JOURNAL OF COMPUTATIONAL PHYSICS 26, 243–251 (1978)

Note

BERKELEY: An "Open Ended" Configuration Interaction (CI) Program Designed for Minicomputers*

Introduction

During the past 3 years we have been experimenting with the use of a high performance minicomputer for relatively large scale theoretical studies of molecular electronic structure [1, 2]. These studies have demonstrated that such minicomputers can be very competitive with traditional large machines for a variety of scientific applications. Specifically, we have found the Harris Corporation Slash Four mini to be a factor of ~27 slower than the CDC 7600. However the cost of 1 h of CDC 7600 time (including associated input/output charges) is at least $1000, while the Harris machine costs us only about $8/hr [1, 2]. Thus we find that in the most straightforward applications, e.g., the use of the self-consistent-field (SCF) programs POLYATOM [3] and GAUSSIAN 70 [4], the mini is at least a factor of three less expensive to use than the traditional large machine.

A lingering doubt concerning the use of minicomputers centers about the ability (or lack of same) to approach the most challenging problem in electronic structure theory—the correlation problem [5]. In a limited sense, this challenge has recently been met by Dykstra's development [6] of the method of self-consistent electron pairs (SCEP) on the Harris machine. The shortcoming of the SCEP method, while being both elegant and efficient, is that it is currently limited, much like the powerful Roos method [7], to singly and doubly substituted configurations relative to a single closed-shell reference configuration. And it is certainly well established [8] that there are numerous important chemical problems, the solution of which requires general configuration interaction (CI) methods. Thus it seems clear that a major challenge to any minicomputer concerns that machine's adaptability to general, large scale CI techniques.

The BERKELEY CI Program

Although the computational details of general CI methods can be formidable, it is generally agreed [8] that the procedure involves several relatively independent steps. We employ four such steps in the BERKELEY system.

* Research supported by the National Science Foundation under Grants GP-39317 and GP-41509X.

0021-9991/78/0262–0243$02.00/0

A. *Integral Transformation*

The transformation from integrals $(ab \mid cd)$ over basis functions to integrals $(ij \mid kl)$ over molecular orbitals has been the subject of several important papers [8–13]. The first critical innovation was the replacement of the straightforward N^8 summation (N is the number of basis functions) by four quarter transformations, yielding an N^5 procedure. The second breakthrough was made more recently by Yoshimine [11] and allows the straightforward use of very large basis sets. By utilizing very elegant direct access sorting techniques, Yoshimine has shown that only N^2 partially transformed integrals need be held in central memory at any given time. Thus with a machine the size of the Harris Slash Four, having ~20,000 words of memory available for floating point arrays, the full four index transformation of 140 basis functions is possible.

The use of the Yoshimine algorithms can sometimes be avoided for systems of high symmetry by carefully blocking [8] integrals by symmetry type (e.g., the $(a_1a_1 \mid b_2b_2)$ block). And in fact we have made extensive use of such blocking techniques in our earlier CI programs [14]. Nevertheless, we have now concluded that the Yoshimine approach is mandatory for even moderate-sized molecular systems (e.g., potential energy surfaces) with little or no spatial symmetry. Therefore, it was decided to ignore the blocking structure in this first version of the BERKELEY programs.

The above decision turned out to be a blessing in disguise since it led to an important finding. In test computations on the $C(CN)_2$ molecule (dicyanocarbene) using 50 basis functions, the first all FORTRAN version of the four index transformation required 300 min. Comparison with published computation times [11, 13] suggests that this timing is representative of a carefully designed program. However, it was clear to us that due to the C_{2v} symmetry of $C(CN)_2$, a vast number of multiplications by zero were being carried out. One of us (RRL) then reorganized the structure of the quarter transformation, with primary emphasis on the rapid identification of zero integrals (or quarter transformed integrals). Upon identification of a zero integral, all multiplications [50 in the case of the $C(CN)_2$ problem described above] of that integral are immediately bypassed.

For the 50 basis function $C(CN)_2$ problem the transformation time is thus reduced to 86 min. It should be noted that of this time, 63 min is primarily devoted to floating point multiplication and 23 min to the Yoshimine sorting procedure. Thus we suspect that our final time of 86 min (roughly 3 min of 7600 time) is not much more than would be required for the comparable fully blocked, completely in-core transformation.

