

Scalable implementation of analytic gradients for second-order Z-averaged perturbation theory using the distributed data interface

Christine M. Aikens, Graham D. Fletcher, Michael W. Schmidt, and Mark S. Gordon

Citation: *The Journal of Chemical Physics* **124**, 014107 (2006); doi: 10.1063/1.2140688

View online: <http://dx.doi.org/10.1063/1.2140688>

View Table of Contents: <http://scitation.aip.org/content/aip/journal/jcp/124/1?ver=pdfcov>

Published by the [AIP Publishing](#)

Articles you may be interested in

[Derivation of general analytic gradient expressions for density-fitted post-Hartree-Fock methods: An efficient implementation for the density-fitted second-order Møller–Plesset perturbation theory](#)

J. Chem. Phys. **141**, 124108 (2014); 10.1063/1.4896235

[Analytic energy gradients for the orbital-optimized second-order Møller–Plesset perturbation theory](#)

J. Chem. Phys. **138**, 184103 (2013); 10.1063/1.4803662

[Analytic energy gradient for second-order Møller–Plesset perturbation theory based on the fragment molecular orbital method](#)

J. Chem. Phys. **135**, 044110 (2011); 10.1063/1.3611020

[Analytical energy gradients in second-order Møller–Plesset perturbation theory for extended systems](#)

J. Chem. Phys. **109**, 4147 (1998); 10.1063/1.477020

[Analytical energy gradients for local second-order Møller–Plesset perturbation theory](#)

J. Chem. Phys. **108**, 5185 (1998); 10.1063/1.475955

 **AIP** | APL Photonics

APL Photonics is pleased to announce
Benjamin Eggleton as its Editor-in-Chief



Scalable implementation of analytic gradients for second-order Z-averaged perturbation theory using the distributed data interface

Christine M. Aikens,^{a)} Graham D. Fletcher,^{b)} Michael W. Schmidt, and Mark S. Gordon^{c)}
Department of Chemistry, Iowa State University, Ames, Iowa 50011

(Received 12 September 2005; accepted 1 November 2005; published online 4 January 2006)

The analytic gradient expression for second-order Z-averaged perturbation theory is revised and its parallel implementation is described in detail. The distributed data interface is used to access molecular-orbital integral arrays stored in distributed memory. The algorithm is designed to maximize the use of local data and reduce communication costs. The iterative solution and the preconditioner used to induce the convergence of the coupled-perturbed Hartree-Fock equations are presented. Several illustrative timing examples are discussed. © 2006 American Institute of Physics.
 [DOI: 10.1063/1.2140688]

I. INTRODUCTION

Second-order Møller-Plesset (MP2) perturbation theory¹ is extensively employed to calculate molecular energies and geometries since it provides a balance between the amount of electron correlation recovered and the computational requirements of a calculation. For closed-shell systems, perturbation theory based on a restricted Hartree-Fock (RHF) wave function is well defined. However, open-shell systems are much more challenging. Although perturbation theory based on an unrestricted Hartree-Fock (UHF) wave function follows directly from the MP2 spin-orbital equations, the wave function is not an eigenfunction of the \hat{S}^2 operator, and this results in spin-contamination problems. In addition, since different spatial orbitals are used for the α and β electrons, the method requires three to four times as much memory, disk space, etc., as a comparable closed-shell calculation. Spin-projected methods may alleviate some of the spin contamination, but this can introduce size-consistency problems.²

As a result, many open-shell perturbation-theory methods based on RHF wave functions have been proposed.³⁻⁹ These methods may be divided into two classes: those that have a configuration state function (CSF) basis (which are \hat{S}^2 eigenfunctions) and those that have a spin-orbital determinant basis. Methods with a CSF basis include the method of Hubac and Carsky (HCPT),³ the OPT1 and OPT2 methods,⁴ and the IOPT method.⁹ Methods using a spin-orbital basis include the ROMP,⁵ RMP (Ref. 6)/ROHF-MBPT,⁷ and Z-averaged perturbation-theory⁸ methods. The convergence of the perturbation series for HCPT and OPT1 is expected to be poor.^{5,10} OPT2 is not invariant with respect to rotations between degenerate singly occupied orbitals.¹¹ IOPT is not size consistent,¹² although the effects of this on its results

have been debated.¹³ ROMP and RMP use different α and β spatial orbitals, so the cost of these two methods will be similar to unrestricted perturbation theory.

Z-averaged perturbation theory,⁸ (ZAPT) is based on a set of “symmetric spin-orbitals,”¹⁴ and the symmetry of these orbitals leads to significant computational savings over ROMP and RMP. The spatial parts of the spin orbitals come from an open-shell ROHF calculation. The standard α and β spin functions are assigned to doubly occupied and unoccupied orbitals in the reference function. The spin parts of the singly occupied orbitals are assigned to be σ^+ and σ^- , where

$$\sigma^+ = \frac{1}{\sqrt{2}}(\alpha + \beta),$$

$$\sigma^- = \frac{1}{\sqrt{2}}(\alpha - \beta).$$

So, the reference function is symmetric with respect to the interchange of α and β spin indices. The single determinant based on these orbitals is an eigenfunction of \hat{S}^2 , although the n th-order perturbed wave functions generally are not. There are no \hat{S}^2 contaminants in the second-order (ZAPT2) energies. The reference function is a linear combination of \hat{S}_z eigenfunctions, but the raising and lowering operators \hat{S}_+ and \hat{S}_- can be used to recover eigenfunctions of \hat{S}_z .¹⁴ The canonicalization chosen for the molecular orbitals (MO's) is equivalent to using $\frac{1}{2}(\hat{F}^\alpha + \hat{F}^\beta)$ to canonicalize the three shells [doubly occupied, singly occupied, and unoccupied (virtual)].¹⁵ The zeroth-order Hamiltonian is defined as the diagonal part of the spin-orbital Fock matrix.

In order to effectively use this method, it is desirable to calculate analytic derivatives of the second-order energy. Equations for the ZAPT2 gradient have been previously derived by Fletcher *et al.*¹⁶ Revisions to these equations are made in Sec. II of this work. The subsequent implementation is described in Sec. III.

The efficient implementation of gradients for post-Hartree-Fock methods has been of interest for many years. The first closed-shell MP2 gradient derivation was presented by Pople *et al.* in 1979.¹⁷ Later, the storage of derivative integrals and three-virtual MO integrals was eliminated,¹⁸ the

^{a)}Current address: Department of Chemistry, Northwestern University, Evanston, IL 60208-3113.

^{b)}Current address: Room A20, CCLRC Daresbury Laboratory, WA4 4AD, UK.

^{c)}Electronic mail: mark@si.fi.ameslab.gov

number of unknown response vectors was reduced to 1,¹⁹ and the need to solve the coupled-perturbed Hartree-Fock equations in the occupied-occupied and virtual-virtual blocks was removed.²⁰ Frozen orbitals further reduced the computational cost.²¹ Direct and semidirect algorithms reduced the required memory and disk storage requirements and permitted larger calculations, although this came at the cost of increased integral computation.²²

Even with these improvements, the computational requirements for large molecules with reasonable basis sets quickly outgrow the capabilities of a single-processor computer. For second-order perturbation theory, the time requirements for a calculation increase as $O(n^5)$ with the number of basis functions n , and the memory needs can grow as $O(n^3)$, depending on the algorithm. Distributed data parallel schemes are desirable, as these alleviate both the single-node memory and time requirements of a calculation.

The distributed data interface (DDI) was developed by Fletcher *et al.* to provide a set of tools for distributed memory programming.^{23,24} Like global arrays (GA),²⁵ DDI provides a programming environment that mimics the ease of a shared-memory environment yet permits the scalability of a distributed memory architecture. Large data arrays can be stored over the collective memory of the system. These arrays can be easily accessed by any of the nodes, although the programmer must be aware of the increased time required for remote memory access and should design algorithms that maximize the use of local memory. In order to maximize scalability, the programmer must also limit the replicated memory requirements. To date, distributed storage algorithms for self-consistent field (SCF) energies,²⁶ analytic SCF Hessians, closed-shell²³ and unrestricted²⁷ MP2 energies and gradients, ZAPT2 energies, multireference self-consistent field energies,²⁸ gradients, and Hessians,²⁹ multi-reference perturbation-theory energies,³⁰ single excitation configuration-interaction (CI) energies²⁸ and gradients,³¹ and full CI energies³² have been implemented using DDI into the electronic structure code GAMESS.³³ Recently, DDI has been extended to take optimal advantage of shared-memory processor (SMP) computers.³⁴

II. THEORY

A. Notation and energy expression

The notation used here is as similar as possible to a recent overview of MP2 gradient equations.³⁵ The ZAPT2 energy is the sum of the SCF and second-order correction energies. The frozen-core ZAPT second-order energy correction may be expressed in the MO basis as

$$E_{\text{ZAPT}}^{(2)} = \frac{1}{2} \sum_{ij} \sum_{pq}^{s.v.} (ip|jq)[C_{pq}(ip|jq) - (iq|jp)]/D_{ij}^{pq} \\ + \frac{1}{2} \sum_{ab} \sum_{pq}^{d.s.} (pa|qb)[C_{pq}(pa|qb) - (pb|qa)]/D_{pq}^{ab} \\ + \sum_{ixya} (ix|ya)(iy|xa)/D_{iy}^{xa}$$

$$+ \frac{1}{2} \sum_{ixya} (ix|xa)(iy|ya)/D_i^a. \quad (1)$$

General MO's are indexed by p, q, r , and s . Doubly occupied active MO's (referred to as DOCC) are indexed by i, j , and k . Singly occupied MO's (SOCC) are indexed by w, x, y , and z . Virtual (unoccupied) MO's (VIRT) are indexed by a, b , and c . Doubly occupied frozen-core MO's (CORE) are indexed by m and n . The summation range denoted *s.v.* extends over the SOCC and VIRT spaces; likewise, *d.s.* extends over the DOCC and SOCC spaces and *c.d.* extends over the CORE and DOCC spaces. The four-index symbol $(pq|rs)$ is a two-electron repulsion integral (ERI) over MO's in Mulliken notation. These integrals are transformed into the MO basis from the atomic-orbital (AO) basis by

$$(pq|rs) = \sum_{\mu\nu\lambda\sigma} C_{\mu p} C_{\nu q} C_{\lambda r} C_{\sigma s} (\mu\nu|\lambda\sigma), \quad (2)$$

where μ, ν, λ , and σ index AO's, $C_{\mu p}$ is a MO coefficient, and $(\mu\nu|\lambda\sigma)$ is an ERI in the AO basis. The MO coefficient $C_{\mu p}$ must be distinguished from the coefficient in the ZAPT2 energy expression in Eq. (1), where the coefficient C_{pq} is

$$C_{pq} = 1 \quad \text{for both } p \in \text{SOCC and } q \in \text{SOCC} \\ C_{pq} = 2 \quad \text{otherwise.} \quad (3)$$

The four-index denominator is expressed as

$$D_{pq}^{rs} = \varepsilon_p + \varepsilon_q - \varepsilon_r' - \varepsilon_s' \quad (4)$$

and the two-index denominator is written as

$$D_p^q = \varepsilon_p - \varepsilon_q'. \quad (5)$$

The orbital energies in Eqs. (4) and (5) are defined as

$$\varepsilon_p = \varepsilon_p' = \varepsilon_{pp} \quad \text{for } p \notin \text{SOCC}, \\ \varepsilon_p = \varepsilon_{pp} - \frac{1}{2} \sum_y (py|py) \quad \text{for } p \in \text{SOCC}, \quad (6)$$

$$\varepsilon_p' = \varepsilon_{pp} + \frac{1}{2} \sum_y (py|py) \quad \text{for } p \in \text{SOCC},$$

with

$$\varepsilon_{pp} = h_{pp} + \sum_q^{\text{all}} f_q [2(pp|qq) - (pq|pq)], \quad (7)$$

where h_{pq} is a core Hamiltonian element and f_p is the occupation number of the p th shell (1 for DOCC; $\frac{1}{2}$ for SOCC). The SOCC orbital energies in Eq. (6) have an additional "integral component" that comes from the choice of canonicalization of the Fock matrix. The abbreviations i.p. and d.p. will be used to denote independent and dependent pairs of molecular orbitals, respectively. A rotation between two doubly occupied, between two singly occupied, or between two virtual orbitals does not change the SCF energy, so these pairs are called dependent pairs. The other orbital pairs are uniquely determined, so these are called independent pairs.

