ELSEVIER

Journal of Computational and Applied Mathematics 149 (2002) 351–357

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

# A parallel algorithm for generating molecular integrals over MO basis sets

Kazuto Nakata[a], Tadashi Murase[a], Toshihiro Sakuma[b], Toshikazu Takada[c,*]

[a] *VALWAY Technology Center, NEC Soft, Ltd., 1-18-6 Shinkiba, Tokyo 136-8608 Japan*
[b] *Platform Software Division, Scientific Platform Department, NEC Informatec Systems, Ltd., Tsukuba, 305-8501, Japan*
[c] *Fundamental Research Laboratories, NEC Corporation, 34 Miyukigaoka, Tsukuba 305-8501, Japan*

Received 16 October 2001; received in revised form 6 December 2001

## Abstract

In the post Hartree–Fock theories such as multi-configuration self consistent field and configuration interaction, two electron integral transformation to molecular orbital sets is the most time consuming process for large-scale calculations. Parallelization is key to minimize the computer time for it. Then, a parallel integral-driven algorithm is presented for the integral transformation.

## 1. Introduction

In the ab initio molecular orbital (MO) theories, the multi-configuration self consistent field (MC-SCF) and configuration interaction (CI) methods are well established as approaches to take electron correlations into account [10]. From the computational point of view, the most time consuming procedure in these theoretical frameworks is to transform two electron integrals from atomic orbital (AO) to MO basis sets. This transformation is inevitable, since their energy expressions are given in terms of these integrals. In this paper, a parallel integral-driven algorithm to transform two electron integrals from AOs to MOs on parallel computers such as PC clusters is presented. In the following sections, the conventional transformation algorithm is reviewed first and then the new method is introduced along with some benchmark results.

[*] Corresponding author.
*E-mail address:* takada@frl.cl.nec.co.jp (T. Takada).

PII: S0377-0427(02)00541-1

## 2. Conventional integral transformation scheme

Two electron integrals over AOs, $\chi$, are defined as

$$(rs|tu)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\chi_r(r_1)\chi_s(r_1)\frac{1}{|r_1-r_2|}\chi_t(r_2)\chi_u(r_2)\,\mathrm{d}r_1\,\mathrm{d}r_2. \tag{1}$$

By using the relations between AOs and MOs, that is,

$$\phi_a=\sum_r^N c_{ra}\chi_r, \tag{2}$$

the two electron integrals over MOs are simply generated to be

$$(ab|cd)=\sum_r^N\sum_s^N\sum_t^N\sum_u^N c_{ra}c_{sb}c_{tc}c_{ud}(rs|tu), \tag{3}$$

where $N$ is the number of the atomic orbitals. With a primitive method having eight do loops for the indices of both AOs and MOs, the number of the numerical operations is $N^8$. But, a very efficient algorithm has been invented for the transformation, consisting of the four steps [10]. This idea has been implemented in major ab initio MO codes like Gaussian98 [3], GAMESS [2] and so on. The first step, for example, is written in Fig. 1 and similar procedures are repeated three more times for transforming from $t$ to $c$ and so on. In this algorithm, the number of the numerical operations is $4N^5$ in total. Conventionally, two electron integrals over AOs are stored on disks and sorted in proper sequences for transformations. But, as molecular sizes to be calculated are growing, the set of integrals to be stored is quickly beyond the storage capacity. On the other hand, in the Hartree–Fock scheme, the direct method [1] became common, in which these two electron integrals are evaluated repeatedly in SCF cycles and each of them is accumulated instantly into the Fock matrix by multiplying density matrix elements. However, since the energy expressions in MCSCF and CI are given in terms of the molecular integrals over MOs, the transformation is requisite.

Simple application of the conventional algorithm on parallel computers causes the following problems, that is,

(1) Since the two electron integrals over AOs are independent of one another, evaluations of these integrals are easily parallelized. But, all the integrals are needed for obtaining one single integral over MOs, causing serious bottleneck by heavy wall-to-wall transmissions among processors.

```
do for r, s and t
 do u
  do d
   (rs | td) = (rs | td) + $c_{ud}$ (rs | tu)
  end do
 end do
end do
```

Fig. 1. First step of the conventional algorithm for transforming two electron integrals from AOs to MOs.