B. *Formula Tape Generation*

In our research on potential energy surfaces and hypersurfaces [15], it has proved advantageous to divide the computation of Hamiltonian matrix elements H_{ij} into a two step procedure [8]. This approach is very effective if a large number of CI calculations are to be carried out using the same set of configurations [16]. In the first step, a "formula tape" is prepared which specifies the precise makeup of each nonzero H_{ij}

as a linear combination of one- and two-electron integrals. This step need be carried out only once for a given molecular system. Hence, if one uses the same formula tape to compute 500 points on a potential surface, the cost per surface point of the formula tape becomes negligible.

In this first version of the BERKELEY program the CI is carried out in terms of Slater determinants, rather than configurations (symmetry-adapted [5, 8] linear combinations of determinants). The obvious disadvantage of this approach is that it increases the size of the matrix **H** to be diagonalized. However, as we shall see, the time required for the extraction of the lowest eigenvalue and corresponding eigenfunction is not great in any case. Furthermore, the restriction to Slater determinants allows us to focus exclusively on the most demanding part of the problem, the manipulation of numbers of integrals and formulas much too large to be stored in central memory.

Following previous work by the Boys–Shavitt school [8, 17], each Slater determinant is stored in four 24-bit words, i.e., a total of 96 bits. This allows each of 96 spin orbitals to be occupied (bit turned on) or unoccupied (bit turned off). This means that only 48 molecular orbitals may be directly involved in the CI. However, this 48 need not include core orbitals, which are doubly occupied in all Slater determinants, or high lying virtual orbitals neglected in the CI procedure. Thus we do not at present anticipate that the restriction to 48 participating molecular orbitals will be a problem. In addition, the present version of BERKELEY requires that all Slater determinants reside in central memory during the formula tape step. Thus we are limited to ~10,000 determinants. However, relatively minor changes could be made to eliminate the latter restriction.

There are essentially four types of matrix elements H_{ij} to be considered, namely, those involving pairs of Slater determinants differing by 0, 1, 2, or more than two spin orbitals. The bit representation of Slater determinants is particularly effective in determining which of the four cases a particular matrix element belongs to. However, it should be noted that the Harris Slash Four is not as well designed in this regard as the CDC 6400, 6600, and 7600, which have explicit bit-counting instructions. On the Harris machine the comparison is carried out by a series of logical XOR statements, each of which compares two 24 bit words. The bit counting is carried out using masking, followed by table look-up.

As an example of the division of labor in the above procedure consider a 32 basis function calculation of singlet methylene, involving 2981 Slater determinants. The determination of which of the four H_{ij} types the 4,444,671 elements belong to requires 676 sec. The further generation of formulas for differences of 0, 1, and 2 spin orbitals requires 23, 99, and 600 sec, respectively.

The above procedure yields H_{ij} formulas ordered with i and j in lower triangle or canonical form, but two-electron integrals $(ij \mid kl)$ in arbitrary places. However, the reverse situation is preferable for the next step, the actual construction of numerical values of the H_{ij}. Therefore at this stage, Yoshimine's sorting algorithm [18] is used to reorder the formulas with integrals $(ij \mid kl)$ in canonical form and H_{ij} in arbitrary order. For the CH_2 case described above [1,008,896 H_{ij} — $(ij \mid kl)$ pairs], 340 sec of

Slash Four time (all times reported here are elapsed times) are required. It is hoped that these detailed timing breakdowns will be of value to future investigators.

C. *Construction of Hamiltonian Matrix Elements*

In this step the symbolic formula tape is used to obtain numerical values of the H_{ij}. This step determines the size of problem which can be handled on the minicomputer, since external storage demands are greatest here. That is, three long lists—the two-electron integrals, the H_{ij} formulas, and the H_{ij} values—are simultaneously processed in the construction stage. In addition a large scratch area is required to implement Yoshimine's algorithm [18].

Each nonzero two-electron integral is stored as a numerical value (48 bits; no labels are required since integrals are used in canonical order) for a total of 6 bytes. Each H_{ij} formula requires an $(ij \mid kl)$ label, an H_{ij} label, and a code representing the coefficient of the $(ij \mid kl)$ in the H_{ij}. All three of these pieces of information are packed into two 24-bit words (6 bytes). The actual H_{ij} values require two integers (each 24 bits long) plus the 48-bit floating point number, for a total of 12 bytes. Finally a pointer array is used to relate a set of ij codes to the actual integer values of i and j.

