet implemented effective integral cutoffs; this is an important aspect of future work.

(iv) Assembly of the least-squares fitting matrix and construction of the quadrature scheme via solution of the normal equations. Substantial modifications have been made in the numerical methods here, as well as in the set of dealiasing functions employed to reduce discrepancies with the RH results. There is, in addition, considerable room for improvement of our current implementation. This will be discussed in some detail later on.

(v) Construction of the Coulomb and exchange operators, and calculation of the Fock matrix elements. In addition to some optimization of the numerics, we have implemented a Fock matrix updating scheme, along the lines of Almlöf and co-workers.[9] This is quite effective when used in conjunction with a multigrid strategy; updates which involve small changes in the Fock matrix are carried out on relatively sparse grids, because highly accurate integrals are not then required. The incorporation of integral cutoffs will further enhance the utility of this procedure.

We remind the reader here of two essential and nontrivial features of the algorithm, discussed in previous papers. First, purely atomic two electron integrals are done analytically; interfacing this with the rest of the pseudospectral code is somewhat complicated, although the actual computational requirements are negligible in terms of CPU time. A future publication will report the results of additionally treating two-center terms analytically. Secondly, the Fock matrix must be symmetrized. We do this here simply by averaging the off-diagonal elements. A better procedure can probably be developed, although we have not yet found one. The more complex methods used in previous versions of the program are more expensive and have not proven to be clearly superior (although our tests were somewhat limited).

(vi) SCF iterations. We have developed an iterative pseudospectral Newton–Raphson procedure which scales like N^2Mk, where M is the size of the grid, and k is the number of iterations required to converge the unitary transformation matrix. Furthermore, we utilize an approximate Hessian constructed on a very coarse grid, so that iterations are relatively inexpensive. The net result is accelerated (although not quadratic) convergence of the SCF procedure at little cost. Note that this technique is readily applicable to MCSCF calculations.

(vii) Integral input/output. This is already a substantial consideration for glycine and of course becomes much more critical as the molecule increases in size. We have developed both a conventional version of our code (in which the three-center, one-electron integrals are stored on a disk) and a "direct" version in which some or all integrals are recalculated at each self-consistent step. For the latter, the relatively low expense of integral generation as compared to conventional programs leads to very large CPU savings. The maximum required storage in this version is in fact proportional to N^2.

The most significant of the above improvements will now be presented in more detail.

B. Assembly of the least-squares matrix

To solve the normal equations, one must construct the matrix

$$\mathbf{C} = \mathbf{R}^\dagger \mathbf{w} \mathbf{R}, \tag{1}$$

where **R** contains the basis and dealiasing functions evaluated at the gridpoints and **w** is a weight matrix produced by the grid generation algorithm. Straightforward matrix multiplication requires J^2M operations, where J is the total number of basis plus dealiasing functions and M is the number of grid points. In order to achieve high accuracy for glycine, we have had to increase both of these parameters above their (proportionate) values for the water molecule. Thus, this has become a potentially expensive part of the calculation (for the largest glycine grid used, J^2M is $\sim 6 \times 10^9$).

We have implemented a major improvement of this simple approach, which saves substantial amounts of CPU time. First, the dealiasing and basis functions are grouped into blocks of s, p, d, and f functions; each block has the same radial exponent and atomic center. The piece of the **C** matrix corresponding to all of the products of the functions in two blocks is then computed by the following algorithm:

(i) Multiply the radial parts together.

(ii) Integrate on the grid the radial product with all possible angular terms (centered at the product center) arising from combining the two blocks. This is a smaller number of functions than is obtained by taking all possible pairs of block functions. For example, for two *spd* blocks, there are 35 angular functions and 100 product pairs. One can then transform the angular functions back to the original polynomial products, at a trivial cost.

Many products extend spatially over only a relatively small part of the molecule and thus need only to be evaluated on a part of the molecular grid. We have developed a cutoff scheme for the above calculation which is compatible with the Cray X-MP architecture, in the sense that it preserves vectorization while eliminating unnecessary operations. The cutoff algorithm involves vectorization over function, grid, and angular blocks, and discarding of grid blocks which estimates indicate can be safely neglected.

The entire procedure reduces the CPU time for the assembly of C by more than a factor of 3 for the finest glycine grid used. Further improvements are available, possibly in conjunction with modification of the dealiasing functions.

C. Multigrid strategy

By multigrid, we mean here the use of several different grids in the SCF procedure. At present, we employ four separate grids, which we refer to as ultrafine, fine, medium, and coarse. These grids have been developed by empirical optimization; their precise characteristics are specified in Appendix A. The optimization depends upon the basis set, in this case 6-31G**. We have constructed atomic grids for hydrogen and the remaining first row atoms; the synthesis into the molecular grid is carried out using the completely automatic algorithm described in Ref. 4. The hope is that the generator atomic grids are transferable to other molecules and are adequate for all molecular geometries, not just the few studied here. This will obviously need to be demonstrated via a large number of molecular calculations. An encouraging feature is that the accuracy does not depend sensitively upon the precise structure of any grid; small changes produce small alterations in the total energy.

Each of the four grids is generated separately and must be accompanied by its own least-squares integration scheme and set of potential integrals. Fortunately, the expense of constructing these for the three smaller grids is relatively low.

The coarse grid, 35 points per atom (348 total points), is used only for the iterations of the unitary transformation matrix in the Newton–Raphson procedure. It is not very accurate and indeed fails to converge to a reasonable total energy when run by itself. However, it performs adequately in the Newton–Raphson scheme, in that replacement by the medium grid does not accelerate convergence sufficiently to warrant the expense. A rationalization of this behavior is not trivial and will not be undertaken here.

The medium grid, 1420 total points (142 per atom; recall that these are in fact unevenly distributed over the atoms because Voronai truncation is quite different for atoms on the interior of the molecule as compared to those on the exterior), produces a total energy accurate to roughly 0.02 hartrees when run by itself. If this is sufficient for the desired purpose (e.g., electronic excitation energies), then the expense of the calculation can be reduced by eliminating the fine and ultrafine grids. For many applications, this is clearly not adequate accuracy. In this case, the medium grid is used for both initial iterations and for Fock matrix updating. Indeed, we have found it necessary to do only one iteration using the fine grid and one using the ultrafine grid. This is extraordinarily advantageous for a direct SCF procedure, as it means that one need only recalculate coarse and medium grid integrals.

