General implementation of the resolution-of-the-identity and Cholesky representations of electron repulsion integrals within coupled-cluster and equation-of-motion methods: Theory and benchmarks

Evgeny Epifanovsky, Dmitry Zuev, Xintian Feng, Kirill Khistyaev, Yihan Shao et al.

Citation: J. Chem. Phys. **139**, 134105 (2013); doi: 10.1063/1.4820484
View online: http://dx.doi.org/10.1063/1.4820484
View Table of Contents: http://jcp.aip.org/resource/1/JCPSA6/v139/i13
Published by the AIP Publishing LLC.

General implementation of the resolution-of-the-identity and Cholesky representations of electron repulsion integrals within coupled-cluster and equation-of-motion methods: Theory and benchmarks

Evgeny Epifanovsky,[1,2,3] Dmitry Zuev,[1] Xintian Feng,[1] Kirill Khistyaev,[1] Yihan Shao,[3] and Anna I. Krylov[1]
[1]*Department of Chemistry, University of Southern California, Los Angeles, California 90089-0482, USA*
[2]*Department of Chemistry, University of California, Berkeley, California 94720, USA*
[3]*6601 Owens Drive, Suite 105, Pleasanton, California 94588, USA*

(Received 20 June 2013; accepted 22 August 2013; published online 2 October 2013)

We present a general implementation of the resolution-of-the-identity (RI) and Cholesky decomposition (CD) representations of electron repulsion integrals within the coupled-cluster with single and double substitutions (CCSD) and equation-of-motion (EOM) family of methods. The CCSD and EOM-CCSD equations are rewritten to eliminate the storage of the largest four-index intermediates leading to a significant reduction in disk storage requirements, reduced I/O penalties, and, as a result, improved parallel performance. In CCSD, the number of rate-determining contractions is also reduced; however, in EOM the number of operations is increased because the transformed integrals, which are computed once in the canonical implementation, need to be reassembled at each Davidson iteration. Nevertheless, for large jobs the effect of the increased number of rate-determining contractions is surpassed by the significantly reduced memory and disk usage leading to a considerable speed-up. Overall, for medium-size examples, RI/CD CCSD calculations are approximately 40% faster compared with the canonical implementation, whereas timings of EOM calculations are reduced by a factor of two. More significant speed-ups are obtained in larger bases, i.e., more than a two-fold speed-up for CCSD and almost five-fold speed-up for EOM-EE-CCSD in cc-pVTZ. Even more considerable speedups (6-7-fold) are achieved by combining RI/CD with the frozen natural orbitals approach. The numeric accuracy of RI/CD approaches is benchmarked with an emphasis on energy differences. Errors in EOM excitation, ionization, and electron-attachment energies are less than 0.001 eV with typical RI bases and with a 10^{-4} threshold in CD. Errors with 10^{-2} and 10^{-3} thresholds, which afford more significant computational savings, are less than 0.04 and 0.008 eV, respectively. © *2013 AIP Publishing LLC*. [http://dx.doi.org/10.1063/1.4820484]

I. INTRODUCTION

Theoretical model chemistries[1] based on wave function methods provide the most reliable approach to electron correlation. Among different *ab initio*-based techniques,[2] coupled-cluster (CC) theory holds a pre-eminent position.[3] The single-reference CC hierarchy of approximations allows one to compute highly accurate molecular structures, reaction energies, and other properties for ground-state species.[2] The equation-of-motion (EOM), or linear response, approach[4–6] extends the CC formalism to a variety of multi-configurational wave functions encountered in electronically excited states and various open-shell species. Unfortunately, similarly to other wave function based methods, the computational cost and hardware requirements (disk and memory) of CC and EOM-CC scale quite steeply with the number of electrons and the size of the one-electron basis set, i.e., the number of occupied (O) and unoccupied, or virtual (V), orbitals. For example, the scaling of a CCSD (coupled-cluster with single and double substitutions) calculation is O^2V^4, and for CCSDT (CCSD plus explicit triple excitations) it is O^3V^5. The disk usage in CC and EOM-CC calculations depends on the implementation specifics and can reach $\mathcal{O}(V^4)$; integral-direct algorithms could be employed to reduce storage requirements.