B. General gradient expression

The derivative of the ZAPT2 energy is the sum of the derivatives of the SCF and second-order correction contributions to the energy. The general form of this derivative with respect to a nuclear displacement γ has the following form:

$$E^\gamma = \sum_{\mu\nu}^{AO} P_{\mu\nu} h_{\mu\nu}^\gamma + \sum_{\mu\nu}^{AO} W_{\mu\nu} S_{\mu\nu}^\gamma + \sum_{\mu\nu\lambda\sigma}^{AO} \Gamma_{\mu\nu\lambda\sigma} (\mu\nu|\lambda\sigma)^\gamma. \quad (8)$$

In Eq. (8), $h_{\mu\nu}^\gamma$ are the core Hamiltonian derivative integrals, $S_{\mu\nu}^\gamma$ are the overlap derivative integrals, and $(\mu\nu|\lambda\sigma)^\gamma$ are the derivatives of the ERI's in the AO basis. The one-particle density matrix $P_{\mu\nu}$, energy-weighted density matrix $W_{\mu\nu}$, and two-particle density matrix $\Gamma_{\mu\nu\lambda\sigma}$ are sums of their SCF analogs and a second-order correction,

$$P_{\mu\nu} = P_{\mu\nu}^{\text{SCF}} + P_{\mu\nu}^{(2)}, \quad (9)$$

$$W_{\mu\nu} = W_{\mu\nu}^{\text{SCF}} + W_{\mu\nu}^{(2)}, \quad (10)$$

$$\Gamma_{\mu\nu\lambda\sigma} = \Gamma_{\mu\nu\lambda\sigma}^{\text{SCF}} + \Gamma_{\mu\nu\lambda\sigma}^{(2)}. \quad (11)$$

The second-order two-particle density correction is expressed as the sum of separable (S) and nonseparable (NS) terms,

$$\Gamma_{\mu\nu\lambda\sigma}^{(2)} = \Gamma_{\mu\nu\lambda\sigma}^{\text{S}} + \Gamma_{\mu\nu\lambda\sigma}^{\text{NS}}. \quad (12)$$

The equations for the SCF terms may be written as follows:

$$P_{\mu\nu}^{\text{SCF}} = 2 \sum_p^{c.d.s.} f_p C_{\mu p} C_{\nu p}, \quad (13)$$

$$W_{\mu\nu}^{\text{SCF}} = 2 \sum_{pq}^{\text{all}} C_{\mu p} C_{\nu q} \zeta_{pq}, \quad (14)$$

$$\Gamma_{\mu\nu\lambda\sigma}^{\text{SCF}} = \frac{1}{2} P_{\mu\nu}^{\text{SCF}} P_{\lambda\sigma}^{\text{SCF}} - \frac{1}{4} P_{\mu\lambda}^{\text{SCF}} P_{\nu\sigma}^{\text{SCF}} - \frac{1}{4} P_{\mu\lambda}^{\text{S}} P_{\nu\sigma}^{\text{S}}, \quad (15)$$

where

$$P_{\mu\nu}^{\text{S}} = \sum_x C_{\mu x} C_{\nu x} \quad (16)$$

and ζ_{pq} is defined in Eq. (41) below.

C. Gradient of the ZAPT2 energy

The usual derivative techniques are employed; these are described in detail in Ref. 16. A few necessary revisions to the previous derivation are presented in this section.

The derivative of the fourth term in the second-order energy expression [Eq. (1)] may be written

$$\begin{aligned} & \left(\frac{1}{2} \sum_{ixya} (ix|xa)(iy|ya)/D_i^a \right)^\gamma \\ &= \frac{1}{2} \sum_{ixya} \{ 2(ix|xa)^\gamma (iy|ya)/D_i^a - [D_i^a]^\gamma \\ & \quad \times (ix|xa)(iy|ya)/(D_i^a)^2 \}, \end{aligned} \quad (17)$$

where (γ) indicates the derivative over MO's. The derivative

of an ERI in the MO basis is well known³⁶ to be

$$\begin{aligned} (pq|rs)^\gamma &= (pq|rs)^\gamma + \sum_t U_{tp}^\gamma (tq|rs) + \sum_t U_{tq}^\gamma (pt|rs) \\ & \quad + \sum_t U_{tr}^\gamma (pq|ts) + \sum_t U_{ts}^\gamma (pq|rt), \end{aligned} \quad (18)$$

where the U^γ matrix describes the response of the MO coefficients to the perturbation γ . Upon substitution of Eq. (18) into Eq. (17), the first term in the resulting equation has the form

$$\sum_{ixya} (ix|xa)^\gamma (iy|ya)/D_i^a, \quad (19)$$

where

$$(ix|xa)^\gamma = \sum_{\mu\nu\lambda\sigma} C_{\mu i} C_{\nu x} C_{\lambda x} C_{\sigma a} (\mu\nu|\lambda\sigma)^\gamma. \quad (20)$$

Equation (19) can be rearranged as follows:

$$\begin{aligned} & \sum_{ixya} (ix|xa)^\gamma (iy|ya)/D_i^a \\ &= \sum_{\mu\nu\lambda\sigma} \sum_{ixya} C_{\mu i} C_{\nu x} C_{\lambda x} C_{\sigma a} (\mu\nu|\lambda\sigma)^\gamma (iy|ya)/D_i^a \\ &= \sum_{\mu\nu\lambda\sigma} (\mu\nu|\lambda\sigma)^\gamma \sum_x \frac{1}{2} C_{\nu x} C_{\lambda x} \sum_{ia} 2 C_{\mu i} C_{\sigma a} \sum_y (iy|ya)/D_i^a \\ &= \frac{1}{2} \sum_{\mu\nu\lambda\sigma} (\mu\nu|\lambda\sigma)^\gamma P_{\nu\lambda}^{\text{S}} \sum_{ia} 2 C_{\mu i} C_{\sigma a} T_{ia}^{(4)} \\ &= \frac{1}{2} \sum_{\mu\nu\lambda\sigma} (\mu\nu|\lambda\sigma)^\gamma P_{\nu\lambda}^{\text{S}} T_{\mu\sigma}^{(4)}, \end{aligned} \quad (21)$$

where

$$T_{ia}^{(4)} = \sum_y (iy|ya)/D_i^a, \quad (22)$$

$$T_{\mu\sigma}^{(4)} = 2 \sum_{ia} C_{\mu i} C_{\sigma a} T_{ia}^{(4)}. \quad (23)$$

This yields a separable density term $T_{\mu\sigma}^{(4)} P_{\nu\lambda}^{\text{S}}$. During the derivation, terms such as

$$2 \sum_{y>x} U_{yx}^\gamma {}^1 P_{yx}^{(2)} (\varepsilon'_x - \varepsilon'_y) \quad (24)$$

are formed, where ${}^1 P_{yx}^{(2)}$ indicates the first term of $P_{yx}^{(2)}$. Using Eq. (6), Eq. (24) can be rewritten as

$$2 \sum_{y>x} U_{yx}^\gamma {}^1 P_{yx}^{(2)} \left\{ \varepsilon_{xx} - \varepsilon_{yy} + \frac{1}{2} \sum_z [(xz|xz) - (yz|yz)] \right\}. \quad (25)$$

Equation (25) may be divided into an orbital energy part

$$2 \sum_{y>x} U_{yx}^\gamma {}^1 P_{yx}^{(2)} \{ \varepsilon_{xx} - \varepsilon_{yy} \} \quad (26)$$

and an integral component part

$$2 \sum_{y>x} U_{yx}^\gamma {}^1P_{yx}^{(2)} F_{yx}, \quad (27)$$

where

$$F_{yx} = \frac{1}{2} \sum_z [(xz|xz) - (yz|yz)]. \quad (28)$$

The orbital energy part in Eq. (26) is treated with the usual procedure.¹⁶ Equation (27) is divided into two pieces, and the x and y indices are swapped in the first term as follows:

$$\sum_{x>y} U_{xy}^\gamma {}^1P_{xy}^{(2)} F_{xy} + \sum_{y>x} U_{yx}^\gamma {}^1P_{yx}^{(2)} F_{yx}. \quad (29)$$

The derivative of the orthogonality condition,

$$U_{pq}^\gamma + U_{qp}^\gamma + S_{pq}^\gamma = 0, \quad (30)$$

may be used to rearrange the second half of Eq. (29) to yield

$$\sum_{x>y} U_{xy}^\gamma {}^1P_{xy}^{(2)} F_{xy} - \sum_{y>x} U_{yx}^\gamma {}^1P_{yx}^{(2)} F_{yx} - \sum_{y>x} S_{xy}^\gamma {}^1P_{yx}^{(2)} F_{yx}. \quad (31)$$

Since ${}^1P_{yx}^{(2)} = {}^1P_{xy}^{(2)}$ and $F_{xy} = -F_{yx}$, the first two terms can be combined and the third term can be modified slightly, giving

$$\sum_{x \neq y} U_{xy}^\gamma {}^1P_{xy}^{(2)} F_{xy} + \sum_{y>x} S_{xy}^\gamma {}^1P_{xy}^{(2)} F_{xy}. \quad (32)$$

The second term in Eq. (32) produces a contribution to the one-particle energy-weighted density matrix. Given that

$$U_{pq}^\gamma = \frac{Q_{pq}^\gamma}{(\epsilon_{qq} - \epsilon_{pp})}, \quad (33)$$

the first term of Eq. (32) becomes

$$\sum_{xy} Q_{xy} \left[\frac{1 - \delta_{xy}}{(\epsilon_{yy} - \epsilon_{xx})} {}^1P_{xy}^{(2)} F_{xy} \right], \quad (34)$$

which yields a contribution to the one-particle density matrix. Due to Eq. (34), a potential singularity exists when singly occupied orbitals are degenerate and $\epsilon_{yy} = \epsilon_{xx}$. Assuming F_{xy} goes to zero faster than the orbital energy difference, this term can be neglected for degenerate SOCC orbitals.