The fine grid, which has an average of 275 points per atom (2757 total), is used only for the first Fock matrix updating step, at which point the corrections to the Fock matrix are still too large to be handled by the medium grid. If the ultrafine grid is not used, the fine grid gives a total energy for glycine which differs from the RH result by about 0.005 hartrees.

The ultrafine grid, 8991 total grid points (an average of 900 per atom; the range is 626–1023), achieves accuracies (as compared to the RH results) on the order of 0.000 05 hartrees, or 0.03 kcal/mol. This is better than the best correlated calculations can do in comparison with experiment and hence constitutes an entirely adequate level of accuracy. Evaluation of spectroscopic constants of course requires a higher level of relative precision; however, this can be accomplished via an analytic gradient approach, as will be shown in a forthcoming paper. The demonstration that these accuracies are generic awaits a more extended list of examples. There is, on the other hand, nothing particularly easy about glycine; it is hard to see why the method would fail in other cases, given that it works well here and is insensitive to small adjustments in the grid.

D. The Newton–Raphson procedure

It has generally been assumed that Newton–Raphson procedures are too costly to be of use in large molecule, closed shell HF calculations; instead, alternate extrapolation procedures have been implemented.[10] It is possible that these procedures, when applied to our method, might also lead to accelerated convergence, although the use of the multigrid iteration scheme makes extrapolation from previous iterations on a different grid of questionable value. We developed the Newton–Raphson approach for two reasons: (i) it produces a very competitive acceleration for us at a minimal computational cost; and (ii) it is a prototype for numerous correlated or coupled-perturbed HF calculations. Indeed, it is clear from the results shown below that the pseudospectral method can profitably be applied to all equations of this form. Subsequent publications will demonstrate this explicitly.

We employ here the formalism of Levy and coworkers,[11] which we believe is a particularly clear description of the required mathematics for the HF and certain [e.g., generalized valence bond (GVB)] types of MCSCF equations (note that this formalism is entirely equivalent to methods proposed by other workers[12]). Briefly, the new set of occupied orbitals is derived from the old via the unitary transformation

$$\mathbf{c}_{\text{new}} = \exp(\mathbf{X})\cdot\mathbf{c}_{\text{old}}, \tag{2}$$

where X is an antisymmetric matrix, i.e., $X_{ij} = -X_{ji}$. It is easily shown that this restriction implies that exp(X) is unitary; thus, Lagrange multipliers are not needed to preserve the orthonormality of the occupied orbitals. Note that formally X acts in the current molecular orbital space and the elements are labeled by the molecular orbital indices.

The unique elements of the $N \times N$ (N is the basis set size) matrix X are placed in a vector x, which is the unknown to be determined by minimizing the total energy with respect to each component. There are $L = N_{\text{occupied}} * N_{\text{virtual}}$ components, setting the mixing inside of the occupied and virtual spaces to zero (as proven in Ref. 11, the last assumption yields a quadratically convergent procedure). The equation for **x** is simply the Newton–Raphson equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{3}$$

where **A** is the matrix of second derivatives, i.e.,

$$\mathbf{A}_{ijkl} = \left(\frac{\delta}{\delta x_{ij}} - \frac{\delta}{\delta x_{ji}}\right)\left(\frac{\delta}{\delta x_{kl}} - \frac{\delta}{\delta x_{lk}}\right) E(\mathbf{X}) \tag{4}$$

and b contains the first derivatives

$$\mathbf{b}_{ij} = \left(\frac{\delta}{\delta x_{ij}} - \frac{\delta}{\delta x_{ji}}\right) E(\mathbf{X}). \tag{5}$$

A is thus an L by L matrix, while **b** is a vector of length L. The formal evaluation of the elements of **A** and **b** is straightforward and is given in detail in Ref. 11. The important points for our purposes are as follows: first, **b** simply contains Fock matrix elements, which we already have an algorithm for evaluating. The difficulties therefore consist of producing **A** and solving Eq. (3).

To actually assemble **A** directly (as is done in some MCSCF codes) is extremely expensive (in both storage and CPU time) for a large system. Furthermore, it would completely negate the advantages of the pseudospectral method, as each two-electron integral would have to be computed explicitly.

Fortunately, if we employ an iterative algorithm, it is not necessary to ever form the matrix elements of **A**. Instead, we simply need an algorithm for acting with **A** on **x**. In Appendix B, we show how such an algorithm can be constructed. The algorithm is in fact isomorphic to that required to generate a Fock matrix given an initial density matrix, with the "old" **x** playing the role of the density matrix and the "new" **x** appearing as the consequent Fock matrix. No transformation of the integrals to molecular orbital space is required. The algorithm is carried out in the atomic orbital basis and requires $\sim N^2M$ operations per iteration, just as Fock matrix assembly does. We make the procedure cost efficient by using the coarse grid to represent **A** (more accurate grids must, of course, be used to construct **b**). The tradeoffs involved are easily worked out for the present (HF) level of calculation.

For MCSCF calculations, the issues involved can be quite different; e.g., in some cases one has difficulty obtaining convergence at all unless some sort of Newton–Raphson procedure is used. Thus, the demonstration here of an algorithm that is valuable even at the HF level is clearly of great importance for extension of the pseudospectral method to MCSCF and some types of CI procedures. The actual performance of the method is difficult to predict at this point; however, the scaling with system size is certainly far superior to any previously proposed strategy. Furthermore, the present equations are quite similar to those which have to be solved in coupled-perturbed HF theory. It seems quite likely that large gains in efficiency in the calculation of second derivatives (a very time consuming, but necessary part of most investigations of molecular potential energy surfaces) can be achieved.

The actual determination of **x** is performed via the direct inversion in the iterative subspace approach. That is, the best solution (in a least-squares sense) possible is assembled from the set of iterative trial vectors. An improved subspace is obtained by preconditioning the **A** matrix by its diagonal elements, a standard strategy in conjugate gradient type methods. Only four iterations are typically required to obtain the best convergence possible with our approximate Hessian.

Optimization of the method is far from complete; in particular, integral cutoffs have not yet been introduced. Various other complex issues also need to be addressed.