The high cost of electronic structure calculations originates in the two-electron part of the molecular Hamiltonian that describes electron-electron repulsion. The representation of the electron-repulsion integrals (ERIs) in an atomic orbital (AO) basis gives rise to a four-index tensor:

$$(\mu\nu|\lambda\sigma) = \int \chi_\mu(\vec{r}_1)\chi_\nu(\vec{r}_1)\frac{1}{|\vec{r}_1-\vec{r}_2|}\chi_\lambda(\vec{r}_2)\chi_\sigma(\vec{r}_2)d\vec{r}_1 d\vec{r}_2.$$

The size of this object scales as N^4 where N is the number of basis functions $\chi_i(\vec{r})$. For accurate results the size of the AO basis needs to be sufficiently large, for example a popular cc-pVTZ basis defines 30 contracted Gaussian functions per second-row atom.

All electronic structure methods include contractions of ERIs with various tensors, such as reduced density matrices, wave functions amplitudes, etc. Thus, the large size of ERIs propagates through the electron structure calculations from self-consistent field up to correlated methods.

Fortunately, the structure of the ERI matrix is sparse, which can be exploited in efficient computer

implementations. It was recognized a long time ago that representing the "densities" by a linear expansion over the products of particular one-electron functions, such as $\chi_\mu(\vec{r}_1)\chi_\nu(\vec{r}_1)$, includes linear dependencies and could be rewritten in a more compact form using a new set of basis functions.

There are two alternative approaches to achieve this goal, the density fitting, or resolution-of-the-identity (RI),[7–13] approximation and the Cholesky decomposition (CD).[14–20] In both approaches, the decomposed ERI matrix is represented as

$$(\mu\nu|\lambda\sigma) \approx \sum_{P=1}^{M} B_{\mu\nu}^{P} B_{\lambda\sigma}^{P}, \tag{1}$$

where M is the rank of the decomposition, which depends on the target accuracy. The algorithm for determining B is different in RI and CD approaches: RI uses a predetermined auxiliary basis set that corresponds to the primary one-electron basis, whereas Cholesky vectors are obtained by performing the Cholesky decomposition of the actual ERI matrix. CD is thus a more general approach that can work with any primary basis and is free from externally optimized auxiliary basis sets. The Cholesky approach can be viewed as system-specific density fitting.[17–19]

Decomposition shown in Eq. (1) produces a more compact representation of ERIs compared with the full ERI matrix, thus enabling memory and disk savings. In addition, it allows one to achieve improved parallel performance of calculations involving ERI through reduced disk input-output (I/O) penalties and better CPU utilization. For example, the AO-MO integral transformation has a computational cost of $\mathcal{O}(N^5)$ when using the canonical procedure, now only involves the transformation of the RI/Cholesky vectors and therefore requires only $\mathcal{O}(N^3M)$ steps. The transformed B-matrices can be used to assemble $\langle pq||rs\rangle$ integrals as needed in integral-direct implementations. However, to realize the maximum potential of the method, programmable equations that involve contractions of ERIs with the amplitudes and density matrices need to be rewritten.

The RI/Cholesky representation by itself does not lead to a scaling reduction in CCSD and EOM equations unless special care is taken about exchange-like terms. A number of strategies have been pursued to this end,[19,21,22] including using Cholesky decomposed wave function amplitudes[23,24] and local correlation schemes (see, for example, Refs. 25, 26 and references therein). However, even without these more advanced algorithms, computational savings due to a straightforward implementation of RI/Cholesky representations are very useful, especially in view of improved parallel scaling.

We present our implementation of RI/Cholesky within the CCSD and EOM-CCSD suite of methods in the Q-Chem electronic structure package.[27,28] The implementation eliminates the storage of the most expensive four-index integrals and intermediates. As described below, in the EOM-CCSD implementation we choose to keep two smallest four-index intermediates, OOOO and OOOV.