Also during the derivation, terms such as

$$2 \sum_{ij} \sum_{pq} \sum_m^{s.v.} U_{mi}^\gamma (mp|jq) [C_{pq}(ip|jq) - (iq|jp)] / D_{ij}^{pq} \quad (35)$$

are produced. This term may be divided into two parts, and Eq. (30) may be used to rearrange Eq. (35) as

$$\begin{aligned} & \sum_{ij} \sum_{pq} \sum_m^{s.v.} U_{mi}^\gamma (mp|jq) [C_{pq}(ip|jq) - (iq|jp)] / D_{ij}^{pq} \\ & - \sum_{ij} \sum_{pq} \sum_m^{s.v.} U_{im}^\gamma (mp|jq) [C_{pq}(ip|jq) - (iq|jp)] / D_{ij}^{pq} \\ & - \sum_{ij} \sum_{pq} \sum_m^{s.v.} S_{mi}^\gamma (mp|jq) [C_{pq}(ip|jq) - (iq|jp)] / D_{ij}^{pq}. \quad (36) \end{aligned}$$

The third term in Eq. (36) includes an overlap derivative

integral, so the expression multiplying the integral is a contribution to the energy-weighted density matrix,

$$W_{im}^{(2)} = - \sum_j \sum_{pq}^{s.v.} (mp|jq) [C_{pq}(ip|jq) - (iq|jp)] / D_{ij}^{pq}. \quad (37)$$

A similar procedure is used to determine the other three amplitude terms of $W_{im}^{(2)}$.

For implementation purposes, the energy-weighted density matrix is symmetrized. Thus,

$$W_{im}^{(2)} = - P_{im}^{(2)} \epsilon_{mm} \quad (38)$$

and

$$W_{mi}^{(2)} = - P_{mi}^{(2)} \epsilon_{ii} \quad (39)$$

become

$$W_{im}^{(2)} = W_{mi}^{(2)} = \frac{1}{2} (W_{im}^{(2)} + W_{mi}^{(2)}) = - \frac{1}{2} P_{mi}^{(2)} (\epsilon_{mm} + \epsilon_{ii}), \quad (40)$$

since the one-particle density is also symmetrized.

In the general restricted open-shell coupled-perturbed Hartree-Fock (CPHF) equations, the Lagrangian matrices ζ_{pq} , the generalized Lagrangian matrices ξ_{pq}^r , and the τ matrices can be defined by

$$\zeta_{pq} = f_p h_{pq} + \sum_s^{c.d.s.} [\alpha_{rs} (pq|ss) + \beta_{rs} (ps|qs)], \quad (41)$$

$$\xi_{pq}^r = f_r h_{pq} + \sum_s^{c.d.s.} [\alpha_{rs} (pq|ss) + \beta_{rs} (ps|qs)], \quad (42)$$

and

$$\tau_{rs}^{pq} = 2(\alpha_{rp} - \alpha_{sp})(pq|rs) + (\beta_{rp} - \beta_{sp})[(pr|qs) + (ps|qr)], \quad (43)$$

where α_{pq} and β_{pq} are the Coulomb and exchange coupling constants, respectively. These constants have the values

$$\begin{array}{ccccc} \alpha_{pq} & D & S & V & \beta_{pq} & D & S & V \\ D & 2 & 1 & 0 & D & -1 & -\frac{1}{2} & 0 \\ S & 1 & \frac{1}{2} & 0 & S & -\frac{1}{2} & -\frac{1}{2} & 0 \\ V & 0 & 0 & 0 & V & 0 & 0 & 0 \end{array}, \quad (44)$$

where p and q index DOCC (D), SOCC (S), or VIRT (V) MO's. From the variational conditions, $\zeta_{pq} = \zeta_{qp}$. In addition, $\xi_{pq}^r = \xi_{qp}^r$. If r and p belong to the same shell, $\xi_{pq}^r = \xi_{qp}^r$. For virtual orbital a , $\zeta_{pa} = \zeta_{ap} = 0$, $\xi_{pa}^a = \xi_{ap}^a = 0$, and $\tau_{rs}^{aa} = 0$. As a result, the ζ_{pq} and ξ_{pq}^r terms do not contribute to the diagonal blocks of the $W^{(2)}$ matrix. The expressions for the off-diagonal blocks are also simpler (see Sec. II D).

Formally, $\tau_{rs}^{pq} \neq \tau_{rs}^{qp}$, but $\tau_{rs}^{pq} = \tau_{rs}^{qp}$ if p and q belong to the same shell. Diagonal blocks of $W^{(2)}$ contain terms with the form

$$W_{rs}^{(2)} = - \frac{\delta_{rs}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^r - \delta_{rs}' \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs} \quad (45)$$

where Z_{pq} is the solution to the Z-vector equations (see Sec. III D) and a modified Kronecker delta is defined as

$$\begin{aligned} \delta'_{pq} &= 1, & p > q, \\ \delta'_{pq} &= 0, & p \leq q. \end{aligned} \quad (46)$$

By symmetrizing the terms, Eq. (45) can subsequently be simplified to

$$\begin{aligned} \frac{W_{rs}^{(2)} + W_{sr}^{(2)}}{2} &= -\frac{\delta_{rs}}{4} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rr} - \frac{\delta_{sr}}{4} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{ss} \\ &\quad - \frac{\delta'_{rs}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs} - \frac{\delta'_{sr}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{sr} \\ &= -\frac{\delta_{rs}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rr} - \frac{\delta'_{rs}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs} \\ &\quad - \frac{\delta_{sr}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs} \\ &= -\frac{\delta_{rs}}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rr} - \frac{(1 - \delta_{rs})}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs} \\ &= -\frac{1}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{rs}. \end{aligned} \quad (47)$$

The resulting density-matrix expressions are summarized in Sec. II D.

D. Density-matrix expressions

In this section, the densitylike quantities, χ , $\mathbf{P}^{(2)}$, $\mathbf{W}^{(2)}$, $\mathbf{\Gamma}^S$, and $\mathbf{\Gamma}^{NS}$, and the Lagrangian \mathbf{L} are summarized in terms of the matrix elements of their various shell-shell blocks. G_{pqrs} is defined to be $G_{pqrs} = 4(pq|rs) - (pr|qs) - (ps|qr)$.

• χ :

$$\begin{aligned} \chi_x &= \sum_{i,j} \sum_p^{s.v.} (ix|jp) [C_{xp}(ix|jp) - (ip|jx)] / D_{ij}^{xp} D_{ij}^{xp} \\ &\quad + \sum_{a,b} \sum_p^{d.s.} (xa|pb) [C_{xp}(xa|pb) - (xb|pa)] / D_{xp}^{ab} D_{xp}^{ab} \\ &\quad + \sum_{i,z,a} [(ix|za)^2 / (D_{iz}^{xa})^2 + (iz|xa)^2 / (D_{ix}^{za})^2]. \end{aligned} \quad (48)$$

• CORE-CORE response density:

$$P_{mn}^{(2)} = 0. \quad (49)$$

• DOCC-CORE response density:

$$\begin{aligned} P_{im}^{(2)} &= \frac{1}{(\epsilon_i - \epsilon_m)} \left[2 \sum_j \sum_{p,q}^{s.v.} (mp|jq) [C_{pq}(ip|jq) \right. \\ &\quad - (iq|jp)] / D_{ij}^{pq} + 2 \sum_{a,b} \sum_p^{d.s.} (ma|pb) [2(ia|pb) \\ &\quad - (ib|pa)] / D_{ip}^{ab} + 2 \sum_{x,y} \sum_a (mx|ya)(ix|ya) / D_{iy}^{xa} \\ &\quad \left. + \sum_{x,y,a} (mx|xa)(iy|ya) / D_i^a \right]. \end{aligned} \quad (50)$$

• DOCC-DOCC response density:

$$\begin{aligned} P_{ij}^{(2)} &= -\sum_k \sum_{p,q}^{s.v.} (ip|kq) [C_{pq}(jp|kq) - (jq|kp)] / D_{ik}^{pq} D_{jk}^{pq} \\ &\quad - \sum_{a,b} \sum_p^{d.s.} (ia|pb) [2(ja|pb) - (jb|pa)] / D_{ip}^{ab} D_{jp}^{ab} \\ &\quad - \sum_{x,y} \sum_a (ix|ya)(jx|ya) / D_{iy}^{xa} D_{jy}^{xa} \\ &\quad - \frac{1}{2} \sum_{x,y,a} (ix|xa)(jy|ya) / D_i^a D_j^a. \end{aligned} \quad (51)$$

• SOCC-SOCC response density:

$$\begin{aligned} P_{xy}^{(2)} &= \sum_{i,j} \sum_p^{s.v.} (ix|jp) [C_{yp}(iy|jp) - (ip|jy)] / D_{ij}^{yp} D_{ij}^{yp} - \sum_{a,b} \sum_p^{d.s.} (xa|pb) [C_{yp}(ya|pb) \\ &\quad - (yb|pa)] / D_{xp}^{ab} D_{yp}^{ab} + \sum_{i,z} \sum_a [(ix|za)(iy|za) / D_{iz}^{xa} D_{iz}^{ya} - (iz|xa)(iz|ya) / D_{ix}^{za} D_{iy}^{za}] + \frac{(1 - \delta_{xy})}{2(\epsilon_y - \epsilon_x)} \sum_w [(yw|yw) \\ &\quad - (xw|xw)] \left[\sum_{i,j} \sum_p^{s.v.} (ix|jp) [C_{yp}(iy|jp) - (ip|jy)] / D_{ij}^{yp} D_{ij}^{yp} + \sum_{a,b} \sum_p^{d.s.} (xa|pb) [C_{yp}(ya|pb) - (yb|pa)] / D_{xp}^{ab} D_{yp}^{ab} \right. \\ &\quad \left. + \sum_{i,z} \sum_a [(ix|za)(iy|za) / D_{iz}^{xa} D_{iz}^{ya} + (iz|xa)(iz|ya) / D_{ix}^{za} D_{iy}^{za}] \right] + \frac{(1 - \delta_{xy})}{(\epsilon_y - \epsilon_x)} \sum_z (xz|yz) [\chi_y - \chi_x]. \end{aligned} \quad (52)$$

