Theor Chem Acc (2004) 112: 442–452
DOI 10.1007/s00214-004-0602-3

Theoretical
Chemistry Accounts

Regular article

A parallelized integral-direct second-order Møller–Plesset perturbation theory method with a fragment molecular orbital scheme

Yuji Mochizuki[1], **Tatsuya Nakano**[2], **Shigeru Koikegami**[1], **Souichirou Tanimori**[1], **Yukinobu Abe**[1], **Umpei Nagashima**[3], **Kazuo Kitaura**[4]

[1] Advancesoft and Institute for Industrial Science, Center for Collaborative Research, The University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo 153-8904, Japan
[2] Division of Safety, Information on Drug, Food, and Chemicals, National Institute of Health Sciences, 1-18-1 Kamiyoga, Setagaya-ku, Tokyo 158-8501, Japan
[3] Grid Technology Research Center, National Institute for Advanced Industrial Science and Technology, 6-9-3 Higashi-ueno, Taito-ku, Tokyo 110-0015, Japan
[4] Research Institute for Computational Sciences, National Institute for Advanced Industrial Science and Technology, Central 2, Tsukuba-si, Ibaraki 305-8568, Japan

Received: 8 March 2004 / Accepted: 25 May 2004 / Published online: 14 October 2004

Abstract. We propose a parallelized integral-direct algorithm of the second-order Møller–Plesset perturbation theory (MP2) as a size-consistent correlated method. The algorithm is a modification of the recipe by Mochizuki et al. [(1996) *Theor Chim Acta* 93:211]. There is no need to communicate the bulky data of integrals across worker processes, keeping the formal fifth-power dependence on the number of basis functions. A multiple integral screening procedure is incorporated to reduce the operation costs effectively. An approximate MP2 density matrix can also be directly calculated through the integral contraction with orbital energies. We implement the MP2 code by accepting Kitaura's fragment molecular orbital (FMO) scheme as in the program ABINIT-MP developed by Nakano et al. [(2002) *Chem Phys Lett* 351:475]. The error in the FMO–MP2 energies is found to be within the order of the chemical accuracy. Timing and parallel acceleration results are shown for test molecules.

Keywords: Fragment molecular orbital – Second-order Møller – Plesset perturbation theory – Parallelism – Integral transformation – Density matrix

Introduction

Among the conventional post-Hartree–Fock (HF) methods, the second-order Møller–Plesset perturbation theory (MP2) [1] is the simplest and the computationally most economical treatment to introduce the electron correlation size-consistently [2, 3, 4]. The MP2 calculations are known to be safer or describe better to weak interactions of hydrogen-bond (HB) and van der Waals (vdW) types than density functional theory (DFT), which has recently become the de facto standard of correlated methods. The computational cost of DFT is as economical as that of HF calculations [5]. Although MP2 requires more computational cost than DFT, the method holds advantage over DFT in providing a safe choice or guideline for correlated calculations of a variety of biochemical molecules in which interactions of HB and vdW are crucial. For example, a comparison between MP2 and DFT results for nucleic acid models was reported in Refs. [6, 7, 8].

The MP2 correlation energy for a closed-shell molecule is given simply by the contraction of two-electron molecular orbital (MO) integrals of the (ia, jb) exchange type (Mulliken notation with i, j for doubly occupied orbitals and a, b for virtual orbitals)

$$E_{\mathrm{MP2}} = \sum_{ijab} \frac{(ia,jb)[2(ia,jb) - (ib,ja)]}{\varepsilon_i + \varepsilon_j - \varepsilon_a - \varepsilon_b}, \tag{1}$$

where $\varepsilon_i, \varepsilon_j, \varepsilon_a$, and ε_b are the HF canonical orbital energies. The molecular integral list (ia, jb) is prepared by the four-index transformation from the atomic orbital (AO) basis integral list (pq, rs)

$$(ia,jb) = \sum_{pqrs} C_{pi} C_{qa} C_{rj} C_{sb} (pq,rs), \tag{2}$$

where the matrix $\mathbf{C}$ is the AO–MO expansion coefficients. It is well known that the process of Eq. (2) is divided into the quarter transformations as follows [9]:

$$(iq,rs) = \sum_{p} C_{pi} (pq,rs), \tag{3}$$

Correspondence to: Y. Mochizuki
e-mail: fullmoon@fsis.iis.u-tokyo.ac.jp

$$(ia, rs) = \sum_q C_{qa}(iq, rs), \tag{4}$$

$$(ia, js) = \sum_r C_{rj}(ia, rs), \tag{5}$$

$$(ia, jb) = \sum_s C_{sb}(ia, js). \tag{6}$$

This enables MP2 to be processed at the cost of N^5, where N is the number of basis functions. Other correlated methods, for example, coupled-cluster singles and doubles [3], scale as N^6 or more, although they provide more accurate results than MP2. The MP2 results can thus be a good compromise in studying large molecules to which more extensive methods cannot be applied.

The AO–MO transformation of two-electron integrals is the central issue to accelerate the MP2 calculations, regardless of canonical orbitals or localized orbitals. The so-called local MP2 (LMP2), pioneered by Pulay and Saebø [10], is based on a different energy expression from Eq. (1) by the use of localized orbitals. The LMP2 technique cuts the computational costs through neglecting spatially distant interactions [11, 12, 13, 14, 15]. The parallelization has been considered as the main route to accelerate the processing of integral transformation in the MP2 scheme with the canonical orbitals [16, 17, 18, 19, 20, 21, 22, 23, 24].

In this paper, we report a new parallelized integral-direct transformation algorithm by modifying a recipe by Mochizuki et al. [25]. The implementation as the MP2 of Eq. (1) is made by accepting Kitaura's fragment MO (FMO) method, which utilizes the potential locality of fragmental units in large molecules like polymers and clusters and makes the calculations of such systems possible with reasonable accuracy [26, 27, 28, 29, 30, 31]. The canonical orbital-based calculations are carried out independently for each fragment X (called "monomer") and fragment-pair XY ("dimer"). In this two-body approximation of the FMO, the total energy and the density are given, respectively, by

$$E = \sum_{X>Y} E^{[XY]} - (M-2)\sum_X E^{[X]} \tag{7}$$

and

$$\rho(\mathbf{r}) = \sum_{X>Y} \rho(\mathbf{r})^{[XY]} - (M-2)\sum_X \rho(\mathbf{r})^{[X]}, \tag{8}$$

where M means the number of fragments. The total cost of the calculations scales as only M^2. Currently, there are three programs by which FMO calculations are available: GAMESS, extended by Fedorov and coworkers [32, 33, 34] to include the three-body approximation, NWChem, modified by Sekino et al. [35, 36] for DFT, and ABINIT-MP, originally developed by Nakano et al. [30] for HF calculations of peptides with the parallelization of the message-passing interface (MPI) [37]. As already mentioned, the HB and the vdW interactions are very important in biochemical systems. The MP2 implementation has been done for the ABINIT-MP program. A flexible combination of the upper fragment monomer/dimer layer and the lower AO indices layer is available in the parallelization of HF with ABINIT-MP. The parallelization of MP2 does likewise. The results of the test calculations will show promising results of parallel acceleration. The MP2 density matrix is not of variational-type but of response-type [4]. We have also implemented an integral-direct generation of the MP2 density matrix by approximating the orbital Hessian of the response equation. One will see that the ionic character in the HF description is relaxed through the MP2 correlation.