Our current external storage devices include [1] a 56,000,000 byte (or 56 MB, where MB designates megabytes) random access disk drive and a standard 9 track tape drive, with a capacity of slightly more than 20 MB. The disk, of course, must also hold the Slash Four operating system. The limitations of this system can be best illustrated by an example, a 42 basis function, 5359 determinant calculation on singlet methylene. At the construction stage there are 124,130 two-electron integrals (this is the number greater in magnitude than 5×10^{-9} hartrees), formulas requiring 2,679,488 48 bit words, 1,652,436 nonzero (by symmetry) H_{ij} values, and the same number of 48-bit words for the pointer array. These require a total of 47 MB of peripheral storage. In addition the scratch area required 23 MB, leaving only 6 MB for the system. Accordingly this scale of computation is about the limit of our current system.

Since the construction involves several long lists, it also is carried out using Yoshimine's direct access sorting techniques [18]. Specifically a block of ordered integrals is read into central memory and used to make contributions to all the H_{ij} values in which those integrals appear. We have found that 68 % of the time required for the construction step in the 5359 determinant CH_2 problem (42 basis functions) is devoted to sorting, while the remaining 32 % of the computation time would be required using a machine with infinitely large central memory. The total computation time for this step is 28 min.

A definite weakness of our present minicomputer system is that the final H_{ij} values are spooled onto the tape drive in the construction stage. This means that in order to proceed to the diagonalization step, one must first transfer the H_{ij} from tape to disk, a much faster input/output device. Since in many cases (e.g., the 5359 determinant CH_2 problem) the tape is nearly full, it must be read in its entirety. With our 37-in./sec (ips) drive this can require as much as 15 min, time during which the machine is essentially unused. Fortunately this situation will soon be relieved by our acquisition of a surplus second 56,000,000-byte disk drive. This will completely eliminate the above cited 15 min, as the H_{ij} values will be stored directly on the new disk. In addition it should allow us to handle problems of size 10,000 Slater determinants quite readily.

D. *Evaluation of Eigenvalues and Eigenvectors*

Certainly one of the important recent computational developments in molecular quantum mechanics has been the discovery of very efficient algorithms for the evaluation of the lowest eigenvalue(s) and corresponding eigenvector(s) of very large sparse matrices [19–22]. In fact, we have found that, even using Slater determinants rather than configurations, the evaluation of the lowest eigenvalue is not a bottleneck in the overall CI procedure.

For the BERKELEY system we have programmed two different iterative methods for the eigenvalue problem. The first of these, the method of optimal relaxation (MOR) developed by Shavitt *et al.* [21] appears fastest for cases where only the lowest eigenvalue (and corresponding eigenvector) is required. In most cases we find that convergence to 10^{-8} hartree in the total energy is achieved in six iterations, using the SCF reference determinant as the initial guess. It is worth noting that the read time for the **H** matrix is greatly reduced if an integral number of rows is contained on each record. This is done most efficiently if the records are of variable lengths.

The second method adopted appears more suitable (especially as regards convergence properties) to higher eigenvalues. This is the Compromise Method recently introduced by Davidson [22] and programmed for the BERKELEY system by BRB. Davidson's method is quite general and as implemented in the BERKELEY system will extract an arbitrary number of desired eigenvalues and eigenvectors. When only the lowest is required, about nine iterations are necessary for an accuracy of 10^{-8} hartrees in E. In the discussion that follows we give timing information for both methods.

Some Test Cases

Although the BERKELEY system is still in a relatively preliminary stage of development, timings achieved to date are sufficiently encouraging to be reported here. In the cases cited below, CI has been carried out including all singly and doubly substituted Slater determinants relative to a single reference determinant. This is done only for convenience and we wish to emphasize that our programs are completely general as regards the type of determinants allowed in the CI.

Tables I and II summarize a number of representative test cases. We emphasize from the outset, of course, that precise comparisons with other methods are never possible since the relative speeds of different computers vary from one program to the next. Nevertheless the times for the first two water calculations may be compared with those reported by Hosteny *et al.* (HGDPS) [23]. The latter computations were carried out on the CDC 6400 machine, which is perhaps 1.5 times [1] faster (in cpu time) than our minicomputer.