(2) To avoid such bottleneck, all the integrals over AOs must be calculated on each processor, which loses the benefit of the parallel computations of two electron integrals.

This conflict has been preventing usage of parallel computers for the post Hartree–Fock calculations. However, as is well recognized in HPC communities, parallelization is now key for large-scale simulations in any research field, no matter what kind of computers, vector or scalar parallel, are used. From this background, as seen below, a new transformation algorithm is invented here to overcome the difficulty, which makes it possible to carry out large-scale calculations even with the post Hartree–Fock theories on parallel computers.

## 3. A parallel algorithm for integral transformations

Fig. 2 schematically illustrates the new transformation algorithm. The indices of AO's, that is, *r* and *s* are assigned to a processor, which are denoted as *R* and *S* in the figure. That is, a pair of *R* and *S*, is distributed to a processor and all the combinations of *t* and *u* are considered there. After evaluating the set of the integrals, $(RS|\mathrm{allall})$, the transformation from *u* to *d* is carried out in the

```
< first step >
do t
  do u
    do d
      $(RS|td) = (RS|td) + c_{ud}(RS|tu)$

    end do
  end do
end do

< second step >
do d
  do t
    do c
      $(RS|cd) = (RS|cd) + c_{tc}(RS|td)$

    end do
  end do
end do

< third step >

do d
  do c
    do B
      $(RB|cd) = (RB|cd) + c_{SB}(RS|cd)$

    end do
  end do
end do

< fourth step >
do d
  do c
    do B
      do A
        $(AB|cd) = (AB|cd) + c_{RA}(RB|td)$

      end do
    end do
  end do
end do
```

Fig. 2. Parallel algorithm for transforming two electron integrals from AOs to MOs.

Table 1
Numerical operation in conventional and present algorithms

| | Conventional | | Present[a] | |
|---|---|---|---|---|
| | Operations | Memory | Operations | Memory |
| First | $N^4n$ | $N^4+N^3n$ | $N^2n$ | $N^2+Nn$ |
| Second | $N^3n^2$ | $N^3n+N^2n^2$ | $Nn^2$ | $Nn+N^2$ |
| Third | $N^2n^3$ | $N^2n^2+Nn^3$ | $n^3$ | $n^2+n^3$ |
| Fourth | $Nn^4$ | $Nn^3+n^4$ | $n^4$ | $n^3+n^4$ |
| Gathering | — | — | $N^2n^4$ | — |

[a] For one processor.

first step. Then, the second step is performed. In the third step, the orbital, $S$, is transformed to all the MOs denoted by $B$. Note that only one component of $S$ is involved in the $(RB|cd)$ integrals. Similarly, the transformation from $R$ to $A$ is carried out as the fourth step, where another do loop for $B$ is needed. When all the transformations have been completed, the set of the integrals, $(AB|\text{allall})$ exists on every processor, which has only the contributions from the assigned orbitals of $R$ and $S$. Therefore, these integrals must be gathered on one processor to obtain the correct two electron integrals $(ab|cd)$ over the molecular orbitals.

Another integral-driven transformation scheme has been presented in [9], in which $N^6$ operations are required, since pair-wise transformation for the overlap charges is carried out. For the direct CI matrix generation, a similar parallelization algorithm using only the index of $r$ has been proposed by [6].

An MCSCF framework called complete active space SCF (CASSCF) has been proposed in [8], which gives a simple energy expression in the MCSCF methodology and consequently a smaller set of molecular orbitals is needed for the transformed integrals. Let $n$ be the number of the CAS orbitals, which is at most nearly 10 in the recent calculations and then $n \ll N$ holds for large basis sets. In Table 1, the number of numerical operations and working memory sizes are summarized along with the conventional technique. The total operations for the conventional are $nN^4+n^2N^3+n^3N^2+n^4N$, while the present requires $N^2(nN^2+n^2N+n^3+n^4)+n^4N^2$ operations. As a result, $n^4N(N-1)+n^4N^2$ operations are extra. However, the cost for these operations is certainly negligible in comparing with the first step, if $N$ is more than 200. The number of numerical operations for the first step is $10\times 200^4$, while the extra operations are $10^4\times(2\times 200^2-200)$, which is only 5% of the former. Since permutations between $r$ and $s$, and between $(rs)$ and $(tu)$ are no longer taken in the present algorithm, evaluation of the integrals over AOs is four times more than the conventional one.