- VIRT-VIRT response density:

$$P_{ab}^{(2)} = \sum_{i,j} \sum_p^{s.v.} (ia|jp)[2(ib|jp) - (ip|jb)]/D_{ij}^{ap} D_{ij}^{bp} + \sum_c \sum_{p,q}^{d.s.} (pa|qc)[C_{pq}(pb|qc) - (pc|qb)]/D_{pq}^{ac} D_{pq}^{bc} + \sum_{x,y} \sum_i (ix|ya)(ix|yb)/D_{iy}^{xa} D_{iy}^{xb} + \frac{1}{2} \sum_{i,x,y} (ix|xa)(iy|yb)/D_i^a D_i^b. \quad (53)$$

- The off-diagonal i.p. response density for $p > q$ is given by:

$$P_{pq}^{(2)} = (f_p - f_q)Z_{pq}. \quad (54)$$

- CORE-CORE energy-weighted response density:

$$W_{mn}^{(2)} = -\frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqmn} - \frac{1}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{mn}. \quad (55)$$

- DOCC-CORE energy-weighted response density:

$$W_{im}^{(2)} = -\sum_j \sum_{p,q}^{s.v.} (mp|jq)[C_{pq}(ip|jq) - (iq|jp)]/D_{ij}^{pq} - \sum_{a,b} \sum_p^{d.s.} (ma|pb)[2(ia|pb) - (ib|pa)]/D_{ip}^{ab} - \sum_{x,y} \sum_a (mx|ya)(ix|ya)/D_{iy}^{xa} - \frac{1}{2} \sum_{x,y,a} (mx|xa)(iy|ya)/D_i^a - \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqim} - P_{im}^{(2)} \frac{\epsilon_m + \epsilon_i}{2} - \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{im}. \quad (56)$$

- DOCC-DOCC energy-weighted response density:

$$W_{ij}^{(2)} = -\sum_k \sum_{p,q}^{s.v.} (ip|kq)[C_{pq}(jp|kq) - (jq|kp)]/D_{jk}^{pq} - \sum_{a,b} \sum_p^{d.s.} (ia|pb)[2(ja|pb) - (jb|pa)]/D_{jp}^{ab} - \sum_{x,y} \sum_a (ix|ya)(jx|ya)/D_{jy}^{xa} - \frac{1}{2} \sum_{x,y,a} (ix|xa)(jy|ya)/D_j^a - \frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqij} - P_{ij}^{(2)} \epsilon_j - \frac{1}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{ij}. \quad (57)$$

- SOCC-SOCC energy-weighted response density:

$$W_{xy}^{(2)} = -\sum_{i,j} \sum_p^{s.v.} (ix|jp)[C_{yp}(iy|jp) - (ip|jy)]/D_{ij}^{yp} - \sum_{a,b} \sum_p^{d.s.} (xa|pb)[C_{yp}(ya|pb) - (yb|pa)]/D_{yp}^{ab} - \sum_{z,a} \sum_i [(ix|za)(iy|za)/D_{iz}^{ya} + (iz|xa)(iz|ya)/D_{iy}^{za}] - \sum_{i,z,a} (ix|ya)(iz|za)/D_i^a - \frac{1}{4} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqxy} - P_{xy}^{(2)} \epsilon_y - \frac{1}{2} \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{xy} - \frac{\delta_{xy}}{2} \sum_z (xz|xz)[\chi_x + \chi_z] + \frac{\delta'_{yx}}{2} \sum_w [(yw|yw) - (xw|xw)] \left[\sum_{i,j} \sum_p^{s.v.} (ix|jp)[C_{yp}(iy|jp) - (ip|jy)]/D_{ij}^{yp} + \sum_{a,b} \sum_p^{d.s.} (xa|pb)[C_{yp}(ya|pb) - (yb|pa)]/D_{yp}^{ab} + \sum_{z,a} \sum_i [(ix|za)(iy|za)/D_{iz}^{ya} D_{iz}^{ya} + (iz|xa)(iz|ya)/D_{ix}^{za} D_{iy}^{za}] \right]. \quad (58)$$

- VIRT-VIRT energy-weighted response density:

$$W_{ab}^{(2)} = -\sum_{i,j} \sum_p^{s.v.} (ia|jp)[2(ib|jp) - (ip|jb)]/D_{ij}^{pb} - \sum_c \sum_{p,q}^{d.s.} (pa|qc)[C_{pq}(pb|qc) - (pc|qb)]/D_{pq}^{bc} - \sum_{x,y} \sum_i (ix|ya)(ix|yb)/D_{iy}^{xb} - \frac{1}{2} \sum_{i,x,y} (ix|xa)(iy|yb)/D_i^b - P_{ab}^{(2)} \epsilon_b. \quad (59)$$

- SOCC-CORE and SOCC-DOCC energy-weighted response densities:

$$\begin{aligned}
 W_{xr}^{(2)} = & -2 \sum_{i,j} \sum_p^{s.v.} (ir|jp)[C_{xp}(ix|jp) - (ip|jx)]/D_{ij}^{xp} - 2 \sum_{a,b} \sum_p^{d.s.} (ra|pb)[C_{xp}(xa|pb) - (xb|pa)]/D_{xp}^{ab} - 2 \sum_{y,a} \sum_i [(ir|ya) \\
 & \times (ix|ya)/D_{iy}^{xa} + (iy|ra)(iy|xa)/D_{ix}^{ya}] - \sum_{i,y,a} [(ix|ra) + (ir|xa)](iy|ya)/D_i^a - \frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqxr} - 2 \sum_y P_{xy}^{(2)} \epsilon_{yr} - \sum_{p>q}^{i.p.} Z_{pq} \tau_{pq}^{xr} \\
 & - \sum_i^{c.d.} Z_{xi} (\xi_{ri}^x - \zeta_{ir}) + \sum_a Z_{ax} \xi_{ra}^x - \sum_z (rz|xz)[\chi_x + \chi_z], \tag{60}
 \end{aligned}$$

where r indexes a CORE or a DOCC MO.

- VIRT-CORE, VIRT-DOCC, and VIRT-SOCC energy-weighted response densities:

$$\begin{aligned}
 W_{ar}^{(2)} = & -2 \sum_{i,j} \sum_p^{s.v.} (ir|jp)[2(ia|jp) - (ip|ja)]/D_{ij}^{ap} - 2 \sum_b \sum_{p,q}^{d.s.} (pr|qb)[C_{pq}(pa|qb) - (pb|qa)]/D_{pq}^{ab} - 2 \sum_{x,y} \sum_i (ix|yr)(ix|ya)/D_{iy}^{xa} \\
 & - \sum_{i,x,y} (ix|xr)(iy|ya)/D_i^a - 2 \sum_b P_{ab}^{(2)} \epsilon_{br} + \sum_i^{c.d.} Z_{ai} \zeta_{ir} + \sum_x Z_{ax} \zeta_{xr}, \tag{61}
 \end{aligned}$$

where r indexes a CORE, DOCC, or SOCC MO.

- SOCC-CORE Lagrangian elements:

$$\begin{aligned}
 L_{xm} = & -2 \sum_{i,j} \sum_p^{s.v.} (im|jp)[C_{xp}(ix|jp) - (ip|jx)]/D_{ij}^{xp} - 2 \sum_{a,b} \sum_p^{d.s.} (ma|pb)[C_{xp}(xa|pb) - (xb|pa)]/D_{xp}^{ab} - 2 \sum_{y,a} \sum_i [(im|ya) \\
 & \times (ix|ya)/D_{iy}^{xa} + (iy|ma)(iy|xa)/D_{ix}^{ya}] - \sum_{i,y,a} [(ix|ma) + (im|xa)](iy|ya)/D_i^a - \sum_y (my|xy)[\chi_x + \chi_y] + \frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqxm} \\
 & + 2 \sum_i^{c.d.} P_{im}^{(2)} \epsilon_{xi} - 2 \sum_y P_{xy}^{(2)} \epsilon_{ym}. \tag{62}
 \end{aligned}$$

- VIRT-CORE Lagrangian elements:

$$\begin{aligned}
 L_{am} = & -2 \sum_{i,j} \sum_p^{s.v.} (im|jp)[2(ia|jp) - (ip|ja)]/D_{ij}^{ap} - 2 \sum_b \sum_{p,q}^{d.s.} (pm|qb)[C_{pq}(pa|qb) - (pb|qa)]/D_{pq}^{ab} - 2 \sum_{x,y} \sum_i (ix|ym)(ix|ya)/D_{iy}^{xa} \\
 & - \sum_{i,x,y} (ix|xm)(iy|ya)/D_i^a + \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqam} + 2 \sum_i^{c.d.} P_{im}^{(2)} \epsilon_{ai} - 2 \sum_b P_{ab}^{(2)} \epsilon_{bm}. \tag{63}
 \end{aligned}$$

- SOCC-DOCC Lagrangian elements:

$$\begin{aligned}
 L_{xi} = & 2 \sum_j \sum_{p,q}^{s.v.} (xp|jq)[C_{pq}(ip|jq) - (iq|jp)]/D_{ij}^{pq} - 2 \sum_{j,k} \sum_p^{s.v.} (ik|jp)[C_{xp}(kx|jp) - (kp|jx)]/D_{jk}^{xp} + 2 \sum_{a,b} \sum_p^{d.s.} (xa|pb)[2(ia|pb) \\
 & - (ib|pa)]/D_{ip}^{ab} - 2 \sum_{a,b} \sum_p^{d.s.} (ia|pb)[C_{xp}(xa|pb) - (xb|pa)]/D_{xp}^{ab} + 2 \sum_{y,z} \sum_a (xy|za)(iy|za)/D_{iz}^{ya} - 2 \sum_{y,a} \sum_j [(ij|ya)(jx|ya)]/D_{jy}^{xa} \\
 & + (jy|ia)(jy|xa)/D_{jx}^{ya} + \sum_{y,z,a} (xy|ya)(iz|za)/D_i^a - \sum_{j,y,a} [(jx|ia) + (ij|xa)](jy|ya)/D_j^a - \sum_z (iz|xz)[\chi_x + \chi_z] + \frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqxi} \\
 & + 2 \sum_j^{c.d.} P_{ij}^{(s)} \epsilon_{xj} - 2 \sum_y P_{xy}^{(2)} \epsilon_{yi}. \tag{64}
 \end{aligned}$$

- VIRT-DOCC Lagrangian elements:

$$\begin{aligned}
 L_{ai} = & 2 \sum_j \sum_{p,q}^{s.v.} (pa|jq)[C_{pq}(ip|jq) - (iq|jp)]/D_{ij}^{pq} - 2 \sum_{j,k} \sum_p^{s.v.} (ik|jp)[2(ka|jp) - (kp|ja)]/D_{jk}^{qp} + 2 \sum_{b,c} \sum_p^{d.s.} (ab|pc)[2(ib|pc) \\
 & - (ic|pb)]/D_{ip}^{bc} - 2 \sum_b \sum_{p,q}^{d.s.} (ip|qb)[C_{pq}(pa|qb) - (pb|qa)]/D_{pq}^{ab} + 2 \sum_{x,y} \sum_b (xa|yb)(ix|yb)/D_{iy}^{xb} - 2 \sum_{x,y} \sum_j (iy|jx)(jx|ya)/D_{jy}^{xa} \\
 & + \sum_{x,y,b} (xa|xb)(iy|yb)/D_i^b - \sum_{j,x,y} (ix|jx)(jy|ya)/D_j^a + \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqai} + 2 \sum_j^{c.d.} P_{ij}^{(2)} \epsilon_{aj} - 2 \sum_b P_{ab}^{(2)} \epsilon_{bi}. \quad (65)
 \end{aligned}$$