Transformation algorithm for MP2

We first discuss the parallelized MP2 algorithm itself before discussing the incorporation into the ABINIT-MP/FMO scheme [30]. There are a variety of parallel MP2 recipes, depending on the target platforms [16, 17, 18, 19, 20, 21, 22, 23, 24]. For example, Refs. [19, 20, 21] assumed the usage of massively parallel type machines equipping high-performance networks. Nowadays, clustered personal computers (PCs) with commodity processors, for example, Intel's Pentium series, have achieved a popularity as an economical in-house platform for parallel computing. The usability of PCs was demonstrated in Ref. [22]. Additionally, the memory area per processor is being extended to a few gigabytes. However, the disk input/output of a PC is much less efficient in speed than the modern processor, and the performance of the cluster network is limited especially in worker-to-worker cross communications. When one considers not only massively parallel machines but also general PC clusters as computing platforms, integral-direct processing (which was originally proposed by Almlöf et al. [38] for the Fock matrix construction in HF calculations) or in-core processing would be desirable. Furthermore, the worker-to-worker communications should be minimized or avoided. In this direction, Mochizuki et al. [25] have proposed a loop structure of parallelized integral-direct transformation for general configuration interaction (CI) calculations, where there is no communication of intermediately transformed integrals across the worker processes and where the σ-vector elements are updated by three-quarter transformed integrals multiplied by AO–MO coefficients in each worker process. The formal cost for the transformation is N^5. Note that Schütz et al. [39] developed the matrix-formulated integral-direct calculations of various correlated treatments including the internally contracted CI. Nakata et al. recently realized a parallelized integral-direct version of the complete-active-space self-consistent-field (CASSCF) [3] code with minimizing memory size for buffers and avoiding worker-to-worker communications needed for the transformation on PC cluster environments [40]. Unfortunately, their transformation algorithm is inefficient for MP2 because it scales as N^6. Taylor [41] also discussed the integral-direct post-HF calculations but the scheme was of the N^6 type. Hereafter we discuss our algorithm. We start with the recipe proposed by Mochizuki et al. in Ref. [25], by paying special attention to the fact that the MP2 energy expression, Eq. (1), requires the fully transformed integrals in contrast to the case for σ vectors. Our MP2 parallelism is of the message-passing type [37], unless otherwise stated.

Since the symmetry of biochemical molecules is usually C_1, no symmetry is considered at all. Italic capitals O and V are defined as the number of occupied orbitals and the number of virtual orbitals, respectively. The relation $O \ll V < N$ is expected. The loop structure for the integral-direct MP2 energy calculations is presented in Fig. 1. The size of the associated buffer and the total (regardless of parallelization) operational cost for each quarter transformation are compiled in Table 1. The outermost loop in Ref. [25] was assigned to the MO label which is blocked to adjust the available memory

```
Do initialization for the MP2 energy calculation.
Determine $O_{[B]}$ and $I_{O_{[B]}}$ depending on $O$ and available memory.
Parallelizable loop over $I_{O_{[B]}}$
  Clear WIAJB for the fourth transformation.
  Set the range of $i_{[B]}$ for the current $I_{O_{[B]}}$ block.
  Main parallelizable loop over $s_{[B]}$
    Clear WIAJS for the third transformation.
    Loop over $r$ ! No canonical relation between $r$ and $s_{[B]}$ is used.
      Clear WIQRS for the first transformation.
      Get $(pq, rs_{[B]})$ with the relation of $p \geq q$ depending on direct or in-core modes.
      Loop over $p$
        Loop over $q \leq p$
          If $[|(pq, rs_{[B]})| \geq T_\alpha]$ then
            If $[p = q]$ then
              Use DAXPY for accumulation of $(pp, rs_{[B]}) * C_{pi_{[B]}}$ to WIQRS$(p, i_{[B]})$.
            Else
              Use DAXPY for accumulation of $(pq, rs_{[B]}) * C_{pi_{[B]}}$ to WIQRS$(q, i_{[B]})$.
              Use DAXPY for accumulation of $(pq, rs_{[B]}) * C_{qi_{[B]}}$ to WIQRS$(p, i_{[B]})$.
            Endif ! $p \geq q$
          Endif ! screening by $T_\alpha$
        End of loop over $q$
      End of loop over $p$
      Loop over $i_{[B]}$
        Loop over $a$
          Use DDOT for $(i_{[B]}a, rs_{[B]})$ as inner product between WIQRS$(q, i_{[B]})$ and $C_{qa}$.
          If $[|(i_{[B]}a, rs_{[B]})| \geq T_\beta]$ then
            Use DAXPY for accumulation of $(i_{[B]}a, rs_{[B]}) * C_{rj}$ to WIAJS$(j, a, i_{[B]})$.
          Endif ! screening by $T_\beta$
        End of loop over $a$
      End of loop over $i_{[B]}$
    End of loop over $r$ ! The third transformation is completed here.
    Loop over $i_{[B]}$
      Loop over $a$
        Loop over $j$
          If $[|(i_{[B]}a, js_{[B]})| \geq T_\gamma]$ then
            Use DAXPY for accumulation of $(i_{[B]}a, js_{[B]}) * C_{s_{[B]}b}$ to WIAJB$(b, j, a, i_{[B]})$.
          Endif ! screening by $T_\gamma$
        End of loop over $j$
      End of loop over $a$
    End of loop over $i_{[B]}$
  End of main parallelizable loop over $s_{[B]}$ ! The "all-reduce" operation is needed.
  Calculate the partial MP2 energy for $(i_{[B]}a, jb)$.
End of parallelizable loop over $I_{O_{[B]}}$
Finalize the MP2 result.
```

Fig. 1. Loop structure of the algorithm for the second-order Møller–Plesset perturbation theory energy. The *DAXPY* and the *DDOT* operations [42] are indicated in each quarter transformation step. The triple screening for intermediate integrals is used

Table 1. Required memory size for buffer and total operational cost for each quarter transformation step in the algorithm for the second-order Møller–Plesset perturbation theory (*MP2*) energy

Step	Transformation	Memory	Cost
First (1/4)	$(pq, rs_{[B]}) \rightarrow (i_{[B]}q, rs_{[B]})$	WIQRS $(N, O_{[B]})$	$\alpha ON^3(N+1)/2$
Second (2/4)	$(i_{[B]}q, rs_{[B]}) \rightarrow (i_{[B]}a, rs_{[B]})$	Not needed	OVN^3
Third (3/4)	$(i_{[B]}a, rs_{[B]}) \rightarrow (i_{[B]}a, js_{[B]})$	WIAJS$(O, V, O_{[B]})$	$\beta O^2 VN^2$
Fourth (4/4)	$(i_{[B]}a, js_{[B]}) \rightarrow (i_{[B]}a, jb_{s_{[B]}})$	WIAJB$(V, O, V, O_{[B]})$	$\gamma O^2 V^2 N$

(denoted q_g). In the present algorithm, the outermost part corresponds to the block of occupied orbitals. The block size, the block (or path), and the blocked orbital label are denoted as $O_{[B]}$, $I_{O_{[B]}}$, and $i_{[B]}$, respectively. The O-blocking was also used in Refs. [19, 21]. If the number of block $I_{O_{[B]}}$ is more than 1 (or $O \neq O_{[B]}$), the parallel processes could be invoked for the blocks. Without the parallelization, the integral generation will be repeated if in-core buffering is not available. The index s of the AO integral (pq, rs) is used as the main parameter of the parallelization. We denote it as $s_{[B]}$.