It should be noticed that no transmissions among processors are required until all the transformations to obtain $(AB|\text{allall})$ integrals are completed and the number of two electron integrals to be calculated is still kept to be $N^4$ through the whole processors. Actually, it is a half of $N^4$. Therefore, it is concluded that the present algorithm solved the contradiction mentioned above.

This algorithm has been implemented in the CASSCF package of AMOSS-H13, which has been being developed by the NEC quantum chemistry group and is especially designed for large molecules like bio-related molecules.

Table 2
Elapsed time, speed up and parallelization ratio[a]

| Basis set | Processors | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 4 | 8 | 16 | 32 |
| MINI-4 (122 AO's) | | | | | | |
| Elapsed time (s) | 1890 | 950 | 495 | 252 | 139 | 76 |
| Speed up | 1 | 1.99 | 3.82 | 7.49 | 13.59 | 24.81 |
| Ratio (%) | — | 99.49 | 98.44 | 99.04 | 98.82 | 99.06 |
| MIDI-4 (224 AO's) | | | | | | |
| Elapsed time (s) | 4417 | 2189 | 1130 | 595 | 296 | 161 |
| Speed up | 1 | 2.02 | 3.91 | 7.42 | 14.93 | 27.45 |
| Ratio (%) | — | 100.88 | 99.22 | 98.89 | 99.52 | 99.47 |

[a] Estimated from Amdahl's law.

## 4. Benchmarks

The speed-up measured on a PC cluster is shown in Table 2. The molecule used is artemisin ($C_{15}H_{22}O_5$). The basis sets are MINI-4 (122 AOs) and MIDI-4 (224 AOs) proposed in [7]. The computer system for the benchmark consists of 33 Pentium III (933MHz) processors with local memory of 1GB each and the OS and communication library are RedHat Linux 6.2 and LAM/MPI 6.5.4. One processor works as the master and the rest are used to be slaves. That is, the master processor spawns the tasks for integral evaluations and transformations to the slaves. In the CASSCF theory, the MOs are divided into three subspaces, that is, inner (I) for doubly occupied space, active (A) for CAS and secondly (S) for virtual. Consequently, for the augmented Hessian scheme based on the Newton–Raphson numerical procedure [4,11] to achieve rapid convergences, the following integral types appear: (AA|AA), (IA|AA), (SA|AA), (II|AA), (IS|AA), (SS|AA), (IA|IA), (IA|SA), (SA|SA), (II|SA), (IS|IA), (IS|SA), (SS|IA), (SS|II), (SI|SI). The elapsed times in Table 2 are for from the first to third steps shown in Fig. 1, while the integral evaluation itself is included. The fourth step is located deep inside a program unit with other functions for the performance efficiency and the separated measurement is rather difficult on the present code. From Table 2, it is clearly seen that the present algorithm works quite well on the parallel computer system. The parallelization ratios are estimated from Amdahl's law [5] using the measured speed up.

To show applicability of the present method, the transformations for the following molecules are carried out and the results are shown in Table 3. Up to now, the numbers of the AOs for the post-Hartree–Fock calculations are limited around several hundreds and therefore the data are showing that the present algorithm is promising.