While CCSD implementations have been reported before,[29,30] the EOM-CCSD methods have not been reimplemented using RI/CD. Below we briefly describe the algorithms used to produce Cholesky and RI vectors (Secs. II A and II B) and give programmable CCSD and EOM-CCSD equations (Sec. III). The following EOM methods have been implemented: EE/SF, IP, and EA. We discuss the performance of the new implementation in Sec. IV.

II. ALGORITHMS

A. Cholesky algorithm

The idea of CD of ERI[14,15,17,31] was proposed over 30 years ago by Beebe and Linderberg.[14] The ERI matrix in the AO basis, which is a positive-semidefinite[14] matrix, can be represented in the Cholesky-decomposed form as given by Eq. (1). The rank of the decomposition, M, is typically 3–10 times the number of basis functions N.[17] It depends on the decomposition threshold δ and is considerably smaller than the full rank of the matrix, $N(N+1)/2$.[14,17,32] CD removes linear dependencies in product densities[17] $(\mu\nu|$, allowing one to approximate the original matrix to arbitrary accuracy.

Decomposition threshold δ defined by the user is the only parameter that controls the accuracy and the rank of the decomposition. The algorithm[15,17,31] proceeds as follows:

(1) Compute all diagonal elements of ERI: $D^0_{\lambda\sigma,\lambda\sigma} = (\lambda\sigma|\lambda\sigma)$.
(2) Choose the largest diagonal element $(\lambda\sigma_0|\lambda\sigma_0)$. $(\lambda\sigma)_0$ here is a fixed index corresponding to the largest diagonal element.
(3) Compute densities $(\mu\nu|\lambda\sigma_0)$.
(4) Compute first Cholesky vector $B^1_{\mu\nu} = (\mu\nu|\lambda\sigma_0)/\sqrt{(\lambda\sigma_0|\lambda\sigma_0)}$.
From this point the algorithm proceeds in an iterative manner, checking the accuracy and generating a new Cholesky vector to refine the previous-step approximation at every iteration. k is an iteration count that starts from 2 and increments after every iteration.
(5) Update the residual of the diagonal by subtracting the Cholesky vector obtained in the previous iteration $D^{(k-1)}_{\lambda\sigma,\lambda\sigma} = D^{(k-2)}_{\lambda\sigma,\lambda\sigma} - B^{(k-1)}_{\lambda\sigma} B^{(k-1)}_{\lambda\sigma}$.
(6) Choose the largest element of the diagonal residual $D^{k-1}_{\lambda\sigma_{k-1},\lambda\sigma_{k-1}}$. If $D^{k-1}_{\lambda\sigma_{k-1},\lambda\sigma_{k-1}} < \delta$, then terminate and return the Cholesky vectors, $\{B^i_{\mu\nu}\}^{k-1}_{i=1}$.
(7) Compute densities $(\mu\nu|\lambda\sigma_{k-1})$ and the corresponding residual, $D^{(k-1)}_{\mu\nu,\lambda\sigma_{k-1}} = (\mu\nu|\lambda\sigma_{k-1}) - \sum_{i=1}^{k-1} B^i_{\mu\nu} B^i_{\lambda\sigma_{k-1}}$.
(8) Compute new Cholesky vector $B^k_{\mu\nu} = D^{k-1}_{\mu\nu,\lambda\sigma_{k-1}}/\sqrt{D^{k-1}_{\lambda\sigma_{k-1},\lambda\sigma_{k-1}}}$. Repeat from step (5).

Since the ERI matrix is positive-semidefinite,[8,14,33] it follows that

$$|D^{k-1}_{\mu\nu,\lambda\sigma}| \le \sqrt{D^{k-1}_{\mu\nu,\mu\nu} D^{k-1}_{\lambda\sigma,\lambda\sigma}}.$$

Thus, the accuracy of the decomposition is given by the largest element of diagonal residual matrix $D_{\lambda\sigma,\lambda\sigma}$ at every iteration (step 6), which ensures that the error in any ERI matrix element does not exceed δ.