E. Fock matrix updating

This strategy, introduced by Almlöf and co-workers for their direct SCF method,[9] is a natural complement to our multigrid approach. The idea is to avoid using expensive (fine or ultrafine) grids more than once. In a Fock matrix updating algorithm, only the correction to the old Fock matrix is assembled; if the difference between the old and new density matrices is small, much less precision in the quadrature scheme is required.

The iteration sequence utilized for glycine is shown in Table I. The initial iterations converge the medium grid to within 10^{-2} hartrees of its own solution. This set of coefficients is then used to assemble an ultrafine Fock matrix, which serves as the base for updating. The next iteration requires the fine grid, while all subsequent iterations can be carried out on the medium grid. The Newton–Raphson procedure is used beginning one iteration prior to the ultrafine iteration.

An important point to note here is that the ultrafine and fine grid integrals are used only once. Consequently, they never need to be stored; a direct procedure with regard to these two grids has no cost. Thus, the storage requirement for our algorithm is proportional at most to the medium grid size. It is, of course, possible to recalculate medium and coarse grid integrals as well; results will be given in Sec. III for both modes of running the program.

TABLE I. Pseudospectral SCF iteration scheme and convergence of total energy of glycine molecule.

	Iteration	Grid	E^{a}	$E - E_{\text{final}}{}^{b}$
	1	Medium	− 280.830 962	2.0
	2	Medium	− 283.567 300	− 0.72
	3	Medium	− 282.407 425	0.44
Begin the Newton–Raphson procedure				
	4	Medium	− 282.855 800	− 0.011
	5	Ultrafine	− 282.853 519	− 0.009 1
Begin Fock matrix updating				
	6	Fine	− 282.850 398	− 0.005 9
	7	Medium	− 282.844 435	0.000 013
	8	Medium	− 282.844 459	− 0.000 011
	9	Medium	− 282.844 447	0.000 001
	10	Medium	− 282.844 448	0.0

[a] Total energy in hartrees for glycine 0° conformation.

[b] Difference between current energy and converged energy, in hartrees.

F. Parameter optimization

Conventional RH programs yield useful chemical information because they use carefully parametrized basis sets. The development of contracted basis sets with optimized contraction structure and exponents required years of intense effort. Indeed, further development along these lines is still taking place, particularly for transition metals and for correlated calculations.

The pseudospectral HF method necessitates in addition design of atomic grids and dealiasing functions. As indicated earlier, we adopt the view that the procedure which combines these into a molecular parameter set must be completely automatic, so that there is no question of having to adjust the grid generation algorithm for each new molecule. The algorithm which carries out this synthesis is in fact identical to that used in Ref. 4; while alterations are certainly possible, its performance up to now has been quite satisfactory.

At present, we have developed four atomic grids for hydrogen and the remaining first row atoms. The same grids have been used for H, C, N, and O in calculations on a variety of small molecules and on glycine. In a subsequent paper, we will report a series of results for a large set of molecules. Here, we focus on the glycine results. However, satisfactory energies were obtained in all test cases to date with the current parametrization.

A standard dealiasing set is appended to each grid for each atom. There are differences among the dealiasing sets for each atom, dictated by the different basis sets for the atoms. In the interest of computational efficiency, we group the dealiasing functions into blocks, containing *s*, *p*, and sometimes *d* and *f* functions. The exponents are arranged such that the uncontracted basis functions can be included in the block structure. For example, the 6-31G** basis set for carbon includes *d* functions with exponent 0.8, so *s*, *p*, and *f* functions with exponent 0.8 are included in the carbon dealiasing set. A crucial point in achieving high accuracy is, for the ultrafine grid, to space the exponents of the blocks by a factor of 2, rather than 3 (the latter ratio was used in our earlier paper).

Appendix A lists the four grids and the associated dealiasing sets for each atom. The only new feature in the grid design is the necessity of choosing shell orientations in the molecular reference frame. This does have some effect on the accuracy and more numerical experiments need to be carried out to achieve a greater understanding of this aspect of the grid design.

Optimization must be performed by evaluating the results of molecular (not atomic) calculations. This is because the most demanding task an atomic grid must perform is quadrature of functions centered on other atoms. Furthermore, truncation of the quadrature scheme by the Voronai polyhedral cutoffs is not present in a purely atomic problem; this is the least accurate part of the integration algorithm.

The present results were obtained by running a large number of numerical experiments, mostly on glycine. The results are robust in the sense that small changes in the grid or dealiasing set produce answers of comparable accuracy. Nevertheless, the reported parameters should be regarded as tentative, pending much more extensive optimization on a variety of molecules.

III. RESULTS

A. Accuracy

We used a 6-31G** basis set for all pseudospectral calculations. The three glycine geometries are all obtained from Ref. 13. These geometries have the C–C torsional bond angle at 0°, 150°, and 180°, respectively; the remaining degrees of freedom were optimized at the 4-21G level in Ref. 13 for each geometry. These geometries represent three points on the torsional potential curve (although they do not include the highest point on the barrier). We have chosen them simply as a test case in which we compare pseudospectral and RH results using the same basis sets. The RH results were obtained by running GAUSSIAN82 and GAUSSIAN86 with the same set of coordinates.

Table II shows the energies obtained for each geometry with pseudospectral and RH calculations, the latter at both the 6-31G** and 4-21G levels. The agreement of the pseudospectral and RH methods when the same basis set is used in all cases within 0.000 06 hartrees, or 0.04 kcal/mol. The pseudospectral relative error is of the same order. Note that the RH 4-21G and 6-31G** calculations differ substantially in total energy, but yield very similar relative energies. This cancellation of error is the property of the RH method which

TABLE II. Comparison of pseudospectral and Roothaan–Hall results: total and relative energies for glycine at three geometries.

	Total energy[a]			Energy relative to 0° conformation[a]		
	Roothaan–Hall[b]		Pseudospectral	Roothaan–Hall[b]		Pseudospectral
Conformation[c]	4-21G	6-31G**	6-31G**	4-21G	6-31G**	6-31G**
0°	– 282.164 275	– 282.844 468	– 282.844 448	0.0	0.0	0.0
150°	– 282.161 212	– 282.841 397	– 282.841 345	0.003 06	0.003 07	0.003 10
180°	– 282.161 379	– 282.841 671	– 282.841 679	0.002 90	0.002 80	0.002 77

[a] All energies in hartrees.

[b] Roothaan–Hall results obtained from programs GAUSSIAN86 (4-21G) and GAUSSIAN82 (6-31G**).