TABLE I

Test Calculations for the BERKELEY Configuration Interaction (CI) Program[a]

Calculation no.	Molecule	Number of basis functions	Reference determinant	Number of determinants	E (SCF)	E (CI)	Time (min.) first computation	Time (min.) subsequent computations
1	H_2O	14	$\cdots 2a_1^2\, 1b_2^2\, 3a_1^2\, 1b_1^2$	523	−76.009257	−76.135407	2.0	0.8
2	H_2O	14	$1a_1^2\, 2a_1^2\, 1b_2^2\, 3a_1^2\, 1b_1^2$	880	−76.009257	−76.148163	4.2	1.3
3	H_2O	35	$\cdots 2a_1^2\, 1b_2^2\, 3a_1^2\, 1b_1^2$	5255	−76.050697	−76.278909	105.6	34.0
4	$CH_2(^1A_1)$	32	$\cdots 2a_1^2\, 1b_2^2\, 3a_1^2$	2981	−38.889788	−39.026407	47.3	18.3
5	$CH_2(^1A_1)$	42	$\cdots 2a_1^2\, 1b_2^2\, 3a_1^2$	5359	−38.892733	−39.040646	163.5	66.1
6	$CH_2(^3B_1)$	42	$\cdots 2a_1^2\, 1b_2^2\, 3a_1\, 1b_1$	4542	−38.932454	−39.062157	129.5	57.0
7	$C(CN)_2(^1A_1)$	50	$\cdots 7a_1^2\, 1b_1^2\, 6b_2^2\, 1a_2^2\, 8a_1^2$	5490	−222.255980	−222.448005	218.2	119.3

[a] In the column labeled Reference Determinant, orbitals implied by the symbol ⋯ are held frozen, i.e., they are doubly occupied in all Slater determinants included in the CI. Energies are in hartree atomic units.

TABLE II

Computation Times for the BERKELEY CI Programs[a]

Calculation no.	Nonzero[b] integrals over basis functions	Nonzero[b] integrals over molecular orbitals	Four-index transformation time	Number of nonzero matrix elements H_{ij}	Formula tape time	Tape copy time	H_{ij} construction time	Lowest eigenvalue and eigenvector	
								Shavitt[c]	Davidson[d]
1	3,009	1,925	14	24,820	75		17	15	20
2	3,009	1,925	14	54,762	173		36	29	36
3	85,387	54,090	610	1,093,160	4295	350	1040	388	561
4	68,291	39,698	437	575,078	1738		392	269	338
5	178,894	112,984	1660	1,652,436	5847	780	1697	607	936
6	178,454	113,018	1671	1,251,860	4347	480	1312	437	634
7	370,642	232,337	5183	905,753	5934	400	1585	390	598

[a] All times are in seconds for the Harris Corporation Slash Four minicomputer.
[b] Nonzero here means greater than 5×10^{-9} in absolute value.
[c] Reference 21.
[d] Reference 22.

HGDPS report 23 sec of 6400 time for the integral transformation, and it seems clear that the present program (14 sec) is faster for this particular H_2O case. However, their formula tape requires only 23 sec, compared to the present 75. For the actual construction of the H matrix both codes require 17 sec, implying that the BERKELEY system is somewhat more efficient, given the speed advantage of the CDC 6400. HGDPS report only 5 sec to evaluate the lowest eigenvalue and eigenvector, and this would appear a much better result than the 15 sec reported in Table II. Such is not the case, however, since the matrix treated by HGDPS is only of size 224 × 224, since it is constructed from pure $^1A_1$ configurations. As noted in Table I our eigenvalue time refers to the 523 × 523 matrix in terms of simple Slater determinants. Although the use of Slater determinants rather than configurations slows down this last step somewhat, the eigenvalue time is still comparable to the transformation and construction times and does not significantly affect our total computation time. Hence the use of Slater determinants is at least partially justified.

For the second test case in Tables I and II HGDPS report 51 sec for the formula tape, 36 sec for H_{ij} construction, and 11 sec for their order 361 eigenvalue problem. It seems fair to conclude that the BERKELEY system is at least as efficient as the program of HGDPS except for the formula tape step. The latter finding is of course not surprising since HGDPS have made effective use of the CDC 6400 bit-counting instruction.

Direct comparison with previous work is not possible for calculations 3–8. However, calculation three was considered especially important since it reproduces (to 1×10^{-6} hartrees in the total energy) the recent result of Rosenberg and Shavitt [24]. It can hardly be overemphasized that there are many possible sources of error in a system of programs as complicated as BERKELEY. In this regard it should be noted that we have also reproduced a number of results obtained by Dykstra [6] on singlet methylene using his SCEP method. Since CI and SCEP are radically different approaches to the correlation problem, this agreement cannot be considered fortuitous.

Note added in proof. The capabilities of the BERKELEY system have been greatly enhanced during the past year. By working in terms of configurations, rather than Slater determinants, fully variational calculations including 15,006 configurations (42,456 determinants) have been completed on the Harris minicomputer.