## 5. Conclusions

Molecular simulations are expected to be key for the research fields like bioinformatics and nano-technology, since the most fundamental procedures are all chemical phenomena, which are

Table 3
Memory sizes and elapse time on PC cluster (32 processors)

| Molecules | Numbers of AOs, I, A, S | Local memory (MB)[a] | Elapse time (min) |
|---|---|---|---|
| $C_{15}H_{22}O_5$ | 122 (MINI-4), 72, 8, 42 | 2 | 1.3 |
| | 224 (MIDI-4), 72, 8, 144 | 5 | 2.7 |
| $C_{31}H_{43}NO_3$ | 218 (MINI-4), 126, 8, 84 | 6 | 2.6 |
| | 401 (MIDI-4), 126, 8, 267 | 18 | 12.3 |
| $C_{30}H_{52}O_{26}$ | 332 (MINI-4), 216, 8, 108 | 14 | 17.5 |
| | 608 (MIDI-4), 216, 8, 384 | 51 | 96.2 |
| $C_{86}H_{26}$ | 456 (MINI-4), 267, 8, 181 | 29 | 52.8 |
| | 826 (MIDI-4), 267, 8, 551 | 105 | 312.3 |
| $C_{72}H_{76}N_8O_{12}Mg_2$ | 548 (MINI-4), 338, 8, 202 | 45 | 119.2 |
| | 994 (MIDI-4), 338, 8, 648 | 173 | 834.3 |

[a] For one processor and the estimations in Table 1 is for extreme cases that every different pair of $r$ and $s$ is assigned to different processors.

to be theoretically predicted by quantum mechanics. The ab initio MO theories are well established as numerical frameworks to solve the Schrodinger equation. The Hartree–Fock theory has been applied to large biological molecules, by taking advantage of its theoretical simplicity. But, as is seen in this article, applications of the post-Hartree–Fock theories have been impractical for the heavy integral transformations. To describe chemical reactions, there are some cases in which mixing of a few configuration state functions are essential to give proper dissociation limits or correct activation energies. In these calculations that may happen in biological molecules, the CASSCF is definitely useful.

A new parallel algorithm is presented to transform two electron integrals from AOs to MOs for the post-Hartree–Fock calculations. There are no transmissions needed among the processors until the half-transformed integrals are all generated. Gathering these integrals is completed at the end once and the data size to be transmitted is only in the order of $n^4$. Furthermore, the number of two electron integrals to be calculated is kept to be still an order of $N^4$ through the entire cluster system. From these advantages of the present algorithm, much larger calculations than present become practical soon on PC clusters.

The results of the benchmarks demonstrate high scalabilities of up to 32 processors. Since the parallelization is accomplished by the indices of the overlap charges denoted by $r$ and $s$, the number of processor to be used reaches up to the range of millions for calculations even discussed here and benchmarks with more processors are necessary for testing applicability of the algorithm to such computers.

## Acknowledgements

This project has been supported by “Research and Development for Applying Advanced Computational Science and Technology” of Japan Science and Technology Corporation.

## References

[1] J. Almlof, K. Fraegri Jr., K. Korsell, J. Comput. Chem. 3 (1982) 385.

[2] GAMESS: http://www.msg.ameslab.gov/GAMESS/GAMESS.html.

[3] Gaussian 98: http://www.gaussian.com/.

[4] B.H. Lengsfield, J. Chem. Phys. 73 (1980) 382.

[5] T.G. Mattson, in: T.G. Mattson (Ed.), Parallel Computing in Computational Chemistry, ACS Symposium Series 592, American Chemical Society, Washington, DC, 1995, pp. 1–15.

[6] Y. Mochizuki, N. Nishi, Y. Hirahara, T. Takada, Theor. Chem. Acta 93 (1996) 211.

[7] J. Andzelm, M. Kłobukowski, E. Radzio-Andzelm, Y. Sakai, H. Tatewaki, Physical Sciences Data 16, in: S. Huzinaga (Ed.), Gaussian Basis Sets for Molecular Calculations, Elsevier, Amsterdam, 1984.

[8] B.O. Roos, P.R. Tayler, P.E.M. Seigbahn, Chem. Phys. 48 (1980) 157.

[9] P.R. Taylor, Internat. J. Quantum Chem. 31 (1987) 521.

[10] A.C. Wahl, G. Das, I, Shavitt, in: H.F. Schaefer III (Ed.), Methods of Electronic Structure Theory, Plenum Press, New York, 1977, pp. 51–78, 189–275.

[11] D.R. Yarkony, Chem. Phys. Lett. 77 (1981) 634.