[c] Angle refers to rotation about the C–C bond, as defined in Ref. 13.

TABLE III. Comparison of pseudospectral (PS) and Roothaan–Hall (RH) results: eigenvalues of the Fock matrix.[a]

	0°[b]		150°		180°	
Orbital	RH[c]	PS	RH	PS	RH	PS
1	– 20.615 33	– 20.615 21	– 20.616 91	– 20.617 00	– 20.618 08	– 20.617 96
2	– 20.560 55	– 20.560 42	– 20.559 60	– 20.559 31	– 20.556 81	– 20.557 11
3	– 15.547 88	– 15.547 93	– 15.551 80	– 15.551 85	– 15.550 93	– 15.550 93
4	– 11.387 58	– 11.387 79	– 11.388 80	– 11.388 83	– 11.389 03	– 11.389 02
5	– 11.276 03	– 11.275 97	– 11.276 68	– 11.276 61	– 11.277 30	– 11.277 17
6	– 1.457 94	– 1.457 84	– 1.458 38	– 1.458 25	– 1.458 73	– 1.458 59
7	– 1.360 62	– 1.360 42	– 1.360 08	– 1.359 84	– 1.359 27	– 1.359 12
8	– 1.187 70	– 1.187 73	– 1.190 97	– 1.190 99	– 1.191 46	– 1.191 43
9	– 0.978 42	– 0.978 46	– 0.972 80	– 0.972 77	– 0.971 75	– 0.971 73
10	– 0.815 57	– 0.815 64	– 0.828 43	– 0.828 43	– 0.829 96	– 0.829 97
11	– 0.701 21	– 0.701 15	– 0.705 97	– 0.706 01	– 0.697 77	– 0.697 84
12	– 0.696 00	– 0.696 10	– 0.688 09	– 0.688 10	– 0.695 01	– 0.695 19
13	– 0.687 84	– 0.687 96	– 0.682 66	– 0.682 56	– 0.686 51	– 0.686 44
14	– 0.623 04	– 0.622 97	– 0.628 80	– 0.628 88	– 0.621 32	– 0.621 38
15	– 0.592 49	– 0.592 36	– 0.585 67	– 0.585 60	– 0.587 88	– 0.587 79
16	– 0.582 61	– 0.582 52	– 0.578 05	– 0.577 98	– 0.578 03	– 0.577 97
17	– 0.539 62	– 0.539 58	– 0.545 00	– 0.545 00	– 0.546 74	– 0.546 72
18	– 0.481 37	– 0.481 30	– 0.479 46	– 0.479 59	– 0.475 77	– 0.475 84
19	– 0.457 93	– 0.457 79	– 0.475 79	– 0.475 58	– 0.475 52	– 0.475 49
20	– 0.397 45	– 0.397 46	– 0.390 49	– 0.390 40	– 0.390 85	– 0.390 84

[a] All energies in hartrees.
[b] Angle refers to rotation about the C–C bond, as defined in Ref. 13.
[c] RH results obtained from program GAUSSIAN82.

is so difficult for a numerical approach to duplicate. Table II illustrates the success of the pseudospectral method in mimicking RH behavior, thus allowing the enormous literature of conclusions based on this cancellation of error to be taken over.

Tables III and IV display, respectively, the eigenvalues and one-electron energies for the RH and pseudospectral methods at the three geometries. Very close agreement is obtained for all of these quantities across geometry. Note, however, that the individual discrepancies in each term are significantly larger than that for the total energy. This is not accidental; we have observed identical behavior for hun-

TABLE IV. Comparison of pseudospectral (PS) and Roothaan–Hall (RH) results: one-electron energies.[a]

	0°[b]		150°		180°	
Orbital	RH[c]	PS	RH	PS	RH	PS
1	– 39.693 74	– 39.693 71	– 40.085 90	– 40.085 89	– 40.104 69	– 40.104 67
2	– 39.976 75	– 39.976 73	– 39.645 66	– 39.645 64	– 39.636 61	– 39.636 57
3	– 32.213 93	– 32.213 92	– 32.287 92	– 32.287 92	– 32.297 61	– 32.297 58
4	– 29.292 92	– 29.292 91	– 29.255 20	– 29.255 20	– 29.247 30	– 29.247 31
5	– 27.752 84	– 27.752 84	– 27.723 79	– 27.723 78	– 27.721 35	– 27.721 36
6	– 15.369 77	– 15.369 36	– 15.450 94	– 15.449 76	– 15.452 37	– 15.452 04
7	– 15.360 18	– 15.359 70	– 15.320 99	– 15.320 66	– 15.327 26	– 15.326 27
8	– 13.340 77	– 13.340 66	– 13.383 37	– 13.383 29	– 13.393 59	– 13.393 33
9	– 13.420 56	– 13.420 68	– 13.424 07	– 13.424 13	– 13.415 75	– 13.415 84
10	– 13.375 66	– 13.376 22	– 13.574 65	– 13.575 08	– 13.598 71	– 13.598 82
11	– 13.826 00	– 13.826 79	– 12.893 47	– 12.888 88	– 12.104 05	– 12.103 40
12	– 12.080 43	– 12.078 46	– 13.628 52	– 13.629 09	– 13.833 44	– 13.833 07
13	– 13.826 71	– 13.826 24	– 13.145 44	– 13.149 60	– 13.788 45	– 13.791 15
14	– 13.228 61	– 13.228 93	– 13.386 46	– 13.388 48	– 13.412 36	– 13.413 05
15	– 12.877 33	– 12.879 30	– 13.448 87	– 13.463 42	– 13.026 06	– 13.041 45
16	– 13.482 36	– 13.480 51	– 13.082 28	– 13.068 60	– 13.437 38	– 13.422 32
17	– 11.951 49	– 11.952 63	– 12.012 05	– 12.011 10	– 12.081 37	– 12.082 15
18	– 13.796 83	– 13.797 83	– 13.662 24	– 13.668 84	– 13.400 34	– 13.400 77
19	– 13.872 31	– 13.871 53	– 13.452 79	– 13.445 60	– 13.607 56	– 13.606 24
20	– 12.211 20	– 12.211 94	– 12.516 16	– 12.516 70	– 12.545 98	– 12.545 06

[a] All energies in hartrees.
[b] Angle refers to rotation about the C–C bond, as defined in Ref. 13.
[c] RH results obtained from program GAUSSIAN82.

dreds of alternative parameter sets. In this sense, the pseudospectral method does have its own cancellation of errors.