Note that the algorithm does not require the calculation and storage of the full ERI matrix $[\mathcal{O}(N^4)]$, which would be

prohibitive for large systems. At the initialization of the algorithm only the calculation of the diagonal elements is necessary [step 1, $\mathcal{O}(N^2)$], which are updated at each iteration by subtracting newly produced Cholesky vectors to form a residual diagonal matrix (step 5). The calculation of the densities $(\mu\nu|\lambda\sigma_{k-1})$ [step 7, $\mathcal{O}(N^2)$] are performed at each step with subsequent calculation of the residual and the corresponding Cholesky vector (step 8). At each iteration only the calculation of new $\mathcal{O}(N^2)$ elements of the ERI matrix is required and the number of Cholesky vectors grows by one resulting in the $\mathcal{O}(MN^2)$ memory storage of all Cholesky vectors for the final decomposition. Thus, only a small fraction of about 1–5% of the full ERI matrix needs to be stored in memory in the decomposition procedure.[17]

The most expensive step is the calculation of the residual matrix[17] (step 7), which requires $(M-1)$ subtractions of previously obtained Cholesky vectors at each iteration [$\mathcal{O}((M-1)N^2)$ operations at each iteration], giving rise to the full complexity of the algorithm of $\mathcal{O}(M^2N^2)$. For correlated calculations, the Cholesky vectors obtained in the AO basis are usually transformed to the molecular orbital (MO) basis.

This algorithm is implemented using our tensor algebra library[34] such that Cholesky vectors are stored as a list of two-dimensional block tensors, i.e., a list of block matrices. The library is based on virtual memory management such that block tensors are stored in RAM if sufficient memory is available or saved on disk and reloaded as necessary. Note that the generation of a new Cholesky vector [steps 5–8] does not require vectors from previous iterations ($k-1$ at step k) to be in RAM; for calculation of the residual matrix (step 5) they can be uploaded from disk sequentially, or even block-by-block. After all Cholesky vectors $B_{\mu\nu}$ are generated, the list of block matrices is merged to form a three-dimensional block tensor $B^M_{\mu\nu}$ containing all the Cholesky vectors.

B. Resolution-of-the-identity algorithm

Similar to the Cholesky decomposition, the RI approach[8–13] allows one to expand product densities $(\mu\nu|$ in an auxiliary basis set:

$$(\mu\nu|\lambda\sigma) \approx \sum_{PQ} C^P_{\mu\nu}(P|Q)C^Q_{\lambda\sigma} \equiv \sum_{PQ}(\mu\nu|P)(P|Q)^{-1}(Q|\lambda\sigma).$$

Indices P and Q denote auxiliary basis functions and $(P|Q)$ defines a Coulomb metric matrix:[13,17,20]

$$(P|Q) = \int \frac{P(\vec{r}_1)Q(\vec{r}_2)}{|\vec{r}_1-\vec{r}_2|}d\vec{r}_1 d\vec{r}_2.$$

The auxiliary basis expansion coefficients $(C^L_{\mu\nu})$ are found by minimizing the difference between the actual and fitted product densities,[13,17,18] leading to the following set of linear equations:

$$\sum_L (K|L)C^L_{\mu\nu} = (K|\mu\nu).$$

By defining new auxiliary basis coefficients

$$B^K_{\mu\nu} \equiv \sum_L C^L_{\mu\nu}(L|K)^{1/2} = \sum_L (K|\mu\nu)(K|L)^{-1/2},$$

we can rewrite approximate ERIs in a form identical to the Cholesky representation[13] as given by Eq. (1).