- VIRT-SOCC Lagrangian elements:

$$\begin{aligned}
 L_{ax} = & 2 \sum_{i,j} \sum_p^{s.v.} (ia|jp)[C_{xp}(ix|jp) - (ip|jx)]/D_{ij}^{xp} - 2 \sum_{i,j} \sum_p^{s.v.} (ix|jp)[2(ia|jp) - (ip|ja)]/D_{ij}^{qp} + 2 \sum_{b,c} \sum_p^{d.s.} (ab|pc)[C_{xp}(xb|pc) \\
 & - (xc|pb)]/D_{xp}^{bc} - 2 \sum_b \sum_{p,q}^{d.s.} (xp|qb)[C_{pq}(pa|qb) - (pb|qa)]/D_{pq}^{ab} + 2 \sum_{i,y} \sum_b [(ia|yb)(ix|yb)/D_{iy}^{xb} + (iy|ab)(iy|xb)/D_{ix}^{yb}] \\
 & - 2 \sum_{y,z} \sum_i (iz|xy)(iz|ya)/D_{iy}^{za} + \sum_{i,y,b} [(ix|ab) + (ia|xb)](iy|yb)/D_i^b - \sum_{i,y,z} (iy|xy)(iz|za)/D_i^a + \frac{1}{2} \sum_{p,q}^{d.p.} P_{pq}^{(2)} G_{pqax} + 2 \sum_y P_{xy}^{(2)} \epsilon_{ay} \\
 & - 2 \sum_b P_{ab}^{(2)} \epsilon_{bx} + \sum_y (xy|ya)[\chi_x + \chi_y]. \quad (66)
 \end{aligned}$$

- Separable two-particle density:

$$\begin{aligned}
 \Gamma_{\mu\nu\lambda\sigma}^S = & P_{\mu\nu}^{(2)} P_{\lambda\sigma}^{SCF} - \frac{1}{2} P_{\mu\lambda}^{(2)} P_{\nu\sigma}^{SCF} + \frac{1}{2} T_{\mu\sigma}^{(4)} P_{\nu\lambda}^S + \frac{1}{4} \chi_{\nu\sigma} P_{\mu\lambda}^S \\
 & + \frac{1}{4} \chi_{\mu\lambda} P_{\nu\sigma}^S + \frac{1}{4} (Z_{\mu\lambda}^{VS} - Z_{\mu\lambda}^{SD}) P_{\nu\sigma}^S, \quad (67)
 \end{aligned}$$

where

$$T_{\mu\nu}^{(4)} = 2 \sum_{ia} C_{\mu i} C_{\nu a} T_{ia}^{(4)}, \quad T_{ia}^{(4)} = \sum_y (iy|ya)/D_i^a, \quad (68)$$

$$\chi_{\mu\nu} = \sum_x C_{\mu x} C_{\nu x} \chi_x, \quad (69)$$

$$Z_{\mu\nu}^{VS} = \sum_{ax} C_{\mu a} C_{\nu x} Z_{ax}, \quad Z_{\mu\nu}^{SD} = \sum_{xi} C_{\mu x} C_{\nu i} Z_{xi}. \quad (70)$$

- Nonseparable two-particle density:

$$\Gamma_{\mu\nu\lambda\sigma}^{NS} = \sum_{p,q}^{d.s.} \sum_{r,s}^{s.v.} C_{\mu p} C_{\nu r} C_{\lambda q} C_{\sigma s} T_{pq}^{rs}, \quad (71)$$

where

$$T_{ij}^{ab} = 2[2(ia|jb) - (ib|ja)]/D_{ij}^{ab},$$

$$T_{ij}^{ax} = [2(ia|jx) - (ix|ja)]/D_{ij}^{ax},$$

$$T_{ij}^{xa} = [2(ix|ja) - (ia|jx)]/D_{ij}^{xa},$$

$$T_{ij}^{xy} = [(ix|jy) - (iy|jx)]/D_{ij}^{xy},$$

$$T_{iy}^{xa} = 2(ix|ya)/D_{iy}^{xa}, \quad (72)$$

$$T_{ix}^{ab} = [2(ia|xb) - (ib|xa)]/D_{ix}^{ab},$$

$$T_{xi}^{ab} = [2(xa|ib) - (xb|ia)]/D_{xi}^{ab},$$

$$T_{xy}^{ab} = [(xa|yb) - (xb|ya)]/D_{xy}^{ab},$$

$$T_{ix}^{yz} = T_{xi}^{yz} = T_{xy}^{za} = T_{xy}^{az} = T_{wx}^{yz} = 0.$$

III. PARALLEL ZAPT2 GRADIENT ALGORITHM

The implementation of the parallel ZAPT2 gradient algorithm is organized below into the following subsections: (1) creation of distributed integral arrays, (2) construction of immediately obtainable terms of $\mathbf{P}^{(2)}$, $\mathbf{W}^{(2)}$, and \mathbf{L} , (3) three-virtual integrals and the completion of \mathbf{L} , (4) solution of the Z-vector equations, (5) completion of the one-particle density matrices and computation of the one-particle gradient, and (6) back-transformation of amplitudes to generate the nonseparable density matrix and calculation of the two-particle gradient.

A. Distributed integral arrays

The equations for the density matrices in Sec. II D require five classes of integrals. Using “*v*” to denote a virtual MO (VIRT) and “*o*” to denote an occupied MO (CORE, DOCC, or SOCC), these classes are (1) (*oo|oo*), (2) (*vo|oo*), (3) (*vv|oo*), (4) (*vo|vo*), and (5) (*vv|vo*).

The first four integral classes are created and stored in distributed memory as described previously.²³ The fifth class is the largest and is only used in the construction of the Lagrangian, so the terms involving these integrals are calculated in a direct fashion as described in Sec. III C.

In addition to the integral arrays, the trial vectors of the Z-vector equation are distributed across the nodes. The density matrices ($\mathbf{P}^{(2)}$ and $\mathbf{W}^{(2)}$) and other data of order n^2 or less are stored in a replicated fashion.

B. Density-matrix creation

The integral component $\frac{1}{2}\sum_y(py|py)$ in Eq. (6) distinguishes ZAPT2 from other related methods. Since these values will be used often in the creation of the density matrices, an array of integral components is stored in replicated memory. These terms require (*xy|xy*) integrals, which are also used in the multiplicative factor in term 4 of P_{xy} and term 9 of W_{xy} . This factor is stored in an intermediate matrix,

$$F_{xy} = \frac{1}{2} \sum_w [(yw|yw) - (xw|xw)]. \quad (73)$$

Each processor reads local integrals and computes a contribution to both arrays. These contributions are then globally summed.

The algorithm for density-matrix term creation is structured so as to reduce communication needs as much as possible. In general, a set of integrals is read from local or remote processors, amplitudes are formed by subtracting appropriate integrals and dividing by an orbital energy denominator, and suitable integrals and amplitudes are contracted (multiplied) in order to create each term of the density-matrix blocks. More details about the structure of this algorithm are available.³⁷ During the density-matrix creation, the ZAPT2 energy can be calculated according to

$$E_{\text{ZAPT}}^{(2)} = -\frac{1}{4} \sum_p^{d.s.v.} W_{pp}^{(2)}, \quad (74)$$

where $W_{pp}^{(2)}$ includes terms 1–4 of $W_{ii}^{(2)}$, $W_{xx}^{(2)}$, and $W_{aa}^{(2)}$ [Eqs. (57)–(59)].

C. Three-virtual terms

The (*vv|vo*) orbitals are required for terms 1-VV (1-VV refers to $p \in \text{VIRT}$ and $q \in \text{VIRT}$ in term 1), 3, and 9 of L_{ai} [Eq. (65)] and terms 3 and 9 of L_{ax} [Eq. (66)]. As discussed previously,^{23,27,38} these terms can be written in a mixed MO/AO basis and consequently use quarter-transformed ERI's with one occupied MO index and three AO indices that are computed in a direct fashion. Terms 1-VV and 3 of L_{ai} and term 3 of L_{ax} may be rearranged as

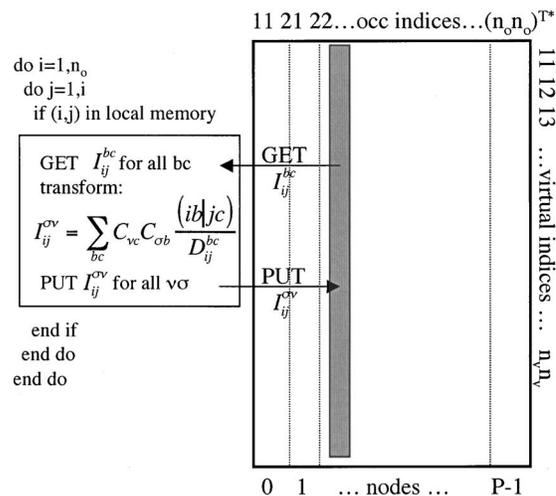


FIG. 1. First half-back-transformation of the (*vo|vo*) integrals. This procedure is used to form half-back-transformed integrals in the creation of (*vv|vo*) integrals. An analogous procedure yields half-back-transformed amplitudes for the creation of the nonseparable two-particle density matrix. No communication is required in this step.

$$\begin{aligned} L_{ar} &= 2 \sum_{b,c}^{d.s.} \sum_p (ab|pc) [C_{rp}(rb|pc) - (rc|pb)] / D_{rp}^{bc} \\ &= 2 \sum_{b,c}^{d.s.} \sum_p \sum_{\nu\lambda\sigma} C_{vc} C_{\lambda a} C_{ob} (p\nu|\lambda\sigma) [C_{rp}(rb|pc) \\ &\quad - (rc|pb)] / D_{rp}^{bc} \\ &= 2 \sum_p^{d.s.} \sum_{\nu\lambda\sigma} C_{\lambda a} (p\nu|\lambda\sigma) [C_{rp} I_{rp}^{\sigma\nu} - I_{rp}^{\nu\sigma}] / D_{rp}^{\sigma\nu}, \end{aligned} \quad (75)$$

where the half-transformed integral is

$$I_{rp}^{\sigma\nu} = \sum_{b,c} C_{vc} C_{ob} (rb|pc) / D_{rp}^{bc}. \quad (76)$$