Of particular note in Table III is the presence of accidentally, nearly degenerate orbitals in glycine which change their energy gaps significantly as a function of geometry. This means that small discrepancies in the off-diagonal Fock matrix elements between these orbitals has potentially large consequences for the eigenvector coefficients and hence for the one-electron energy. The success of the pseudospectral method in dealing with this problem is encouraging with regard to its generality.

Table I shows the total energy error as a function of iteration number, when the code is run according to the iterative scheme indicated. We use the complete neglect of differential overlap (CNDO) initial guess of GAUSSIAN82 as a starting point and iterate until the total energy is converged to within 10^{-6} hartrees. Convergence at the start of the Newton–Raphson procedure (iteration 4) is nearly quadratic, then deteriorates as the solution is reached. This is expected when utilizing an approximate Hessian, as is done here (the coarse grid is used to construct the action of the second derivatives on the trial rotation vector).

B. Computational efficiency

We report in Table V CPU times in seconds for our pseudospectral code run in two modes; with and without recalculation of integrals, and for the programs GAUSSIAN82, GAUSSIAN86, and GRADSCF. All timings are for closed shell HF calculations on glycine at the 0° geometry, run on the University of Texas Center for High Performance Computing Cray X-MP/24 (the X-MP/24 has two processors, but all of the codes tested here, including the pseudospectral, use only one). A significant advantage is achieved even as compared to GRADSCF, which is specifically designed for Cray computers. Furthermore, we expect our timing results to improve substantially as algorithmic development proceeds.

Table V also displays the disk storage, input/output, and core memory requirements of each program. We have chosen to report the I/O in terms of the total number of words which must be read or written during the entire program cycle. The use of the solid state storage device (SSD) on the Cray means that I/O is in fact quite inexpensive for glycine in the RH programs, because all of the integrals can fit on the SSD. However, this would not be true for larger molecules.

TABLE V. Comparison of pseudospectral and conventional programs: computational efficiency.[a]

	CPU time (s)	Disk storage[b] (megawords)	Total I/O [c] (megawords)	Memory[d] (megawords)
GAUSSIAN82	711.7	20.0	439.2	0.164
GAUSSIAN86	564.5	24.6	406.5	0.381
GRADSCF	194.8	13.5	286.5	0.704
Pseudospectral, integrals stored	89.3	16.6	200.1	1.020
Pseudospectral, integrals recalculated	129.2	6.6	63.0	1.020

[a] All data are from glycine 6-31G** single-geometry (0°) HF calculations on a Cray X-MP/24 without multitasking.

[b] Maximum disk storage space used.

[c] Total amound read from or written to disk during calculation.

[d] Central memory used (maximum).

We have not efficiently optimized our usage of memory and disk space (the pseudospectral code is at present a research project rather than a commercial product). We currently run our program using about one megaword of core memory; this could be reduced. Most importantly, the memory size does not increase substantially as the molecule grows in size. Thus, if all integrals are recalculated, there is no barrier to carrying out calculations on a molecule of arbitrary size; the disk storage needed grows only as N^2 (or, incorporating various types of cutoffs N).

The fact that integral recalculation (without use of integral cutoffs) increases CPU time by only 45% is of great importance for practical applications. It implies that current limitations on many machines due to lack of disk storage can be overcome. How significant this will be in the future remains to be seen; there are many directions in which computational hardware can be improved and the balance of CPU efficiency, fast memory, and disk space will be different in every new design. Nevertheless, in the present environment, elimination of large scale disk requirements appears to be quite useful.

IV. CONCLUSION

We have demonstrated in this paper that a numerical method is capable of obtaining accurate solutions (in the sense of computing reliable energy differences) to the HF equations for a reasonably large and complicated molecule by the standards of current quantum chemical technology. This represents the first such demonstration in the roughly 35 years that computational quantum chemistry has been a workable proposition. In particular, the present approach efficiently solves the problem of numerically treating nonlocal exchange operators, which is essential if one wishes to pursue a systematic *ab initio* approach, as opposed to a local density functional type of theory.

It is still too early to precisely predict what sort of advantage (if any) will be realized over state-of-the-art conventional quantum chemistry programs, when both methods are optimized for a given hardware configuration. However, it is quite clear that the pseudospectral method is at this point a serious competitor, with a number of intrinsic properties which suggest that it will ultimately lead to significant reductions in computational effort for large molecules. It should be kept in mind that substantial changes in the algorithm are not only possible, but even likely, and that these will yield significant improvements over the timings reported in this paper.

One example of a major algorithmic modification that should yield large improvements in efficiency is the expression of the primitive integrals of the theory (three-center, one-electron integrals) in terms of multipole moments at long distances. The multipoles can be evaluated much more rapidly (by a factor of 5–10) than the current, exact integral evaluation procedure; furthermore, it is possible to carry out integrals over the multipoles analytically. Preliminary results indicate that this technique, in conjunction with suitable integral cutoffs, will lead to substantial CPU and storage reductions even for glycine and considerably more for larger systems. In effect, the asymptotic scaling of integral evaluation and storage can be reduced to $NN_{nn}N_g$, where N is the number of basis functions, N_{nn} the number of neighboring basis functions for which nonzero primitive integrals must be evaluated, and N_g is the number of grid points in the near field (for which exact integral evaluation is required). Note that only N grows with molecular size. This is to be contrasted with conventional two-electron integral codes, where the asymptotic number of required integrals scales roughly as $N^2N_{nn}^2$ (the assumption has been made here that the molecule is not of sufficient size to allow truncation based on diminishment of the Coulomb field—this limit is not a practical one). When one in addition considers that the individual integrals in the pseudospectral method are much less expensive and that the multigrid approach reduces N_g substantially for all but a few iterations, the potential advantages for very large molecules become overwhelming.

A second key modification is localization of the least-squares procedure. In principle, it should not be necessary to fit functions over the entire molecule in order to produce a quadrature scheme for a localized basis function. Preliminary numerical experiments indicate that one only needs to keep basis and dealiasing functions centered on atoms which are within range of the function in question. This means that the maximum linear equation system that will need to be solved is on the order of 500–1000 in dimension. Such a restriction in turn implies that this part of the algorithm will not be an important contributor to CPU time or storage for a very large molecule.