When the loop of $s_{[B]}$ is parallelized, the list of $(i_{[B]}a, jb_{s_{[B]}})$ would be created in each worker process in which the transformation over the index b is not completed. To form the complete list of $(i_{[B]}a, jb)$, the so-called all-reduce operation with a barrier must be carried out, and it associates with the worker-to-master one-sided data transmissions of $O_{[B]}OV^2$ quantity. This all-reduce treatment is similar to that in Ref. [40], where the parallel distribution for partial transformations was performed by a combination of two AO indices. One might feel that the handling of the $(i_{[B]}a, jb)$ buffer, WIAJB, could be demanding; however, the situation would not be serious in most cases of FMO–MP2 calculations, as will be addressed later in this section. Once a chunk of $(i_{[B]}a, jb)$ is available, the partial MP2 energy is evaluated by the DDOT function of basic linear algebra subroutines (BLAS) [42].

Now, we discuss each step of the quarter transformation in Fig. 1. Using the screened integrals $(pq, rs_{[B]})$ by the threshold, T_α, the first transformation of $p \rightarrow i_{[B]}$ is performed, where the corresponding pq list is just generated in the integral-direct mode or is copied from AO integral buffers in the in-core mode (when the list is retained after HF calculations with memory-rich environments). The canonical relation of $p \geq q$ is utilized here. To parallelize the transformation processing without the worker-to-worker communications, the usual rs and pq/rs canonical relations are not usable as in Refs. [25, 40]. The DAXPY routine [42] is used for the operation about $i_{[B]}$, where the buffer is WIQRS, whose size is $NO_{[B]}$. The cost is $\alpha ON^3(N+1)/2$, where α is the survival ratio of the screening of AO integrals. The value of α is highly dependent on the sizes and the shapes of the target molecules and the basis sets.

Under the loop nest of $i_{[B]}$ and of a, the second quarter transformation of $q \rightarrow a$ is done with an inner product operation with DDOT [42] about q, where the cost is OVN^3. The resulting $(i_{[B]}a, rs_{[B]})$ value is screened by a proper threshold, T_β, before the third quarter transformation. This screening reduces the operations costs with the survival ratio of β. The styles of the first and second quarter transformations and the associated screenings are essentially the same as the conventional direct-access file-based transformation algorithm by Yamamoto et al. [43]. The third quarter transformation of $r \rightarrow j$ is performed by DAXPY about j, where the size of the buffer, WIAJS which holds $(i_{[B]}a, js_{[B]})$, is $OVO_{[B]}$. When the loop of the index r is finished, this third transformation would be completed. The total cost of the process is $\beta O^2 VN^2$.

The fourth quarter transformation concerns $s_{[B]} \rightarrow b$, where the all-reduce operation is necessary [40]. The DAXPY routine would be called in triply nested loops for the accumulation of WIAJB about b, when the element of $(i_{[B]}a, js_{[B]})$ in WIAJS survives the screening. The cost is given by $\gamma O^2 V^2 N$, with γ as the survival ratio for the third screening with T_γ.

The energy calculation of the target system consists of the respective calculations of monomer X and dimer XY in the two-body approximation of the FMO scheme [26, 27, 28, 29, 30, 31]. Equation (7) is used to sum up the MP2 correlation energies of fragments, as in the case of HF calculations, based on the size-consistency and the unitary invariance of MP2 [2, 3, 4]. Since the grand calculations of monomers and dimers are independently parallelizable in the ABINIT-MP system [30], there are in total three layers of possible parallelism for the FMO–MP2 energy calculations, as summarized in Table 2. The MP2 processing follows when the HF procedure, whose Fock matrix construction is of course parallelized, is completed for the given fragment. Here, we would address the size of $(i_{[B]}a, jb_{s_{[B]}})$ which is held in the WIAJB buffer of worker processes. In the calculations of peptides using ABINIT-MP, a monomer includes two amino acid residues when one gives priority to the overall accuracy, and thus the dimer consists of four amino acid residues. Even with valence double-zeta (VDZ) basis sets, for example, 6-31G [44], the number of AO basis functions, N, for the dimer can potentially reach about 600, depending on the combination of

Table 2. Layers of possible parallelization for the fragment molecular orbital (*FMO*)–MP2 energy calculation

Layer	Target	Index	Quantity	Barrier
1	Fragment	X or XY	$E^{[X]}$[a] or $E^{[XY]}$[a]	No
2	Occupied orbital block[b]	$I_{O_{[B]}}$	$E_{\mathrm{MP2}}(I_{O_{[B]}})$	No
3	Atomic orbital basis function	$s_{[B]}$	$(i_{[B]}a, jb_{s_{[B]}})$	Yes

[a] Consisting of Hartfree-Fock (HF) energy and MP2 correlation energy. The preceding HF calculation is also parallelized for the Fock matrix construction in ABINIT-MP [30]

[b] For the case of $O \neq O_{[B]}$

amino acids (glycine to tryptophan). If one considers a demanding dimer whose $N^{[XY]}$, $O^{[XY]}$, and $V^{[XY]}$ are 600, 200, and 400, respectively, it would lead to a WIAJB buffer size of 256 MB if $i_{[\mathrm{B}]}$ is supposed to be 1. Even though a memory area of a few gigabytes is available per worker process, we obviously need the parallelization by $I_{O_{[\mathrm{B}]}}$ for such a case. Very recently, Fedorov and Kitaura reported the FMO three-body approximation at the HF [33] and DFT [34] levels with their modified GAMESS code. They showed that the accuracy using the three-body approximation with one amino acid residue per monomer is better than that using the two-body approximation with two amino acid residues per monomer. A similar situation would be expected for the MP2 calculations. If we employ the three-body approximation, the number of amino acid residues even for a trimer could be retained as 3. This could help the handling of the WIAJB buffer to a certain extent. The introduction of the three-body approximation into ABINIT-MP should be a pending issue.