Then, a mixed MO/AO Lagrangian may be formed as follows:

$$L_{ar} = \sum_{\lambda} C_{\lambda a} L_{\lambda r} = 2 \sum_p^{d.s.} \sum_{\nu\lambda\sigma} C_{\lambda a} (p\nu|\lambda\sigma) [C_{rp} I_{rp}^{\sigma\nu} - I_{rp}^{\nu\sigma}] / D_{rp}^{\sigma\nu}, \quad (77)$$

$$L_{\lambda r} = 2 \sum_p^{d.s.} \sum_{\nu\sigma} (p\nu|\lambda\sigma) [C_{rp} I_{rp}^{\sigma\nu} - I_{rp}^{\nu\sigma}] / D_{rp}^{\sigma\nu}. \quad (78)$$

Term 9 of L_{ai} and L_{ax} may be similarly rearranged to yield

$$L_{vr} = \sum_{\lambda\sigma} (r\nu|\lambda\sigma) P_{\lambda\sigma}^{(2)}. \quad (79)$$

Half-transformed integrals are generated from (*vo|vo*) integrals. The transformation is done locally, so no communication is required (Fig. 1). In order to save memory, the half-transformed integrals overwrite the (*vo|vo*) integrals. The quarter-transformed integrals are generated as described previously.²³ After the inner two loops, half-transformed integrals for a given ν, σ pair are read, which requires communication among the processors (Fig. 2). Once the mixed

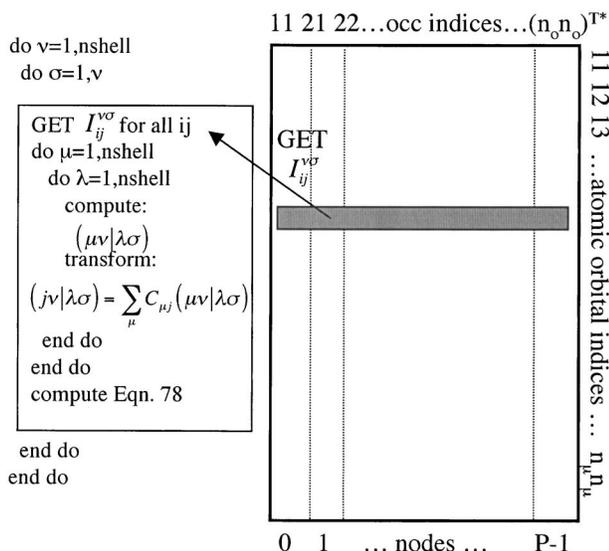


FIG. 2. Algorithm for the creation of the Lagrangian $(vv|vo)$ terms. This step requires communication across the processors in order to read in the half-back-transformed integrals.

MO/AO Lagrangian terms are created, these terms are transformed to the MO basis. The complete Lagrangian is globally summed so that each processor holds the full matrix. Then, the $(vo|vo)$ integrals are restored by transforming with the inverse MO coefficient matrix $C^{-1}=C^T S$, where S is the overlap matrix over AO's. Multiplication by the appropriate orbital-energy factor (D_{rp}^{bc}) recreates the original $(vo|vo)$ integrals. This step requires no communication.

D. Solution of the CPHF equations

The general restricted open-shell coupled-perturbed Hartree-Fock (CPHF) equations must be solved for the response vectors. The Z-vector substitution is used,¹⁹ where Z is the solution to the linear equation

$$\sum_{p,q}^{\text{i.p.}} A'_{pqrs} Z_{pq} = L_{rs}. \quad (80)$$

A reduced Lagrangian L_{rs} containing only symmetry-allowed elements is used in the solution of (80). The ZAPT2 orbital Hessian A'_{pqrs} has more terms than the closed-shell form; this may be visualized in Fig. 3. The equations governing the six unique matrix blocks may be written as

$$\begin{array}{c}
 \begin{array}{ccc}
 & yj & bj & by \\
 xi & \begin{array}{c} A_{xij} \\ (ooloo) \end{array} & \begin{array}{c} A_{xibj} \\ (voloo) \end{array} & \begin{array}{c} A_{xiby} \\ (voloo) \end{array} \\
 ai & \begin{array}{c} A_{aij} \\ (voloo) \end{array} & \begin{array}{c} A_{aibj} \\ (vvloo) \\ (volvo) \end{array} & \begin{array}{c} A_{aiby} \\ (vvloo) \\ (volvo) \end{array} \\
 ax & \begin{array}{c} A_{axyj} \\ (voloo) \end{array} & \begin{array}{c} A_{aiby} \\ (vvloo) \\ (volvo) \end{array} & \begin{array}{c} A_{axby} \\ (vvloo) \\ (volvo) \end{array}
 \end{array}
 \times
 \begin{array}{c}
 yj \\
 bj \\
 by
 \end{array}
 =
 \begin{array}{c}
 xi \\
 ai \\
 ax
 \end{array}
 \begin{array}{c}
 L_{xi} \\
 L_{ai} \\
 L_{ax}
 \end{array}
 \end{array}$$

FIG. 3. Pictorial representation of the Z-vector equations.

$$\begin{aligned}
 A'_{xij} &= (xi|yj) - \frac{1}{2}(xj|yi) - \frac{1}{2}(xy|ij) + \delta_{xy}(\xi_{ij}^x - \zeta_{ji}) \\
 &\quad + \delta_{ij}(\xi_{xy}^i - \zeta_{yx}), \\
 A'_{aij} &= 2(ai|yj) - \frac{1}{2}(aj|yi) - \frac{1}{2}(ay|ij) + \delta_{ij}\xi_{ay}^i, \\
 A'_{axyj} &= (ax|yj) - \delta_{xy}\xi_{aj}^x, \\
 A'_{aibj} &= 4(ai|bj) - (aj|bi) - (ab|ij) - \delta_{ab}\zeta_{ji} + \delta_{ij}\xi_{ab}^i, \\
 A'_{axbj} &= 2(ax|bj) - \frac{1}{2}(aj|bx) - \frac{1}{2}(ab|xj) - \delta_{ab}\zeta_{jx}, \\
 A'_{axby} &= (ax|by) - \frac{1}{2}(ay|bx) - \frac{1}{2}(ab|xy) - \delta_{ab}\zeta_{yx} + \delta_{xy}\xi_{ab}^x.
 \end{aligned} \quad (81)$$

The notation A_{pqrs} (without a prime) will be used to indicate the integral-only part of the orbital Hessian (no ξ_{pq}^r or ζ_{pq} contributions). Portions of the ξ_{pq}^r and ζ_{pq} values are formed from locally held integrals and then summed onto all processors. The explicit storage of the orbital Hessian is impractical for systems with a large number of orbitals, so the CPHF equations are solved iteratively. In each step of the iterations, blocks of the orbital Hessian are created from locally held integrals. For A_{aibj} (the closed-shell-like block), A_{axbj} , and A_{axby} , the $(vo|vo)$ and $(vv|oo)$ integrals reside on the same processor for a given oo pair. A local contribution to the orbital Hessian is constructed and subsequently multiplied by the appropriate section of the trial vector in order to yield a segment of the new vector. For A_{xij} , a similar procedure is followed for the $(xy|ij)$ integrals and for the $(xi|yj)$ integrals. Contributions to the orbital Hessian are formed from local $(vo|oo)$ integrals for the A_{aij} and A_{axyj} blocks.

For closed-shell systems, the linear equations are

$$\sum_b \sum_j \{A_{aibj} + \delta_{ab}\delta_{ij}(\epsilon_b - \epsilon_j)\} Z_{bj} = L_{ai}. \quad (82)$$

Normally, the orbital energy term is taken outside the summation to yield

$$(\epsilon_a - \epsilon_i)Z_{ai} + \sum_b \sum_j A_{aibj} Z_{bj} = L_{ai}. \quad (83)$$

This equation is divided by the orbital energy terms to yield a "preconditioned" equation,

$$Z_{ai} + \sum_b \sum_j A_{aibj} \frac{Z_{bj}}{(\varepsilon_a - \varepsilon_i)} = \frac{L_{ai}}{(\varepsilon_a - \varepsilon_i)}, \quad (84)$$

whose diagonal element $[A_{aiai}/(\varepsilon_a - \varepsilon_i)]$ are of the order of unity, and then this is solved iteratively for Z_{ai} . In order to restructure the ZAPT2 equations and generate a similar form, a quantity m_{pq} is added and subtracted in diagonal elements of Eq. (80),

$$\sum_{p,q}^{\text{i.p.}} \{A_{pqrs} + \delta_{pr}(\xi_{ps}^p - \zeta_{sq}) + \delta_{qs}(\xi_{pr}^q - \zeta_{rp}) - m_{pq}\delta_{pr}\delta_{qs}\} Z_{pq} = L_{rs}, \quad (85)$$

where

$$m_{xi} = \xi_{ii}^x + \xi_{xx}^i - \zeta_{ii} - \zeta_{xx}, \quad (86)$$

$$m_{ai} = \xi_{aa}^i - \zeta_{ii} = \varepsilon_a - \varepsilon_i,$$

$$m_{ax} = \xi_{aa}^x - \zeta_{xx}.$$

The last term in Eq. (85) is separated out,

$$\sum_{p,q}^{\text{i.p.}} Z_{pq} m_{pq} \delta_{pr} \delta_{qs} + \sum_{p,q}^{\text{i.p.}} \{A_{pqrs} + \delta_{pr}(\xi_{qs}^p - \zeta_{sq}) + \delta_{qs}(\xi_{pr}^q - \zeta_{rp}) - m_{pq}\delta_{pr}\delta_{qs}\} Z_{pq} = L_{rs}, \quad (87)$$

and all terms are divided by m_{rs} to yield

$$Z_{rs} + \sum_{p,q}^{\text{i.p.}} \{A_{pqrs} + \delta_{pr}(\xi_{qs}^p - \zeta_{sq}) + \delta_{qs}(\xi_{pr}^q - \zeta_{rp}) - m_{pq}\delta_{pr}\delta_{qs}\} \frac{Z_{pq}}{m_{rs}} = \frac{L_{rs}}{m_{rs}}. \quad (88)$$

This can be solved iteratively for Z_{rs} . The subtraction of and division by m_{rs} is a necessary preconditioning in order to keep the diagonal elements close to 1 and induce convergence of the iterative equations.

E. Completion of the one-particle density matrices

Once Z_{rs} is known, the remaining terms in the one-particle density matrices may be finished. The off-diagonal blocks of $\mathbf{P}^{(2)}$ are determined by Eq. (54). Terms 8 and 9 of W_{xr} and terms 6 and 7 of W_{ar} are computed from the ξ and ζ values. Contributions to terms 1 and 2 of W_{mn} and terms 5 and 7 of W_{im} , W_{ij} , W_{xr} , and W_{xy} are created from local integrals and then globally summed. The $(vv|oo)$ integrals are no longer required, so the memory for these is released. The remaining integrals are needed in the two-particle nonseparable density matrix, so these are retained in memory. The completed $\mathbf{P}^{(2)}$ and $\mathbf{W}^{(2)}$ matrices are back transformed to the AO basis, combined with their SCF analogs, and contracted with the appropriate derivative integrals.