The accuracy and performance of RI depend on the quality of the chosen auxiliary basis set. Ideally an auxiliary basis set should be balanced between accuracy and compactness. Errors should be at least an order of magnitude smaller than the error due to one-electron basis set incompleteness. Rank M should be no more that 2–4 times larger than the number of AO basis functions N.[7,17,35–40] To achieve these goals, auxiliary basis sets are usually optimized for each atom, AO basis set, and level of theory (e.g., Hartree–Fock, MP2).[7,17,35–40] In this work, we employ auxiliary basis sets developed for MP2.

III. RI/CD CCSD AND EOM-CCSD METHODS: THEORY

A. Coupled-cluster equations with single and double substitutions

The exact solution of the Schrödinger equation can be written as the exponential of a cluster operator $\hat{T}$ operating on a reference function:[41]

$$\Psi_{exact} = \Psi_{CC} = e^{\hat{T}}\Phi_0,$$

where Φ_0 is a single Slater determinant. In CCSD, the expansion of $\hat{T}$ is truncated at a two-electron excitation level:

$$\hat{T} \approx \hat{T}_1 + \hat{T}_2.$$

For $\hat{T}_1$ and $\hat{T}_2$, the expansions are:[42]

$$\hat{T}_1 = \sum_{ia} t^a_i a^\dagger i \quad \hat{T}_2 = \frac{1}{4}\sum_{ijab} t^{ab}_{ij} a^\dagger i b^\dagger j.$$

Thus,

$$\Psi_{CCSD} = e^{\hat{T}_1+\hat{T}_2}\Phi_0.$$

The equations to determine CCSD correlation energy E_{CCSD} and cluster amplitudes t^a_i, t^{ab}_{ij} are derived algebraically by a projection approach such that the Schrödinger equation is satisfied in the subspace spanned by the reference, singly, and doubly excited determinants:

$$E_{CCSD} = \langle\Phi_0|\hat{H}|\Psi_{CCSD}\rangle = \langle\Phi_0|\hat{H}|\left(1+\hat{T}_1+\frac{1}{2}\hat{T}_1^2+\hat{T}_2\right)\Phi_0\rangle, \tag{2}$$

$$0 = \langle\Phi^a_i|\hat{H}-E_{CCSD}|\Psi_{CCSD}\rangle = \langle\Phi^a_i|\hat{H}-E_{CCSD}|\left(1+\hat{T}_1+\frac{1}{2}\hat{T}_1^2+\hat{T}_2+\hat{T}_1\hat{T}_2+\frac{1}{3!}\hat{T}_1^3\right)\Phi_0\rangle, \tag{3}$$

$$0 = \langle\Phi^{ab}_{ij}|\hat{H}-E_{CCSD}|\Psi_{CCSD}\rangle = \left\langle\Phi^{ab}_{ij}|\hat{H}-E_{CCSD}|\left(1+\hat{T}_1+\frac{1}{2}\hat{T}_1^2+\hat{T}_2+\hat{T}_1\hat{T}_2+\frac{1}{3!}\hat{T}_1^3+\frac{1}{2}\hat{T}_2^2+\frac{1}{2}\hat{T}_1^2\hat{T}_2+\frac{1}{4!}\hat{T}_1^4\right)\Phi_0\right\rangle. \tag{4}$$

Evaluating Eq. (2) in terms of amplitudes t_i^a and t_{ij}^{ab} yields the following expression:

$$E_{CCSD} = \langle\Phi_0|\hat{H}|\Phi_0\rangle + \sum_{ia} f_{ia}t_i^a + \frac{1}{2}\sum_{ijab}\langle ij||ab\rangle t_i^a t_j^b + \frac{1}{4}\sum_{ijab}\langle ij||ab\rangle t_{ij}^{ab}, \quad (5)$$

where:

$$f_{ia} = \langle\Phi_i^a|\hat{H}|\Phi_0\rangle = h_{ia} + \sum_j \langle ij||aj\rangle$$

$$\langle ij||ab\rangle = \langle ij|ab\rangle - \langle ij|ba\rangle = (ia|jb) - (ib|ja)$$