A final topic is extension of the method to other types of calculations (e.g., analytic derivatives, excited states) and to correlated wave functions. As stated above, an analytic gradient procedure has been successfully developed and will be reported shortly. No difficulties are expected in second derivative evaluation. Any one-electron properties (e.g., dipole moments) can be computed as in conventional codes, as a solution in terms of standard basis sets is available. Similarly, modification of the one-electron matrix (e.g., use of pseudopotentials) are straightforward to incorporate into the present code.

We have recently combined our HF program with the GVB2P5 program of Goddard and co-workers.[12] Accuracy for pseudospectral GVB calculations appears to be comparable to that obtained with HF wave functions. These results will be reported in a subsequent publication.

In summary, we believe that pseudospectral electronic structure calculations have a promising future and hold out the prospects of carrying out high quality *ab initio* calculations on large molecules and molecular assemblies in reasonable CPU times. If this promise can be realized, the impact of *ab initio* quantum chemistry upon systems of interest to the majority of chemists will be greatly enhanced.

ACKNOWLEDGMENTS

This work was supported by grants from the National Institute of Health and the National Science Foundation. R. A. F. is a Camille and Henry Dreyfus Teacher–Scholar and the recipient of Research Career Development Award from the National Institutes of Health, Institute of General Medical Sciences. Use of the Cray X-MP/24 supercomputer at the University of Texas Center for High Performance Computing is gratefully acknowledged.

APPENDIX A: ATOMIC GRIDS AND DEALIASING FUNCTIONS

As in Ref. 4, the atomic grids are composed of spherical shells containing $(L+1)^2$ gridpoints. The current grids use L values of 1, 2, 4, 10, and 11. For spherical shells with $L=1$, the angular distribution scheme of Ref. 4 was modified such that there are now two rings of four (rather than

TABLE VI. Coarse and medium grids for atoms H, C, N, and O.

	Coarse grid for H		Coarse grid for C, N, O		Medium grid for H		Medium grid for C, N, O	
Shell	Radius[a]	L[b]	Radius	L	Radius	L	Radius	L
1	0.2	1	0.022	1	0.113	2	0.029	2
2	0.8	1	0.25	1	0.3	2	0.102	2
3	1.85	1	0.75	1	0.7	2	0.202	2
4	3.4	1	1.438	2	1.085	4	0.307	2
5			3.1	2	1.369	4	0.4	2
6					1.818	4	0.6	2
7					2.7	4	1.114	4
8					4.3	4	1.515	4
9					6.5	2	1.889	4
10							2.322	4
11							2.9	4
12							4.1	4
13							8.0	2

[a] Distance in bohr from nucleus to spherical shell.

[b] Spherical shells contain $(L+1)^2$ gridpoints, with angular distribution as given in Ref. 4, except $L=1$, which has $2^*(L+1)^2 = 8$ points.

TABLE VII. Fine and ultrafine grids for atoms H, C, N, and O.

	Fine grid		Ultrafine grid	
Shell	Radius[a]	L[b]	Radius[a]	L[b]
1	0.065	2	0.065	4
2	0.146	2	0.146	4
3	0.254	4	0.254	4
4	0.4	4	0.4	4
5	0.6	4	0.6	4
6	0.811	4	0.874	10
7	1.037	4	1.173	10
8	1.279	4	1.501	11
9	1.539	4	1.862	10
10	1.820	4	2.261	11
11	2.123	4	2.704	10
12	2.452	4	3.2	11
13	2.809	4	3.8	10
14	3.2	4	4.222	10
15	3.8	4	4.765	2
16	5.098	4	5.486	2
17	8.0	2	6.494	2
18			8.0	2

[a] Distance in bohr from nucleus to spherical shell.

[b] Spherical shell contains $(L+1)^2$ grid points, with angular distribution as given in Ref. 4.

TABLE IX. Dealiasing set for medium and fine grids.

Atom	L	Exponents[a]						
H	*s*	0.08	*	0.32	0.64	1.1	3.3	9.9
	p	0.08	0.161 277 8	0.32	0.64	*	3.3	9.9
C	*s*			0.3	0.8	2.4	7.2	21.6
	p			0.3	0.8	2.4	7.2	21.6
	d			0.3	*	2.4	7.2	
	f				0.8			
N	*s*			0.35	0.8	2.4	7.2	21.6
	p			0.35	0.8	2.4	7.2	21.6
	d			0.35	*	2.4	7.2	
	f				0.8			
O	*s*			0.4	0.8	2.4	7.2	21.6
	p			0.4	0.8	2.4	7.2	21.6
	d			0.4	*	2.4	7.2	
	f				0.8			

[a] Asterisks indicate the location of uncontracted basis functions.

two) points each, making eight gridpoints altogether. Tables VI and VII list the radial shell positions and L values for the four grids (coarse, medium, fine, and ultrafine). For the coarse and medium grids, two parameter sets were used; one for hydrogen and another for C, N, and O. For the larger grids, a single gridpoint distribution was used for all atoms.

Table VIII shows the dealiasing functions used with the coarse grid. Because there are so few dealiasing exponents, no attempt was made to use the block algorithm described in Sec. II B with this grid. Tables IX and X display the dealiasing sets used for the three larger grids. The exponents are fit to the basis set in such a way that the uncontracted basis functions fill gaps in the *sp*, *spd*, or *spdf* block structure of the dealiasing set. Blocks of dealiasing exponents are otherwise spaced in a regular fashion.

There is clearly room for further optimization here; e.g., a separate dealiasing set could be developed for the fine grid and individual grids could be created for each atom. Nevertheless, the fact that satisfactory results have been obtained without extensive parameter optimization is encouraging.

TABLE VIII. Dealiasing set for coarse grid.