The structure of our direct MP2 loops could be used backwards for the integral-driven general CI calculations [25]. We found that the third screening with T_γ for (ia, js) was not so effective in the canonical MP2 case owing to the inherent delocalized nature of canonical MOs, relative to the first screening for (pq, rs) and the second screening for (ia, rs). However, when MO sets with the locality of orbital domains or fragmental units are used, the third screening of integrals should also be effective in processing the contribution of integrals after third quarter transformations have been added to the σ vector elements, as in Ref. [25]. The partially constructed σ vector in each worker process should be gathered with the all-reduce operation. As the first step in this direction, work to develop CI singles calculations [44, 45] for the excited-states survey within the FMO scheme is in progress.

MP2 density

According to Eq. (8), the whole electron density by FMO–MP2 is calculated using fragment densities of monomers and dimers. The issue is how to evaluate the MP2 density matrix for each monomer or dimer. The MP2 wavefunction is not of variational type [4]. The response density matrix elements, P_{ai}, which correspond to the orbital relaxation by introducing electron correlation, should be evaluated by solving the so-called coupled perturbed HF (CPHF) equation [46, 47, 48, 49, 50, 51, 52]. The CPHF equation is formally written as follows

$$(\varepsilon_i - \varepsilon_a)P_{ai} + \sum_{bj} G_{ia,jb}P_{bj} = L_{ai}, \tag{9}$$

$$G_{ia,jb} = 4(ia, jb) - (ij, ab) - (ib, ja), \tag{10}$$

where L_{ai} corresponds to the MP2 Lagrangian which is constructed by a series of contractions of (ia, bc) and (ia, jk) type integrals adding to the usual exchange integrals (ia, jb) for the MP2 energy. The terms of $G_{ia,jb}$ are the integral part of the orbital Hessian. The amplitudes are expected to be small relative to $(\varepsilon_i - \varepsilon_a)$ as long as the HF wavefunction is good enough for the zeroth-order description [3]. In order to avoid the costly process of solving Eq. (9), we thus neglect the elements of $G_{ia,jb}$. We further neglect the similar terms of $G_{ia,jk}$ and $G_{ia,bc}$ in the MP2 Lagrangian, L_{ai}. As a result, the approximated expression for P_{ai} is given by a simple contraction

$$P_{ai} = \frac{1}{\varepsilon_i - \varepsilon_a}[\sum_{jbc} A_{ij}^{bc}(jc, ab) - \sum_{jkb} A_{jk}^{ab}(jb, ik)], \tag{11}$$

$$A_{ij}^{ab} = \frac{2[2(ia, jb) - (ib, ja)]}{\varepsilon_i + \varepsilon_j - \varepsilon_a - \varepsilon_b}, \tag{12}$$

where the matrix $\mathbf{A}$ is the MP2 amplitude with a factor of 2. To form the total density matrix, there are the other contributions of the occupied–occupied block P_{ij} and the virtual–virtual block P_{ab}

$$P_{ij} = 2\delta_{ij} - \sum_{kab} \frac{A_{jk}^{ab}(ia, kb)}{\varepsilon_i + \varepsilon_k - \varepsilon_a - \varepsilon_b}, \tag{13}$$

$$P_{ab} = \sum_{ijc} \frac{A_{ij}^{ac}(ib, jc)}{\varepsilon_i + \varepsilon_j - \varepsilon_b - \varepsilon_c}. \tag{14}$$

The first term in Eq. (13) is the closed-shell occupations of HF. The final expressions of Eqs. (11), (12), (13), and (14) are equivalent to those given by Jensen et al. [53].

When both the MP2 energy and the density are calculated, the direct integral transformation should generate not only (ia, jb) type but also (ia, jk) and (ia, bc) types, as shown in Eq. (11). The third and fourth quarter transformation steps in Fig. 1 should be modified to do it, where the canonical relation of both $j \geq k$ for $(i_{[\mathrm{B}]}a, jk_{s_{[\mathrm{B}]}})$ and $b \geq c$ for $(i_{[\mathrm{B}]}a, bc_{s_{[\mathrm{B}]}})$ reduces each corresponding buffer size. The amount of data sent increases substantially for the density calculation, relative to the only-energy case. After the last quarter transformation is completed through the all-reduce of corresponding buffers, the contribution to the density matrix elements is calculated by the contractions. The DDOT function [42] is used in contraction processing for the MP2 density, as for the MP2 energy. The contraction of Eqs. (11) and (14) might be a future target to be parallelized for acceleration, since it scales as O^2V^3.

The density matrix $\mathbf{P}$ provides several options. By diagonalizing $\mathbf{P}$, a set of natural orbitals (NO) is obtained [2, 3, 4]. The set could be used as a preparation for CASSCF calculations, as done in Ref. [53]. It is known that the MP2–NO occupation number can override the range between 0 and 2, because of the nonvariational nature of the response density [4]. The degree of the violation in occupation numbers is a measure of near-degeneracy or of linear dependency of a basis set [54]. If one uses standard basis functions, such difficulty would merely take place as far as usual peptides are concerned.

The FMO–HF molecular gradient has already been developed [27], and it has also been used in a molecular dynamics simulation [55]. The extension to the FMO–

MP2 gradient would be straightforward if the approximated P_{ai} as Eq. (11) is used for the gradient expressions [46, 47, 48, 49, 50, 51, 52]. We have started such work for ABINIT-MP [30]. Property evaluation, for example, the polarizability tensors as the response to the electric field, could be similarly performed as described in Ref. [56].

Implementation

In the integral transformation algorithm, we use the libraries of DAXPY and the DDOT of level-1 BLAS [42], as shown in Fig. 1. The reason why we do not use higher-level libraries such as DGEMM, unlike Ref. [21], is to perform the three screening steps. It is well known that a loop-unrolled processing is available in DAXPY and DDOT. The execution efficiency becomes high when the stride of access is one for both associated vector elements. The second quarter transformation for the index q by DDOT is just the case. For the third and fourth steps, a transposed coefficient matrix is thus introduced to make the stride of both accesses of j and b one in the implementation.

The chance to do the in-core mode parallel processing could be substantial, depending on both the available memory area and the fragmentation of the target molecular system. In fact, the default mode of the HF calculations in the ABINIT-MP code [30] is of in-core type. In other words, to reduce the computational cost of total HF calculations by avoiding repeated evaluations of integrals, the distributed integral list is first calculated into some buffer area. It is used to form the partial Fock matrix in each worker process during the iteration of MO optimization. We have thus implemented both the in-core mode and the integral-direct mode for the MP2 calculations in the ABINIT-MP system. When the in-core MP2 processing is requested, the integral list generated in the precedent HF procedure should be reused. In the integral-direct MP2 case, the distribution of $s_{[B]}$ in Fig. 1 is actually driven by shells of AO basis functions, as in Refs. [19, 21]. The so-called load-balancing should be an issue in the parallelism of integral processing. Schütz and Lindh [21] proposed a "self-scheduling" to take the balancing. But, our first implementation for the $s_{[B]}$ distribution is based on a naive round-robin fashion, as in the HF parallelization of ABINIT-MP.