F. Two-particle gradient

The four-index back transformation of the amplitude terms in $\mathbf{\Gamma}^{\text{NS}}$ is similar to the procedure described previously.²³ For the closed-shell case, T_{ij}^{ab} amplitudes

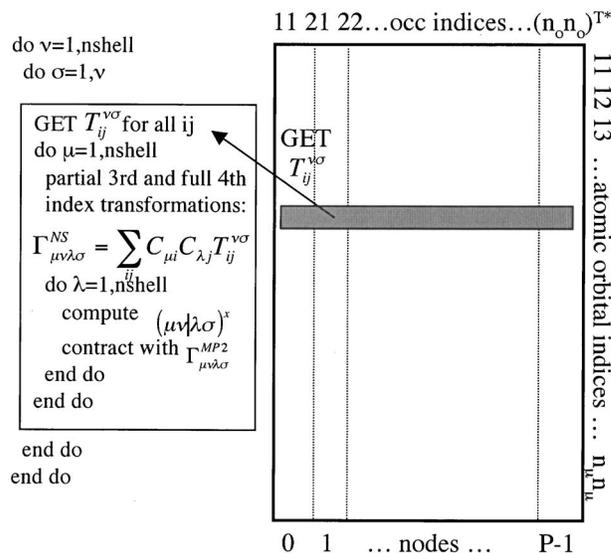


FIG. 4. Second half-back-transformation of the amplitudes for the creation of the nonseparable two-particle density matrix. This step requires communication across the processors in order to retrieve the half-back-transformed amplitudes.

formed from $(vo|vo)$ integrals that are locally held for a given oo pair are half-back-transformed to yield amplitudes with the form $T_{ij}^{\sigma\nu}$, which overwrite the $(vo|vo)$ integrals in storage. For ZAPT2, an analogous procedure is employed for the closed-shell-like terms T_{ij}^{ab} , T_{ix}^{ab} , T_{xi}^{ab} , and T_{xy}^{ab} to yield amplitudes with the form $T_{pq}^{\sigma\nu}$, where p and q are occupied (DOCC and SOCC) indices. This part of the half-back-transformation requires no communication. The T_{ij}^{ax} , T_{ij}^{xa} , T_{ij}^{xy} , and T_{iy}^{xa} amplitudes can also be transformed to yield $T_{pq}^{\sigma\nu}$ amplitudes, which can be added to the half-back-transformed closed-shell-like amplitudes in the $(vo|vo)$ storage. Since the two occupied indices that have not been back transformed do not constitute a local pair in either the $(vo|oo)$ or $(oo|oo)$ integral storage, communication between processors must occur. The number of amplitudes that must be transmitted will likely be small compared to the number of closed-shell-like amplitudes since these amplitudes involve at least one singly occupied index. Once these amplitudes have been created, the memory for the $(vo|oo)$ and $(oo|oo)$ integrals is released.

The separable two-particle density matrix for ZAPT2 includes terms that come from the back transformation of Z_{xi} , Z_{ax} , χ_x , and \mathbf{P}^S . These two-index transformations are not done in parallel since they are fairly trivial. A fourfold loop over shells is used to calculate the derivative ERI's. Inside the outer two loops, a set of $T_{pq}^{\sigma\nu}$ amplitudes is read in for a given σ, ν pair, which requires communication between the processors (Fig. 4). Inside the third and fourth loops, the p and q indices are back transformed to the AO basis. These steps are done locally. Inside the loops, the nonseparable and separable second-order two-particle density matrices are created, added to their SCF analog, and contracted with the appropriate derivative integrals. Since $\mathbf{\Gamma}^{\text{NS}}$ is not symmetrized, the derivative ERI's are computed four times more than the minimal list. This increased n^4 computation eliminates the need for n^4 storage and n^5 effort in the symmetrization.

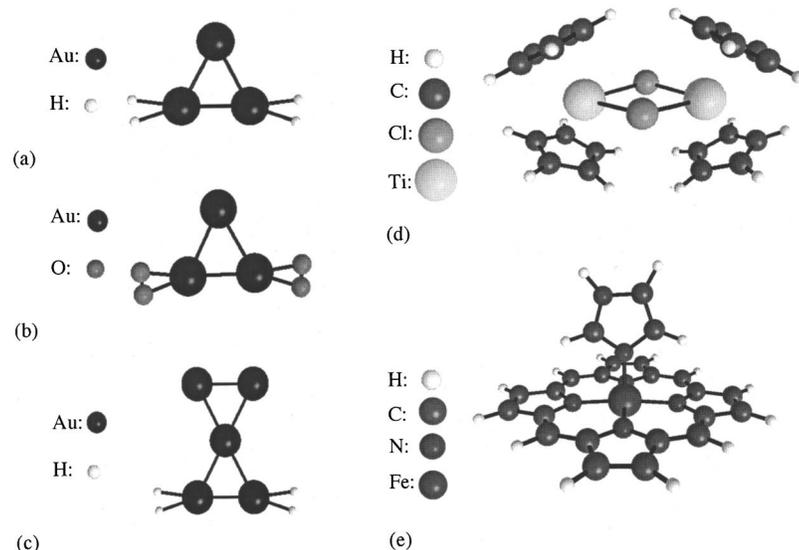


FIG. 5. (a) Structure of Au_3H_4 . (b) Structure of Au_3O_4 . (c) Structure of Au_5H_4 . (d) Structure of $\text{Ti}_2\text{Cl}_2\text{Cp}_4$. (e) Structure of iron-porphyrin:imidazole.

IV. TIMINGS

In this section, the scalability of the ZAPT2 gradient algorithm described in Sec. III is examined. All calculations were performed on a cluster system comprised of IBM p640 nodes connected by dual Gigabit Ethernet. Each p640 node has four 375 MHz Power3-II processors and 16 Gbytes of memory.

Five molecular systems are considered in the benchmark calculations: three small gold clusters (Au_3H_4 , Au_3O_4 , and Au_5H_4), a dimeric dicyclopentadienyltitanium(III) chloride system, and a Fe-porphyrin:imidazole system. The smallest molecule under consideration is Au_3H_4 [Fig. 5(a)]. The basis set used on this molecule consists of the aug-cc-pVTZ (Ref. 39) basis set on H and the uncontracted Stevens-Basch-Krauss-Jasien-Cundari⁴⁰ (SBKJC) effective core potential basis set, augmented with a set of $3f2g$ polarization functions and one set of sp diffuse functions on Au, for a total of 380 spherical harmonic basis functions. This molecule has 31 doubly occupied orbitals and one singly occupied orbital. A ZAPT2 gradient calculation requires 9.5 MWords of replicated memory and 170 MWords of distributed memory, so it fits in the memory of a single processor on the IBM cluster. For Au_3O_4 [Fig. 5(b)], the aug-cc-pVTZ (Refs. 39 and 41) basis set is used on O, and the uncontracted SBKJC basis set with $3f2g$ polarization functions and one sp diffuse function is used on Au, for a total of 472 spherical harmonic basis functions.

Calculations on this molecule involve 44 doubly occupied orbitals and one singly occupied orbital. A ZAPT2 gradient calculation requires 20.7 MWords of replicated memory and 562 MWords of distributed memory, so the memory allotted to two nodes of the IBM cluster is necessary. Calculations on Au_5H_4 [Fig. 5(c)] employ the same basis set as that used for Au_3H_4 . A total of 572 spherical harmonic basis functions are used in the ZAPT2 gradient calculations. This molecule has 49 doubly occupied orbitals and one singly occupied orbital. Each calculation requires 30.1 MWords of replicated memory and 1011 MWords of distributed memory. The memory allotted to three processors

of the IBM cluster is sufficient to run a calculation on this system. For the $\text{Ti}_2\text{Cl}_2\text{Cp}_4$ system [Fig. 5(d)], the TZV basis set as defined in GAMESS (Ref. 33) was employed, yielding 486 basis functions. A ZAPT2 gradient calculation on the lowest-energy triplet state of this system involves 108 doubly occupied orbitals and two singly occupied orbitals and requires 30.5 MWords of replicated memory and 2470 MWords of distributed memory. Six processors of the IBM cluster are required in order to run a calculation on this system. For the iron-porphyrin:imidazole system [Fig. 5(e)], two basis sets were used. The smaller basis set consists of the MIDI basis set⁴² with d polarization functions, and a calculation with this basis set has 493 basis functions. The larger basis set consists of the TZV basis set with d and p polarization functions, and a calculation with this basis set has 728 basis functions. For the triplet state, 110 orbitals are doubly occupied and two orbitals are singly occupied. A ZAPT2 gradient calculation with the smaller basis set requires 32.1 MWords of replicated memory and 2635 MWords of distributed memory, which corresponds to the memory allotted to seven processors on the IBM system. A similar calculation with the larger basis set requires 52.1 MWords of replicated memory and 5536 MWords of distributed memory, which corresponds to memory allotted to 15 processors on the IBM system.

TABLE I. Wall clock time (seconds) for ZAPT2 gradient step on IBM cluster.

	Au_3H_4	Au_3O_4	Au_5H_4	$\text{Ti}_2\text{Cl}_2\text{Cp}_4$	Fe-porphyrin	Fe-porphyrin
n	380	472	570	486	493	728
P						
1	54 943					
2	28 292	68 645				
4	13 904	35 181	105 567			
8	6 975	18 324	53 171	24 852	65 042	
16	3 596	9 422	26 690	12 205	31 866	232 551
32	1 895	4 965	13 858	6 859	16 671	96 404
64	1 087	2 843	7 453	4 214	9 437	55 019

TABLE II. Speedup for ZAPT2 gradient step on IBM cluster.

n	Au ₃ H ₄	Au ₃ O ₄	Au ₅ H ₄	Ti ₂ Cl ₂ Cp ₄	Fe-porphyrin	Fe-porphyrin
	380	472	570	486	493	728
P						
1	1.0					
2	1.9	1.0				
4	4.0	2.0	1.0			
8	7.9	3.7	2.0	1.0	1.0	
16	15.3	7.3	4.0	2.0	2.0	1.0
32	29.0	13.8	7.6	3.6	3.9	2.4
64	50.5	24.1	14.2	5.9	6.9	4.2

Table I lists the wall clock time in seconds for a ZAPT2 gradient calculation on 1, 2, 4, 8, 16, 32, and 64 processors for the molecules benchmarked in this study. Tables II and III list the associated speedups and parallel efficiencies, respectively. The speedups may be visualized in Fig. 6. Some superlinear speedup is noted for Ti₂Cl₂Cp₄ with the TZV basis set and Fe-porphyrin:imidazole with the MIDI(*d*) basis set with 16 processors and for the Fe-porphyrin:imidazole with the TZV(*d,p*) basis set with 32 and 64 processors. This is most likely due to an unusually long time for the run with 16 processors, and an investigation is in progress. The parallel efficiency for the ZAPT2 gradients is at least 85% on 32 processors and at least 73% on 64 processors for the systems examined in this analysis. As the size of the system increases, the parallel efficiency tends to increase as the time required for computation grows faster than the time required for communication.