Atom	L	Exponents
H		None
C, N, O	*s*	3.0
	p	5.8

APPENDIX B: PSEUDOSPECTRAL NEWTON–RAPHSON ALGORITHM

Following Levy *et al.*,[11] the elements of the second derivative matrix **A** are given by

$$\mathbf{A}_{vo,vo} = \langle v|F|v\rangle - \langle o|F|o\rangle + 3(ov|ov) - (oo|vv), \tag{B1}$$

$$\mathbf{A}_{vo,v'o} = \langle v'|F|v\rangle + 3(vo|v'o) - (oo|vv'), \tag{B2}$$

$$\mathbf{A}_{vo,vo'} = \langle o'|F|o\rangle + 3(vo'|vo) - (oo'|vv), \tag{B3}$$

$$\mathbf{A}_{vo,v'o'} = 4(vo|v'o') - (vo'|v'o) - (o'o|v'v), \tag{B4}$$

where o and o' represent (distinct) occupied molecular orbitals and v and v' virtual orbitals. Equations (B1)–(B4) correspond to Eqs. (44)–(47) of Ref. 11.

We split this matrix into three parts: a diagonal part **D** whose elements are the difference in diagonal Fock matrix elements between the two molecular orbital indices; the part which depends upon off-diagonal Fock matrix elements (this is typically quite small as one approaches the self-consistent solution), which we call **H**; and the remainder **G** which is composed of two electron integrals over molecular orbitals. Thus

$$\mathbf{D}_{vo,vo} = \langle v|F|v\rangle - \langle o|F|o\rangle, \tag{B5}$$

$$\mathbf{D}_{vo,v'o} = \mathbf{D}_{vo,vo'} = \mathbf{D}_{vo,v'o'} = 0, \tag{B6}$$

$$\mathbf{H}_{vo,v'o} = \langle v'|F|v\rangle, \quad \mathbf{H}_{vo,vo'} = \langle o'|F|o\rangle, \tag{B7}$$

$$\mathbf{H}_{vo,vo} = \mathbf{H}_{vo,v'o'} = 0. \tag{B8}$$

If the notational restriction that v and v' (o and o') represent necessarily distinct orbitals is relaxed, the contributions to **G** from Eqs. (B1)–(B4) can be written compactly as

$$\mathbf{G}_{vo,v'o'} = 4(vo|v'o') - (vo'|v'o) - (o'o|v'v). \tag{B9}$$

Note that **G** does retain some diagonal terms.

The most straightforward approach to solving the Newton–Raphson equations is to directly construct **G** via a four-index transform of two-electron integrals, assemble **A**, and then use a standard matrix inversion algorithm to solve the linear system $\mathbf{Ax} = \mathbf{b}$. However, the effort involved in this procedure scales as N^5 and furthermore is not diminished by using the pseudospectral method, because the last step (in which the elements of **G** are actually evaluated) cannot require less effort than N^4 (the number of matrix ele-

TABLE X. Dealiasing set for ultrafine grid.

Atom	L	Exponents[a]									
H	s	0.04	0.08	*	0.32	0.64	1.1	2.2	4.4	8.8	17.6
	p	0.04	0.08	0.161 277 8	0.32	0.64	*	2.2	4.4	8.8	17.6
	d		0.08	0.161 277 8	0.32	0.64	1.1	2.2			
C	s	0.04	0.08	*	0.35	0.8	1.6	3.2	6.4	12.8	25.6
	p	0.04	0.08	*	0.35	0.8	1.6	3.2	6.4	12.8	25.6
	d		0.08	0.168 714	0.35	*	1.6	3.2	6.4		
	f					0.8					
N	s	0.05	0.11	*	0.41	0.8	1.6	3.2	6.4	12.8	25.6
	p	0.05	0.11	*	0.41	0.8	1.6	3.2	6.4	12.8	25.6
	d		0.11	0.212 031	0.41	*	1.6	3.2	6.4		
	f					0.8					
O	s	0.07	0.14	*	0.48	0.8	1.6	3.2	6.4	12.8	25.6
	p	0.07	0.14	*	0.48	0.8	1.6	3.2	6.4	12.8	25.6
	d		0.14	0.270 005 8	0.48	*	1.6	3.2	6.4		
	f					0.8					

[a] Asterisks indicate the location of uncontracted basis functions.

ments) $\times M$ (the number of grid points). In any case, this procedure is clearly not competitive with alternative acceleration methods for large problems.

A much more efficient approach is to use an iterative method to solve the linear system, in which case one needs an algorithm for acting with **G** on **x**. This may be carried out via the pseudospectral method as follows (note that this algorithm avoids the necessity of transforming integrals to the molecular frame):

(i) Form the matrix **Y** from **x** as follows:

$$\mathbf{Y}_{vo} = \mathbf{Y}_{ov} = \mathbf{x}_{vo}, \quad \mathbf{Y}_{oo'} = \mathbf{Y}_{vv'} = 0. \tag{B10}$$

Note that **Y** differs from **X** in that **Y** is symmetric, while **X** is skew symmetric.

(ii) Transform **Y** to the atomic orbital space, creating a "pseudodensity" matrix **P**:

$$\mathbf{P} = \mathbf{c}\mathbf{Y}\mathbf{c}^{\dagger} \tag{B11}$$

which then has elements

$$\mathbf{P}_{kl} = \sum_{v'} \sum_{o'} (\mathbf{c}_{kv'}\mathbf{c}_{lo'} + \mathbf{c}_{ko'}\mathbf{c}_{lv'})\mathbf{x}_{v'o'}. \tag{B12}$$

This requires an effort of magnitude N^3 and hence is negligible compared to other steps.

(iii) Pass **P** to the usual algorithm for assembling the two-electron part of the Fock matrix from the density matrix.

(iv) Transform the resultant "Fock" matrix **F** back to molecular orbital (MO) space. This is then the result of **G** acting on **x**.