The control of MP2 parallelization both of the in-core and of the integral-direct modes obeys the standard MPI libraries [37]. To complete the $(i_{[\mathrm{B}]}a, jb)$ list via gathering the partially constructed $(i_{[\mathrm{B}]}a, jb_{s_{[\mathrm{B}]}})$ lists which are held in WIAJB$(V, O, V, O_{[\mathrm{B}]})$ in worker processes (refer again to Tables 1, 2), the MPI_ALLREDUCE routine playing as a barrier is used after the exit of $i_{[\mathrm{B}]}$–a–j nested loops. In contrast to Refs. [20, 21, 22], no special "global memory" aggregated over distributed processors is necessary in our implementation in ABINIT-MP. The OpenMP parallelization [57] would, however, be usable, if clusters of shared-memory multiprocessors (SMPs) are available. This could relax the memory requirement of WIAJB in treating large fragments, where the parallelization control of the AO index s would be of two levels. Namely, the WIAJB array is shared by OpenMP in a SMP at the lower level, and the corresponding arrays on SMPs are all-reduced by MPI at the upper level. The usage of OpenMP will be a concern for us.

Test calculations for energy and density

Before discussing the results of test calculations of the parallelized MP2 calculations with the modified ABINIT-MP, we should check the effects of the fragmentation and of the approximation for the P_{ai} density of Eq. (11). To obtain the reference data, we used the GAMESS program, downloaded from the official Ames Laboratory site [58]. The MP2 density obtained by the exact CPHF equation was implemented by Dupuis et al. [48] in GAMESS. The 6-31G basis of VDZ type [44] was used as the standard basis function for all the test molecules, unless otherwise denoted. The frozen-core restriction for the K shell of carbon, nitrogen, and oxygen atoms was applied throughout. The threshold of T_α, T_β, and T_γ was set to 1.0×10^{-8} in ABINIT-MP. Through test calculations without fragmentation, we found that the MP2 energies coincided with the reference energy in six decimals.

The MP2–NO occupation numbers without the fragmentation were compared with the GAMESS [58] results of some water clusters and amino acids. The difference in the NO occupations was at most 3×10^{-3} around the "occupied–virtual boundary", and that of the total increase in the virtual space was at most 2×10^{-2}. These small errors in occupations would be rather encouraging, considering the merit that appropriate MP2 densities could be obtained by a very simple procedure. Note that negative occupation numbers were scarcely found.

The HF total energy and the MP2 correlation energy of the cyclic water trimer are shown in Table 3, where the single water molecule corresponds to a monomer in the FMO calculations. The reference results by GAMESS [58] are also compiled in the table. The errors due to fragmentation are very small in both HF and MP2 energies, showing the validity of Eq. (7). The Mulliken populations are compared with the reference results in Table 4. Equation (8) was used to construct a total density in the FMO scheme. It is found that the

Table 3. HF total energy and MP2 correlation energy for the cyclic water trimer with 6-31G basis (the values of N, O, and V of the cyclic water trimer are 39, 12, and 24, respectively) (in atomic units). The fragment "monomer" corresponded to a single water molecule in ABINIT-MP. The monomer calculation corresponds to that for the single water molecule. Namely, there are three monomer calculations with an $(N^{[X]}/O^{[X]}/V^{[X]})$ combination of (13/4/8). The number of dimer calculations with two water molecules is 3; $(N^{[XY]}/O^{[XY]}/V^{[XY]})$ is (26/8/16). Reference results by GAMESS [58] are included

	GAMESS	ABINIT-MP	Δ
HF	−227.988838	−227.988849	0.000010
MP2	−0.384144	−0.384237	0.000093

Table 4. HF- and MP2-based Mulliken populations for the cyclic water trimer with the 6-31G basis (in the net charge presentation). The fragment monomer corresponded to a single water molecule in ABINIT–MP. Reference results by GAMESS [58] are included. MP2 populations without fragmentation (*WF*) are also shown

		GAMESS		ABINIT−MP		
		HF	MP2	*WF* MP2	1 × 3 HF	MP2
Water 1	O	−0.8613	−0.7941	−0.8006	−0.8611	−0.8003
	H	0.4004	0.3755	0.3771	0.3999	0.3765
	H	0.4576	0.4138	0.4193	0.4576	0.4190
Water 2	O	−0.8617	−0.7929	−0.7997	−0.8613	−0.7991
	H	0.4041	0.3793	0.3808	0.4036	0.3802
	H	0.4594	0.4162	0.4213	0.4595	0.4213
Water 3	O	−0.8595	−0.7917	−0.7983	−0.8588	−0.7975
	H	0.4596	0.4174	0.4224	0.4599	0.4226
	H	0.4013	0.3765	0.3778	0.4008	0.3773

approximation for P_{ai} affects at most in three decimals, comparing between GAMESS's full CPHF values [43, 53] and "without fragmentation" (*WF*) MP2 values. The calculated dipole moment values are 1.4858 D by the full CPHF and 1.4863 D by the *WF*. By comparing the MP2 population with the HF population, one can see that excess ionicity is relaxed by introducing the electron correlation, and that this is consistent with the reduction of the dipole moment from the HF value of 1.5300 D. The population differences between with fragmentation and *WF* by HF and MP2 are sufficiently small.

Another example is the glycine pentamer with an α-helix structure. The HF and the MP2 energies are listed in Table 5. For this molecule, two types of fragmentation were adapted, "1×5" and "$2 \times 2 + 1$". Each glycine residue corresponds to monomer in the former, but two glycine residues form a monomer in the latter; one terminal glycine remains odd. Details of the fragmented calculations can be found in Table 5. The error in the MP2 energy by the $2 \times 2 + 1$ fragmentation is better (less than 1 kcal/mol) than that of the 1×5 fragmentation. The dipole moment is calculated to be 45.416 D by the full CPHF treatment with GAMESS [58] and 45.121 D by the *WF* scheme, where the HF value is 48.839 D. This fact indicates again that the ionic character of the system is relaxed by introducing electron correlation and the approximation for P_{ai} is acceptable. The standard deviations in Mulliken atomic populations of the 1×5 fragmentation are 0.0068 for HF and 0.0551 for MP2, in comparison with the reference results by GAMESS. The corresponding values of the $2 \times 2 + 1$ fragmentation were 0.0037 and 0.0271. In the 1×5 case, the deviation from the reference is unfortunately enhanced at the MP2 level. For the two-body approximation in the ABINIT-MP code [30], we recommend using a monomer including two amino acid residues if possible.

Test calculations for parallel acceleration

A small chunk of the SGI Altix 3700 cluster system, equipped with Intel Itanium-2 processors (1.3 GHz) and 4 GB memory per processor, was used as a main platform to test the acceleration of the MP2 parallelism. The fragment-based parallelization was not invoked for simplicity to check mainly the $s_{[B]}$ parallelization itself in Fig. 1, even though the order of parallelism in the ABINIT-MP/FMO calculation [30] was generally in "fragment first" and "integral second". In other words, the number of processors for each monomer or dimer calculation was assumed not to be so many in the FMO scheme.