$$(ia|jb) = \int \phi_i(1)\phi_j(2)\frac{1}{r_{12}}\phi_a(1)\phi_b(2)d1d2.$$

Once Eq. (5) is substituted into Eqs. (3) and (4), t_i^a and t_{ij}^{ab} amplitudes can be solved iteratively by

$$t_i^a\Delta_i^a = f_{ia} - \sum_l F_{li}^{(3)}t_l^a + \sum_d F_{ad}^{(1)}t_i^d + \sum_{kc} F_{kc}^{(2)}t_{ik}^{ac} - \sum_{kc}\langle ic||ka\rangle t_k^c - \frac{1}{2}\sum_{klc}\langle kl||ic\rangle t_{kl}^{ac} + \frac{1}{2}\sum_{kcd}\langle ka||cd\rangle t_{ki}^{cd}$$

and

$$t_{ij}^{ab}\Delta_{ij}^{ab} = \langle ij||ab\rangle + \mathcal{P}_{ab}^{-}\left\{\sum_c t_{ij}^{ac}F_{bc}^{(2)} - \sum_k I_{ijkb}^{(2a)}t_k^a + \mathcal{P}_{ij}^{-}\sum_{kc} I_{kbic}^{(1a)}t_{jk}^{ac}\right\} + \mathcal{P}_{ij}^{-}\left\{\sum_c\langle jc||ba\rangle t_i^c - \sum_k t_{ik}^{ab}F_{jk}^{(2)}\right\} + \frac{1}{2}\sum_{cd}\langle ab||cd\rangle\tilde{t}_{ij}^{cd} + \frac{1}{2}\sum_{kl} t_{kl}^{ab}I_{ijkl}^{(4)},$$

where $\Delta_i^a = f_{ii} - f_{aa}$ and $\Delta_{ij}^{ab} = \Delta_i^a + \Delta_j^b$. The expressions for the intermediates are given in Table I.

Memory requirements for the T amplitude update procedure in the closed-shell case are[43]

$$\frac{9}{8}O^4 + 3O^3V + 6O^2V^2 + \frac{3}{2}OV^3 + \frac{3}{8}V^4. \quad (6)$$

This estimate includes all the blocks of ERIs, necessary four-index intermediates, and two sets of T_2 amplitudes. Excluded are additional copies of T amplitudes required by the DIIS[44] procedure and lower-dimensional quantities. The $\mathcal{O}(N^6)$ part of the total computational cost of updating T amplitudes is $\frac{7}{4}O^4V^2 + \frac{29}{4}O^3V^3 + \frac{5}{8}O^2V^4$.

B. Equations for CD/RI CCSD in the spin-orbital basis

Because RI and Cholesky representations of ERI use identical expressions, we begin with the following expression for anti-symmetrized integrals:

$$\langle\mu\lambda||\nu\sigma\rangle \approx \sum_P B_{\mu\nu}^P B_{\lambda\sigma}^P - \sum_P B_{\mu\sigma}^P B_{\lambda\nu}^P = \mathcal{P}_{\nu\sigma}^{-}\sum_P B_{\mu\nu}^P B_{\lambda\sigma}^P. \quad (7)$$

Upon substituting Eq. (7) into Eq. (5), the RI/CD CCSD energy can be computed as follows:

$$E_{CCSD} = \langle\Phi_0|\hat{H}|\Phi_0\rangle + \sum_{ia}\left(f_{ia} + \frac{1}{2}\sum_P B_{ia}^P M_P^{2T} - \frac{1}{2}\sum_{jP} B_{ja}^P\left(M_{ij}^P - B_{ij}^P\right)\right)t_i^a + \frac{1}{2}\sum_{ijab}\left(\sum_P B_{ia}^P B_{jb}^P\right)t_{ij}^{ab}.$$

TABLE I. Intermediates for CCSD calculations[45] and estimates to store and compute them (closed-shell case).