Proof: We wish to show that

$$[\mathbf{c}^{\dagger}\mathbf{F}\mathbf{c}]_{vo} = [\mathbf{G}\mathbf{x}]_{vo}. \tag{B13}$$

Expansion of the left-hand side of Eq. (A13) gives

$$[\mathbf{c}^{\dagger}\mathbf{F}\mathbf{c}]_{vo} = \sum_{i} \sum_{j} \mathbf{c}_{iv}\mathbf{c}_{jo}\mathbf{F}_{ij} \tag{B14}$$

and the elements of F are

$$\mathbf{F}_{ij} = \sum_{k} \sum_{l} \mathbf{P}_{kl}[2(ij|kl) - (ik|jl)]. \tag{B15}$$

Combination of Eqs. (B12), (B14), and (B15) yields, aftei some rearrangement,

$$\begin{aligned}[\mathbf{c}^{\dagger}\mathbf{F}\mathbf{c}]_{vo} = \sum_{v'o'} \Bigg[& 2 \sum_{ijkl} \mathbf{c}_{iv}\mathbf{c}_{jo}\mathbf{c}_{kv'}\mathbf{c}_{lo'}(ij|kl) \\ & + 2 \sum_{ijkl} \mathbf{c}_{iv}\mathbf{c}_{jo}\mathbf{c}_{ko'}\mathbf{c}_{lv'}(ij|kl) \\ & - \sum_{ijkl} \mathbf{c}_{iv}\mathbf{c}_{ko'}\mathbf{c}_{jo}\mathbf{c}_{lv'}(ik|jl) \\ & - \sum_{ijkl} \mathbf{c}_{iv}\mathbf{c}_{kv'}\mathbf{c}_{jo}\mathbf{c}_{lo'}(ik|jl) \Bigg] \mathbf{x}_{v'o'},\end{aligned} \tag{B16}$$

which can be written in terms of two-electron integrals in MO space as

$$\begin{aligned}[\mathbf{c}^{\dagger}\mathbf{F}\mathbf{c}]_{vo} &= \sum_{v'o'} [2(vo|v'o') + 2(vo|o'v') \\ &\qquad - (vo'|ov') - (vv'|oo')]\mathbf{x}_{v'o'} \\ &= \sum_{v'o'} [4(vo|v'o') - (vo'|v'o) - (o'o|v'v)]\mathbf{x}_{v'o'} \\ &= \sum_{v'o'} \mathbf{G}_{vo,v'o'}\mathbf{x}_{v'o'} = [\mathbf{G}\mathbf{x}]_{vo}.\end{aligned} \tag{B17}$$

The above algorithm thus scales like N^2M, just as the ordinary Fock matrix assembly in the pseudospectral method does. We considerably reduce M by using the coarse grid. This implies the use of an approximate Hessian in the Newton–Raphson procedure, so that quadratic convergence degrades as one approaches the solution. Fortunately, when one is very close to the solution, Fock matrix updating is inexpensive, so that carrying out a few extra iterations in this region is not costly.

Given the above procedure for acting with **G** on **x**, we can now specify the entire iterative algorithm:

(i) As noted above, the elements of **H** are very small and exclusion of these terms does not change the convergence rate significantly. Therefore, we assume $\mathbf{A} = \mathbf{D} + \mathbf{G}$.

(ii) Produce an initial guess for **x**; this is done by assuming that **G** is zero and solving the diagonal system $\mathbf{Dx} = \mathbf{b}$.

(iii) Obtain the vector **Gx** using the algorithm outlined above.

(iv) Form the vector $\mathbf{Ax} = \mathbf{Dx} + \mathbf{Gx}$.

(v) Solve the diagonal system

$$\mathbf{Dx}_{\text{new}} = \mathbf{b} - \mathbf{Gx} \qquad \text{(B18)}$$

to obtain the new, preconditioned guess for **x**.

(vi) Repeat steps (iii)–(v), storing all of the **x** and **Ax** vectors. This allows construction of the L by k [k is the number of iterations of steps (iii)–(v) above] matrices **M** and **N**, where the columns of **M** are the various **x** vectors and the columns of **N** are the **Ax** vectors.

(vii) Solve the equation $\mathbf{Nz} = \mathbf{b}$ by least squares

$$\mathbf{z} = [\mathbf{N}^{\dagger}\mathbf{N}]^{-1}\mathbf{N}^{\dagger}\mathbf{b}. \qquad \text{(B19)}$$

(viii) Find the best solution **x** to the original problem $\mathbf{Ax} = \mathbf{b}$ by forming a linear combination of the k trial solutions contained in **M**:

$$\mathbf{x} = \mathbf{Mz}. \qquad \text{(B20)}$$

If one wishes to avoid storage of the **x** vectors, the above iterative procedure can be equivalently replaced by the conjugate gradient procedure with **D** as a preconditioner. The computational effort is roughly the same, but significantly less storage is required if one needs to carry out a substantial number of iterations. As only four iterations are needed in the present case, this is not a major issue.

Once the optimal **x** vector is determined, the corresponding **X** matrix is exponentiated approximately via a Taylor series expansion, a rapidly convergent procedure when one is close to the solution.

[1] R. Friesner, Chem. Phys. Lett. **116**, 39 (1985).

[2] R. Friesner, J. Chem. Phys. **85**, 1462 (1986).

[3] R. Friesner, J. Chem. Phys. **86**, 3522 (1987).

[4] R. Friesner, J. Phys. Chem. **92**, 3091 (1988).

[5] S. A. Orszag, Stud. Appl. Math. **51**, 253 (1972); **50**, 293 (1971); Phys. Rev. Lett. **26**, 1100 (1971); D. Gottlieb and S. Orszag, *Numerical Analysis of Spectral Methods: Theory and Application* (SIAM, Philadelphia, 1977).

[6] D. Kosloff and R. Kosloff, J. Comput. Phys. **52**, 35 (1983); R. B. Gerber, A. T. Yinnon, and R. Kosloff, Chem. Phys. Lett. **105**, 523 (1984); J. C. Light, I. P. Hamilton, and J. V. Lill, J. Chem. Phys. **82**, 1400 (1985).

[7] C. C. J. Roothaan, Rev. Mod. Phys. **23**, 69 (1951).

[8] W. J. Hehre, R. Ditchfield, and J. A. Pople, J. Chem. Phys. **56**, 2257 (1972); P. C. Harihan and J. A. Pople, Theor. Chim. Acta (Berlin) **28**, 213 (1973).

[9] J. Almlöf, K. Faegri, Jr., and K. Korsell, J. Comput. Chem. **3**, 385 (1982).

[10] P. Pulay, Chem. Phys. Lett. **73**, 393 (1980).

[11] J. Douady, Y. Ellinger, R. Subra, and B. Levy, J. Chem. Phys. **72**, 1452 (1980).

[12] F. W. Bobrowicz and W. A. Goddard III, in *Modern Theoretical Chemistry: Methods of Electronic Structure Theory*, edited by H. F. Schaefer III (Plenum, New York, 1977), Vol. 3, p. 79.

[13] K. Siam, V. J. Klimkowski, J. D. Ewbank, C. Van Alsenoy, and L. Schäfer, J. Mol. Struct. Theochem. **110**, 171 (1984).