Two types of elapsed time were watched. This means timing data of the first to third quarter transformation steps (denoted simply as 1–3 steps in the forthcoming tables) in which there is no communication and of the total calculations including the all-reduce operation and the contraction of the final MP2 energy (plus the MP2 density when denoted). It should be noted here that there is an inherent load-unbalancing source even in the in-core mode runs because the double screening by T_α and T_β is effective during our transformation algorithm. The "slowest timing" among worker processes was thus taken for the elapsed time as the sum of the 1–3 transformation steps. This might not be a good way to demonstrate "impressive speed-up" against the increase of the

Table 5. HF total energy and MP2 correlation energy for the α-helix glycine pentamer with the 6-31G basis (the values of N, O, and V of the glycine pentamer are 223, 59, and 143, respectively) (in atomic units). Two fragmentation schemes, 1×5 (There are five monomer calculations with $(N^{[X]}/O^{[X]}/V^{[X]})$ combinations of (28/6/20), (51/10/36), (51/10/36), (51/10/36), and (78/19/51). The sum of $N^{[X]}$ must not be the same as N because the "bond detached atoms" are introduced in the fragmentation of peptides [27]. The number of dimer calculations is 10. The corresponding $(N^{[XY]}/O^{[XY]}/V^{[XY]})$ combinations are (70/17/47), (79/16/56), (93/21/36), (79/16/56), (102/20/72), (93/21/63), (106/25/71), (129/29/87), (129/29/87), and (120/30/78)) and $2 \times 2 + 1$ (There are three monomer calculations with $(N^{[X]}/O^{[X]}/V^{[X]})$ combinations of (70/17/47), (93/21/63), and (78/19/51). The number of dimer calculations is 3; $(N^{[XY]}/O^{[XY]}/V^{[XY]})$ are (154/39/101), (148/36/98), and (162/41/105)) are used in ABINIT-MP (see text). Reference results by GAMESS [58] are included

	GAMESS	ABINIT-MP			
		1 × 5	Δ	2 × 2 + 1	Δ
HF	−1109.315300	−1109.309545	0.005755	−1109.314019	0.001280
MP2	−2.185625	−2.194660	0.009035	−2.185517	0.000108

number of processors, but we are interested in the reduction of operational costs in applications.

The results of timing and acceleration for the water trimer without fragmentation are listed in Table 6. For both the in-core and the integral-direct modes, the overall acceleration looks encouraging for this very small example ($N = 39, O = 12, V = 24$). The acceleration ratio between two and four processors is better for the only-energy case than that for the energy and density case, mainly owing to the extra cost for the contractions to form the density matrix in serial executions. The parallelization of the density matrix formation will be done in conjunction with the implementation of the MP2 gradient calculations.

A famous drug molecule, aspirin, was employed for the next test on the Altix system, where a polarized 6-31G* basis set [44] was also used. The timing and acceleration for the MP2 energy are compiled in Table 7. Only the integral-direct mode was performed for the case of 6-31G*. With the 6-31G basis set, the acceleration of the integral-direct mode is better than that by the in-core mode, because the granularity is increased by the integral generation in worker processes. The speed-up is 3.5 for the integral-direct mode with four processors. The observed acceleration ratio is close to 2, as the number of processors increases, $2 \rightarrow 4$. The use of 6-31G* is less efficient because of an increased load-unbalance of integral generation. Certainly, the load-balancing is the issue to be improved.

The case of coparallelization of $I_{O_{[B]}}$ and $s_{[B]}$ for the aspirin molecule, where the integral-direct mode was used is listed in Table 8. By parallelizing $I_{O_{[B]}}$ with the processor increase, $4 \rightarrow 8$, the number of the integral generation is reduced from 2 to 1. The observed acceleration ratio is 2, as expected from the fact that O is 34 and is thus divisible ($O_{[B]} = 17$).

Here, the acceleration on a PC cluster system is checked. A cluster with Intel dual Xeon processors (2.2

Table 6. Elapsed time (in seconds) and acceleration for the cyclic water trimer with the 6-31G basis on an Altix 3700. The MP2 energy and density are calculated. Blocking by $O_{[B]}$ is not used

No. of processors		1	2	4
In-core				
Energy	1–3 steps	0.46	0.27	0.14
	Speed-up		1.71	3.33
	Ratio			1.95
	Total	0.53	0.30	0.17
	Speed-up		1.75	3.15
	Ratio			1.79
Energy plus density	1–3 steps	0.53	0.30	0.15
	Speed-up		1.75	3.45
	Ratio			1.79
	Total	0.86	0.53	0.33
	Speed–up		1.62	2.57
	Ratio			1.59
Integral-direct				
Energy	1–3 steps	2.60	1.58	0.79
	Speed-up		1.65	3.31
	Ratio			2.00
	Total	2.73	1.64	0.83
	Speed-up		1.66	3.28
	Ratio			1.97
Energy plus density	1–3 steps	2.77	1.65	0.83
	Speed-up		1.68	3.34
	Ratio			1.99
	Total	3.34	1.98	1.09
	Speed-up		1.68	3.07
	Ratio			1.82

Table 7. Elapsed time (in seconds) and acceleration for the aspirin molecule on an Altix 3700. The MP2 energy is calculated

No. of processors		1	2	4
In-core				
6-31G[a]	1–3 steps	90.8	62.2	31.1
	Speed-up		1.46	2.92
	Ratio			2.00
	Total	112.2	73.7	38.3
	Speed-up		1.52	2.93
	Ratio			1.92
Integral-direct				
6-31G	1–3 steps	339.5	192.4	96.2
	Speed-up		1.76	3.53
	Ratio			2.00
	Total	360.9	204.1	102.9
	Speed-up		1.77	3.51
	Ratio			1.98
Integral-direct				
6-31 G*[b]	1–3 steps	2462.0	1518.3	777.8
	Speed-up		1.62	3.17
	Ratio			1.95
	Total	2571.7	1579.3	815.0
	Speed-up		1.63	3.16
	Ratio			1.94

[a] The HF total energy and MP2 energy are −644.663639 and −1.283435, respectively. The values of N, $O(= O_{[B]})$, and V are 133, 34, and 86, respectively

[b] The HF total energy and MP2 energy are −644.946978 and −1.850902, respectively. The values of N, $O(= O_{[B]})$, and V are 211, 34, and 164, respectively

Table 8. Elapsed time (in seconds) and acceleration ratio for the aspirin molecule on an Altix 3700. The MP2 energy is calculated in the integral-direct mode by splitting the occupied orbital list. Four processors are assigned for the $s_{[B]}$ parallelization

No. of processors		4[a]	8[b]
	$O_{[B]}$	17 + 17	17/17
6-31G	1–3 steps	164.2	82.0
	Ratio		2.00
	Total	171.6	85.8
	Ratio		2.00
6-31G*	1–3 steps	1398.6	699.4
	Ratio		2.00
	Total	1457.5	738.5
	Ratio		1.97

[a] Integral generation is repeated twice for two blocks of $I_{O_{[B]}}$ with four processors

[b] Each block of $I_{O_{[B]}}$ is processed with four processors

GHz clock rate and 2 GB memory per processor) and the usual 100BASE-TX ethernet was used as the second platform. Test calculations were made for the water trimer with 6-31G, 6-31G*, and 6-31G** basis sets [44] in the integral-direct mode, without the fragmentation. The results for the MP2 energy calculation are shown in Table 9. The acceleration is encouraging. One can see that the difference between the speed-up for the 1–3 quarter transformation steps and that of the total calculation is somewhat enlarged, relative to the case of Altix (refer to the results of the integral-direct mode in Table 6). The difference in the performance of the network would be responsible for it. The values of 2.02 and 4.03 are found as the acceleration for the 1–3 steps in the 6-31G calculation, but this ‘‘overlinear speed-up’’ is not surprising because of the efficiency of the cache.