Acknowledgments

We are deeply indebted to Professor I. Shavitt and Dr. Charles F. Bender for many stimulating discussions and very helpful suggestions. We benefited especially from an outstanding seminar given to the Electronic Structure Group at Berkeley by Dr. M. Yoshimine. Discussions with Professor William H. Fink were also very helpful. Finally HFS thanks Professor Frank E. Harris and Dr. Robert K. Nesbet for introducing him to the configuration interaction method 10 years ago.

References

1. (a) H. F. Schaefer, Are minicomputers suitable for large scale scientific computation? *in* "Proceedings of the Eleventh Annual IEEE Computer Science Conference (Washington, D. C., September 1975)."

(b) H. F. Schaefer and W. H. Miller, *Comput. Chem.* **1** (1977) 85.

(c) W. H. Miller and H. F. Schaefer, "Final Report to the National Science Foundation, Grant GP-39317, April 1976."

2. A. L. Robinson, *Science* **193** (1976), 470; W. G. Richards, *Nature (London)* **266** (1977), 18.
3. D. B. Neuman, H. Basch, R. L. Kornegay, L. C. Snyder, J. W. Moskowitz, C. Hornback, and S. P. Liebmann, "POLYATOM 2," Program No. 199, Quantum Chemistry Program Exchange, Indiana University, Bloomington, Indiana, 1972.
4. W. J. Hehre, W. A. Lathan, R. Ditchfield, M. D. Newton, and J. A. Pople, "GAUSSIAN 70," Program No. 236, Quantum Chemistry Program Exchange, Indiana University, Bloomington, Indiana, 1973.
5. H. F. Schaefer, "The Electronic Structure of Atoms and Molecules: A Survey of Rigorous Quantum Mechanical Results," Addison–Wesley, Reading, Mass., 1972.
6. C. E. Dykstra, H. F. Schaefer, and W. Meyer, *J. Chem. Phys.* **65** (1977).
7. B. Roos, *Chem. Phys. Lett.* **15** (1972), 153.
8. I. Shavitt, The method of configuration interaction, *in* "Modern Theoretical Chemistry" (H. F. Schaefer, Ed.), Plenum, New York, 1977.
9. R. K. Nesbet, *Rev. Mod. Phys.* **35** (1963), 552.
10. C. F. Bender, *J. Computational Phys.* **9** (1972), 547.
11. M. Yoshimine, "IBM Research Report RJ 555," San Jose, Calif., 1969. See also A. D. McLean, *in* "Proceedings of the Conference on Potential Energy Surfaces in Chemistry" (W. A. Lester, Ed.), IBM Research Report RA 18, San Jose, Calif., 1971.
12. P. Pendergast and W. H. Fink, *J. Computational Phys.* **14** (1974), 286.
13. G. H. F. Diercksen, *Theor. Chim. Acta* **33** (1974), 1.
14. H. F. Schaefer, Ph. D. Thesis, Stanford University, April 1969.
15. See, for example, C. F. Bender, S. V. O'Neil, P. K. Pearson, and H. F. Schaefer, *Science* **176** (1972), 1416.
16. For example, using the type of CI described by the first-order wavefunction: H. F. Schaefer, *J. Chem. Phys.* **54** (1971), 2207.
17. S. F. Boys, G. B. Cook, C. M. Reeves, and I. Shavitt, *Nature (London)* **178** (1956), 1207. For our first experience in the use of individual bits to store Slater determinants, see S. V. O'Neil, P. K. Pearson, and H. F. Schaefer, *Chem. Phys. Lett.* **10** (1971), 404.
18. M. Yoshimine, *J. Computational Phys.* **11** (1973), 449.
19. R. K. Nesbet, *J. Chem. Phys.* **43** (1965), 311.
20. I. Shavitt, *J. Computational Phys.* **6** (1970), 124.
21. I. Shavitt, C. F. Bender, A. Pipano, and R. P. Hosteny, *J. Computational Phys.* **11** (1973), 90.
22. E. R. Davidson, *J. Computational Phys.* **17** (1975), 87.
23. R. P. Hosteny, R. R. Gilman, T. H. Dunning, A. Pipano, and I. Shavitt, *Chem. Phys. Lett.* **7** (1970), 325.
24. B. J. Rosenberg and I. Shavitt, *J. Chem. Phys.* **63** (1975), 2162.

Received: December 8, 1976; revised: February 1, 1977

Robert R. Lucchese, Bernard R. Brooks,
James H. Meadows, William C. Swope,
and Henry F. Schaefer III

Department of Chemistry,
University of California,
Berkeley, California 94720