ZAPT2 gradients require much less memory and computational time than similar UMP2 gradient calculations. A UMP2 gradient calculation on the Au₃H₄ system requires 18 MWords of replicated memory and 647 MWords of distributed memory, so it requires two processors on the IBM cluster. This UMP2 calculation takes 79 828 s, which is approximately a factor of 3 times the time required for a ZAPT2 gradient calculation (28 292 s). For Au₅H₄, a UMP2 gradient calculation requires 59.2 MWords of replicated memory and 3911 MWords of distributed memory. This

TABLE III. Parallel efficiencies for ZAPT2 gradient step on IBM cluster.

n	Au ₃ H ₄	Au ₃ O ₄	Au ₅ H ₄	Ti ₂ Cl ₂ Cp ₄	Fe-porphyrin	Fe-porphyrin
	380	472	570	486	493	728
P						
1	100.0					
2	97.1	100.0				
4	98.8	97.6	100.0			
8	98.5	93.7	99.3	100.0	100.0	
16	95.5	91.1	98.9	101.8	102.1	100.0
32	90.6	86.4	95.2	90.6	97.5	120.6
64	79.0	75.5	88.5	73.7	86.2	105.7

would require the use of 11 processors on the IBM system. For a calculation with 16 processors on this system, the UMP2 gradient step takes 98 418 s in contrast to a ZAPT2 gradient step that takes 26 690 s.

V. CONCLUSION

The ZAPT2 gradient equations have been revised and subsequently implemented in GAMESS. Features of the scalable implementation using DDI have been discussed in detail. Data of order n^2 or less are replicated across the nodes, while molecular-orbital integrals with two virtual indices or fewer are distributed across the nodes. Gradient terms involving molecular orbital integrals with three virtual indices have been adapted to utilize quarter-transformed integrals in a direct fashion. The algorithm has been designed to use locally held data as much as possible in order to reduce communication costs.

Benchmark calculations have been run on an IBM cluster for five molecular systems. The parallel efficiency for 32 processors is over 85%, and the parallel efficiency for 64 processors is over 73%. As the number of basis functions increases, the parallel efficiencies tend to increase. ZAPT2 gradient calculations require approximately one-fourth of the memory and one-third of the computational time necessary for a UMP2 gradient calculation.

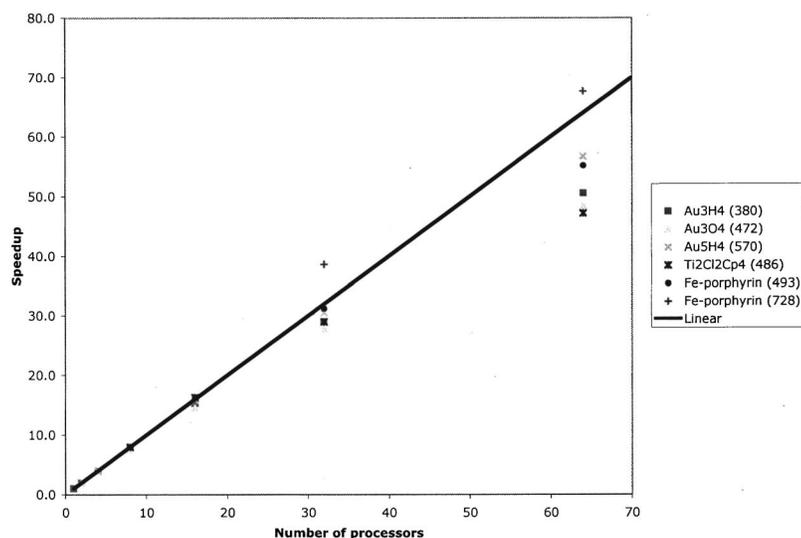


FIG. 6. Speedup curves for benchmark systems. The size of the basis set is listed in parentheses.

ACKNOWLEDGMENTS

This work was supported by a Department of Energy SciDAC grant to the Ames Laboratory. The authors would like to thank Professor Daniel Crawford and Dr. Tim Dudley for helpful discussions. In addition, they would like to thank Ryan Olson for writing the five-point numerical gradient code used in testing this work.

- ¹C. Möller and M. S. Plesset, *Phys. Rev.* **46**, 618 (1934).
- ²H. B. Schlegel, *J. Chem. Phys.* **84**, 4530 (1986).
- ³I. Hubac and P. Carsky, *Phys. Rev. A* **22**, 2392 (1980).
- ⁴C. Murray and E. R. Davidson, *Chem. Phys. Lett.* **187**, 451 (1991).
- ⁵R. D. Amos, J. S. Andrews, N. C. Handy, and P. J. Knowles, *Chem. Phys. Lett.* **185**, 256 (1991).
- ⁶P. J. Knowles, J. S. Andrews, R. D. Amos, N. C. Handy, and J. A. Pople, *Chem. Phys. Lett.* **186**, 130 (1991).
- ⁷W. J. Lauderdale, J. F. Stanton, J. Gauss, J. D. Watts, and R. J. Bartlett, *Chem. Phys. Lett.* **187**, 21 (1991).
- ⁸T. J. Lee and D. Jayatilaka, *Chem. Phys. Lett.* **201**, 1 (1993).
- ⁹P. M. Kozłowski and E. R. Davidson, *Chem. Phys. Lett.* **226**, 440 (1994).
- ¹⁰C. Murray and N. C. Handy, *J. Chem. Phys.* **97**, 6509 (1992).
- ¹¹T. J. Lee, A. P. Rendell, K. G. Dyall, and D. Jayatilaka, *J. Chem. Phys.* **100**, 7400 (1994).
- ¹²P.-Å. Malmqvist, *Chem. Phys. Lett.* **241**, 429 (1995).
- ¹³E. R. Davidson, *Chem. Phys. Lett.* **241**, 432 (1995).
- ¹⁴D. Jayatilaka and T. J. Lee, *Chem. Phys. Lett.* **199**, 211 (1992).
- ¹⁵M. F. Guest and V. Saunders, *Mol. Mater.* **28**, 819 (1974).
- ¹⁶G. D. Fletcher, M. S. Gordon, and R. L. Bell, *Theor. Chem. Acc.* **107**, 57 (2002).
- ¹⁷J. A. Pople, R. Krishnan, H. B. Schlegel, and J. S. Binkley, *Int. J. Quantum Chem., Quantum Chem. Symp.* **13**, 225 (1979).
- ¹⁸P. Pulay and S. Saebø, *Theor. Chim. Acta* **69**, 357 (1986).
- ¹⁹N. C. Handy and H. F. Schaefer III, *J. Chem. Phys.* **81**, 5031 (1984).
- ²⁰N. C. Handy, R. D. Amos, J. F. Gaw, J. E. Rice, and E. D. Simandiras, *Chem. Phys. Lett.* **120**, 151 (1985).
- ²¹T. J. Lee, S. C. Racine, J. E. Rice, and A. P. Rendell, *Mol. Phys.* **85**, 561 (1995).
- ²²M. J. Frisch, M. Head-Gordon, and J. A. Pople, *Chem. Phys. Lett.* **166**, 275 (1990); **166**, 281 (1990); M. Head-Gordon, *Mol. Phys.* **96**, 673 (1999).
- ²³G. D. Fletcher, M. W. Schmidt, and M. S. Gordon, in *Advances in Chemical Physics*, edited by I. Prigogine and S. A. Rice (Wiley, New York, 1999), Vol. 110, p. 267.
- ²⁴G. D. Fletcher, M. W. Schmidt, B. M. Bode, and M. S. Gordon, *Comput. Phys. Commun.* **128**, 190 (2000).
- ²⁵J. Nieplocha, R. J. Harrison, and R. J. Littlefield, in *Proceedings of Supercomputing* (IEEE Computer Society, Washington, DC, 1994), p. 340.
- ²⁶Y. Alexeev, R. A. Kendall, and M. S. Gordon, *Comput. Phys. Commun.* **143**, 69 (2002).
- ²⁷C. M. Aikens and M. S. Gordon, *J. Phys. Chem. A* **108**, 3103 (2004).
- ²⁸G. D. Fletcher, *Mol. Phys.* (revised).
- ²⁹T. J. Dudley, R. M. Olson, M. W. Schmidt, and M. S. Gordon, *J. Comput. Chem.* (in press).
- ³⁰H. Umeda, S. Koseki, U. Nagashima, and M. W. Schmidt, *J. Comput. Chem.* **22**, 1243 (2001).
- ³¹G. Krishnamoorthy, S. P. Webb, T. Nguyen, P. K. Chowdhury, M. Halder, N. J. Wills, S. Carpenter, G. A. Kraus, M. S. Gordon, and J. W. Petrich, *Photochem. Photobiol.* **81**, 924 (2005).
- ³²Z. Gan, Y. Alexeev, R. A. Kendall, and M. S. Gordon, *J. Chem. Phys.* **119**, 47 (2003).
- ³³M. W. Schmidt, K. K. Baldrige, J. A. Boatz *et al.*, *J. Comput. Chem.* **14**, 1347 (1993).
- ³⁴R. M. Olson, M. W. Schmidt, M. S. Gordon, and A. P. Rendell, *Proceedings of Supercomputing*, Phoenix, AZ, 2003.
- ³⁵C. M. Aikens, S. P. Webb, R. L. Bell, G. D. Fletcher, M. W. Schmidt, and M. S. Gordon, *Theor. Chem. Acc.* **110**, 233 (2003).
- ³⁶Y. Yamaguchi, Y. Osamura, J. D. Goddard, and H. F. Schaefer III, *A New Dimension to Quantum Chemistry. Analytic Derivative Methods in Ab Initio Molecular Electronic Structure Theory* (Oxford University Press, New York, 1994).
- ³⁷See EPAPS Document No. E-JCPSA6-123-306547 for detailed description of density matrix creation. This document can be reached via a direct link in the online article's HTML reference section or via the EPAPS homepage (<http://www.aip.org/pubservs/epaps.html>).
- ³⁸I. M. B. Nielsen, *Chem. Phys. Lett.* **255**, 210 (1996); G. D. Fletcher, A. P. Rendell, and P. Sherwood, *Mol. Phys.* **91**, 431 (1997).
- ³⁹T. H. Dunning Jr., *J. Chem. Phys.* **90**, 1007 (1989).
- ⁴⁰W. J. Stevens, H. Basch, and M. Krauss, *J. Chem. Phys.* **81**, 6026 (1984); W. J. Stevens, H. Basch, M. Krauss, and P. Jaisan, *Can. J. Chem.* **70**, 612 (1992); T. R. Cundari and W. J. Stevens, *J. Chem. Phys.* **98**, 5555 (1993).
- ⁴¹R. A. Kendall, T. H. Dunning Jr., and R. J. Harrison, *J. Chem. Phys.* **96**, 6769 (1992).
- ⁴²S. Huzinaga, J. Andzelm, M. Klobukowski, E. Radzio-Andzelm, Y. Sakai, and H. Tatewaki, *Gaussian Basis Sets for Molecular Calculations* (Elsevier, Amsterdam, 1984).