The α-helix glycine pentamer was used for the acceleration test up to eight processors for the distribution of the index s on the Altix, both with 1×5 and with $2 \times 2 + 1$ fragmentations. The numbers of both the monomer and dimer calculations and the cost of the calculations are of course different between the two fragmentation schemes (Table 5). The elapsed time is the sum of the FMO–MP2 calculations in this example, and thus its observed speed-up would become the average. The timings by the in-core mode runs are shown in Table 10. For the energy calculation, the values of the speed-up are 2.5 for the 1×5 fragmentation and 2.8 for the $2 \times 2 + 1$ fragmentation with four processors. The corresponding values with eight processors are 4.8 and 5.0. The acceleration ratio for the increase of the number of processors, 4 → 8, is better than that for 2 → 4. These results of the speed-up are reasonable, implying that the integral-driven parallelism could be effective when the number of fragments in the target peptide is rather low, just as for the glycine pentamer. With four processors,

Table 9. Elapsed time (in seconds) and acceleration for the cyclic water trimer on a Xeon-based PC cluster (see text). The MP2 energy is calculated in the integral-direct mode

No. of processors		1	2	4
6-31G	1–3 steps	5.00	2.48	1.24
	Speed-up		2.02	4.03
	Ratio			2.00
	Total	5.05	2.63	1.50
	Speed-up		1.92	3.38
	Ratio			1.76
6-31G*[a]	1–3 steps	24.76	13.45	7.26
	Speed-up		1.84	3.41
	Ratio			1.85
	Total	24.96	13.92	8.05
	Speed-up		1.79	3.10
	Ratio			1.73
6-31G**[b]	1–3 steps	52.76	29.35	18.71
	Speed-up		1.80	2.82
	Ratio			1.57
	Total	53.29	30.37	20.36
	Speed-up		1.75	2.62
	Ratio			1.49

[a] The HF total energy and the MP2 energy are −228.059840 and −0.562968, respectively. The values of N, $O(= O_{[B]})$, and V are 57, 12, and 42, respectively

[b] The HF total energy and the MP2 energy are -228.098127 and −0.594003, respectively. The values of N, $O(= O_{[B]})$, and V are 75, 12, and 60, respectively

Table 10. Elapsed time (in seconds) and acceleration for the α-helix glycine pentamer with 6-31G basis on an Altix 3700. FMO–MP2 calculations with 1×5 and $2 \times 2 + 1$ fragmentations are performed in the in-core mode. (See Table 5 for details of the fragmented calculations. Blocking by $O_{[B]}$ is not used)

No. of processors		1	2	4	8
1×5					
Energy	1–3 steps	288.8	196.3	110.8	55.0
	Speed-up		1.47	2.61	5.25
	Ratio			1.77	2.01
	Total	353.0	231.9	139.6	73.3
	Speed-up		1.52	2.53	4.82
	Ratio			1.66	1.91
Energy plus density	1–3 steps	443.9	255.8	149.1	80.6
	Speed-up		1.74	2.98	5.51
	Ratio			1.72	1.85
	Total	835.7	464.8	332.5	241.9
	Speed-up		1.80	2.51	3.46
	Ratio			1.40	1.37
$2 \times 2 + 1$					
Energy	1–3 steps	520.6	336.7	194.9	98.2
	Speed-up		1.55	2.67	5.30
	Ratio			1.73	1.98
	Total	657.0	440.5	238.4	131.7
	Speed-up		1.84	2.76	4.99
	Ratio			1.85	1.81
Energy plus density	1–3 steps	889.7	617.1	318.2	162.2
	Speed-up		1.44	2.80	5.48
	Ratio			1.94	1.96
	Total	1568.8	1106.0	721.5	539.9
	Speed-up		1.42	2.17	2.91
	Ratio			1.53	1.34

the turnaround times for the "HF job" and the "HF plus MP2 energy job" were 263 and 539 s, respectively, for the $2 \times 2 + 1$ fragmentation. Unfortunately, the overall acceleration is decreased for the energy plus density calculation. The total speed-up with eight processors is rather discouraging. The reason should be attributed to the increased portion of serial execution for the contraction processing to form the density matrices of Eqs. (11) and (14) (which scale as O^2V^3), as stated previously for the water trimer.

Another PC cluster system with the latest Xeon processors (3.06 GHz) and the giga-ethernet was available to us. Using 60 processors, we carried out a benchmark FMO–MP2 calculation for a HB cluster consisting of 1024 water molecules with the 6-31G basis set. The single water molecule was treated as a monomer, and the parallelization was done both for the monomer/dimer and for the AO index of s. The in-core mode was used for integral transformations. The turnaround time for the "HF plus MP2 energy and density job" was only 394 s, showing that PC clusters would be useful as a platform for the parallelized FMO–MP2 calculations. The results including chemical interest will be published elsewhere [59].

Summary

A new parallelized integral-direct and in-core MP2 scheme, which calculates both the correlation energy and the approximated response density, has been developed and implemented in the ABINIT-MP FMO calculation system [30]. The fundamental transformation algorithm was based on a modification of recipe by Mochizuki et al. [25]. The triple screening procedure was incorporated to effectively reduce the operational costs. No communication across the worker processes is needed during the integral transformation. Encouraging acceleration by the parallelization was observed through the test calculations. Work to implement the MP2 gradient and to improve the efficiency of parallelism is in progress. The fragmentation error in the MP2 energies was confirmed to be small with chemical accuracy. The introduction of MP2 correlation compensates the excess ionic contribution by the zeroth-order HF description, as demonstrated by the water trimer and the glycine pentamer.

Acknowledgements. The present work was partially supported by the Frontier Simulation Software for Industrial Science of the Information Techonology project being operated by the Ministry of Education, Culture, Sports, Science and Technology (MEXT) at the Institute for Industrial Science (IIS) of The University of Tokyo. All test calculations were carried out on the computing facilities of the SGI Altix at IIS and of the PC cluster at the National Institute of Health Sciences. Y.M. thanks Shigenori Tanaka and Kaori Fukuzawa for the discussion on the importance of HB and vdW interactions in biochemical systems. The molecular geometry data of the test calculations are available upon request by e-mail.

References

1. Møller C, Plesset MS (1934) Phys Rev 46:618
2. Szabo A, Ostlund NS (1982) Modern quantum chemistry. Macmillan, New York
3. Helgaker T, Jørgensen P, Olsen J (2000) Molecular electronic-structure theory. Wiley, Chichester
4. Creamer D (1988) In: Schleyer PvR, Aliinger NL, Clark T, Gasteiger J, Kollman PA, Schaefer HF III (eds) Encyclopedia of computational chemistry. Wiley, Chichester, pp 1706–1735
5. Hu C-H D, Chong DP (1988) In: Schleyer PvR, Aliinger NL, Clark T, Gasteiger J, Kollman PA, Schaefer HF III (eds) Encyclopedia of computational chemistry. Wiley, Chichester, pp 664–677
6. Elstner M, Hobza P, Frauenheim T, Suhai S, Kaxiras E (2001) J Chem Phys 114:5149
7. Cybulski SM, Bledson TM, Toczylowski RR (2002) J Chem Phys 116:11039
8. Kurita N, Inoue H, Sekino H (2003) Chem Phys Lett 370:161
9. Tang KC, Edmiston C (1972) J Chem Phys 52:547
10. Pulay P, Saebø S (1986) Theor Chim Acta 69:357
11. Bernholdt DE, Harrison RJ (1995) J Chem Phys 102:9582
12. Maslen PE, Head-Gordon M (1998) J Chem Phys 109:7093
13. Rauhut G, Pulay P, Werner H-J (1998) J Comput Chem 19:1241
14. Pipek J, Bogár F (1999) In: Surján PR (ed) Correlation and localization, Springer, Berlin Heidelberg New York, pp 43–61
15. Nakao Y, Hirao K (2004) J Chem Phys 120:6375
16. Watts JD, Dupuis M (1988) J Comput Chem 9:158
17. Limaye AC, Gadre SR (1994) J Chem Phys 100:1303
18. Márquez AM, Dupuis M (1995) J Comput Chem 16:395
19. Nielsen IMB, Seidl ET (1995) J Comput Chem 16:1301
20. Wong AT, Harrison RJ, Rendell AP (1996) Theor Chim Acta 93:317
21. Schütz M, Lindh R (1997) Theor Chim Acta 95:13
22. Fletcher GD, Schmidt MW, Bode BM, Gordon MS (2000) Comput Phys Comm 128:190
23. Baker J, Pulay P (2002) J Comput Chem 23:1150
24. Wolinski K, Pulay P (2003) J Chem Phys 118:9497
25. Mochizuki Y, Nishi N, Hirahara Y, Takada T (1996) Theor Chim Acta 93:211
26. Kitaura K, Sawai T, Asada T, Nakano T, Uebayasi M (1999) Chem Phys Lett 312:319
27. Kitaura K, Ikeo E, Asada T, Nakano T, Uebayasi M (1999) Chem Phys Lett 313:701
28. Nakano T, Kaminuma T, Sato T, Akiyama Y, Uebayasi M, Kitaura K (2000) Chem Phys Lett 318:614
29. Kitaura K, Sugiki S, Nakano T, Komeiji Y, Uebayasi M (2001) Chem Phys Lett 336:163
30. (a) Nakano T, Kaminuma T, Sato T, Fukuzawa K, Akiyama Y, Uebayasi M, Kitaura K (2002) Chem Phys Lett 351:475; (b) ABINIT-MP http://moldb.nihs.go.jp/abinitmp/
31. Inadomi Y, Nakano T, Kitaura K, Nagashima U (2002) Chem Phys Lett 364:139
32. Fedorov DG, Olson RM, Kitaura K, Gordon MS, Koseki S, (2004) J Comput Chem 25:872
33. Fedorov DG, Kitaura K (2004) J Chem Phys 120:6832
34. Fedorov DG, Kitaura K (2004) Chem Phys Lett 389:129
35. Sekino H, Sengoku Y, Sugiki S, Kurita N (2003) Chem Phys Lett 378:589
36. Sugiki S, Kurita N, Sengoku Y, Sekino H (2003) Chem Phys Lett 382:611
37. Message passing interface http://www-unix.mcs.anl.gov/mpi/
38. Almlöf J, Faegri K, Korsell K (1982) J Comput Chem 3:385
39. Schütz M, Lindh R, Werner H-J (1999) Mol Phys 96:719
40. Nakata K, Murase T, Sakuma T, Takada T (2002) J Comput Appl Math 149:351
41. Taylor PR (1987) Int J Quantum Chem 31:521
42. Basic linear algebra subroutines, http://www.netlib.org/blas/
43. Yamamoto S, Nagashima U, Aoyama T, Kashiwagi H (1988) J Comput Chem 9:627
44. Foresman JB, Frisch A (1996) Exploring chemistry with electrononic structure methods, 2nd edn. Gaussian, Pittsburgh
45. Foresman JB, Head-Gordon M, Pople JA, Frisch MJ (1992) J Phys Chem 96:135

46. Frisch MJ, Head-Gordon M, Pople JA (1990) Chem Phys Lett 166:281
47. Handy NC, Schaefer HF (1984) J Chem Phys 81:5031
48. Dupuis M, Chin S, Márquez A (1994) In: Malli GL (ed) Relativistic and electron correlation effects in molecules and solids. Plenum, New York, pp 315–338
49. Yamaguchi Y, Osamura Y, Goddard JD, Schaefer HF III (1994) A new dimension to quantum chemistry: analytic derivative methods in ab initio molecular electronic structure theory. Oxford University Press, New York
50. Fletcher GD, Rendell AP, Sherwood P (1997) Mol Phys 91:431
51. Nielsen IMB (1996) Chem Phys Lett 255:210
52. Aikens CM, Webb SP, Bell RL, Fletcher GD, Schmidt MW, Gordon MS (2003) Theor Chem Acc 110:233
53. (a) Jensen HJA, Jørgensen P, Ågren H, Olsen J (1988) J Chem Phys 88:3834; (b) Erratum (1988) J Chem Phys 89:5354
54. Gordon MS, Schmidt MW, Glaesemann K, Stevens WJ, Gonzalez C (1999) J Chem Phys 110:4199
55. Komeiji Y, Nakano T, Fukuzawa K, Ueno Y, Inadomi Y, Nemoto T, Uebayashi M, Fedorov DG, Kitaura K (2003) Chem Phys Lett 372:342
56. Minikis R, Kairys V, Jensen JH (2001) J Phys Chem A 105:3829
57. OpenMP http://www.openmp.org/
58. (a) Schmidt MW, Baldridge KK, Boatz JA, Elbert ST, Gordon MS, Jensen JH, Koseki S, Matsunaga N, Nguyen KA, Su SJ, Windus TL, Dupuis M, Montgomery JA (1993) J Comput Chem 14:1347; (b) GAMESS http://www.msg.ameslab.gov/GAMESS/
59. Mochizuki Y, Koikegami S, Nakano T, Amari S, Kitaura K (2004) Chem Phys Lett, in press