Equation	Memory	Flops
$F_{bc}^{(1)} = f_{bc} + \sum_{kd}\langle kb\|\|dc\rangle t_k^d - \frac{1}{2}\sum_{kld}\langle kl\|\|cd\rangle t_{kl}^{bd}$		
$F_{ij}^{(2)} = f_{ij} + \sum_a f_{ja}t_i^a + \sum_{ka}\langle jk\|\|ia\rangle t_k^a + \sum_{kab}\langle jk\|\|ab\rangle t_i^a t_k^b + \frac{1}{2}\sum_{kab}\langle jk\|\|ab\rangle t_{ik}^{ab}$		
$F_{ia}^{(2)} = f_{ia} + \sum_{jb}\langle ij\|\|ab\rangle t_j^b$		
$F_{bc}^{(2)} = F_{bc}^{(1)} - \sum_k f_{kc}t_k^b - \sum_{kld}\langle kl\|\|cd\rangle t_k^b t_l^d$		
$F_{ki}^{(3)} = f_{ki} + \sum_c F_{kc}^{(2)}t_i^c + \frac{1}{2}\sum_{jab}\langle kj\|\|ab\rangle t_{ij}^{ab} + \sum_{lc}\langle kl\|\|ic\rangle t_l^c$		
$\tilde{t}_{ij}^{ab} = t_{ij}^{ab} + \frac{1}{2}\mathcal{P}_{ij}^{-}\mathcal{P}_{ab}^{-}t_i^a t_j^b$	$\frac{3}{4}O^2V^2$	
$I_{iajb}^{(1a)} = \langle ia\|\|jb\rangle - \sum_c\langle ia\|\|bc\rangle t_j^c - \sum_k\langle ik\|\|jb\rangle t_k^a - \frac{1}{2}\sum_{kc}\langle ik\|\|cb\rangle\left(t_{jk}^{ca} + 2t_j^c t_k^a\right)$	$2O^2V^2$	$3O^3V^3$
$I_{ijkb}^{(2a)} = \langle ij\|\|kb\rangle - \frac{1}{2}\sum_l I_{ijkl}^{(4)}t_l^b + \frac{1}{2}\sum_{cd}\langle kb\|\|cd\rangle\tilde{t}_{ij}^{cd} + \mathcal{P}_{ij}^{-}\sum_c\langle kb\|\|ic\rangle t_j^c$	$\frac{3}{2}O^3V$	$\frac{5}{4}O^3V^3$
$I_{ijkl}^{(4)} = \langle ij\|\|kl\rangle + \frac{1}{2}\sum_{ab}\langle kl\|\|ab\rangle\tilde{t}_{ij}^{ab} + \mathcal{P}_{ij}^{-}\sum_a\langle kl\|\|ia\rangle t_j^a$	$\frac{3}{4}O^4$	$\frac{5}{8}O^4V^2$
$\sum_{kc} I_{kbic}^{(1a)}t_{jk}^{ac}$		$3O^3V^3$
$\sum_{cd}\langle ab\|\|cd\rangle\tilde{t}_{ij}^{cd}$		$\frac{5}{8}O^2V^4$
$\sum_{kl} t_{kl}^{ab}I_{ijkl}^{(4)}$		$\frac{5}{8}O^4V^2$

The expressions for t_i^a and t_{ij}^{ab} amplitudes become

$$\begin{aligned} t_i^a \Delta_i^a = {} & f_{ia} + \sum_d f_{ad} t_i^d - \sum_{kP} M_{ki}^P \left(M_{kaP}^{2T} - M_{kaP}^{2TT} \right) \\ & + \sum_P M_P^{2T} M_{ia}^P \\ & - \sum_l \left(\sum_c f_{lc} t_i^c + f_{li} \right) t_l^a - \sum_{kP} M_{kaP}^{3T} \left(M_{ki}^P - B_{ki}^P \right) \\ & + \sum_{cP} M_{ca}^P M_{icP}^{2T} \\ & + \sum_{kc} t_{ik}^{ac} \left[f_{kc} - \sum_{jP} \left(M_{kj}^P - B_{kj}^P \right) B_{jc}^P \right] - \sum_{cP} M_{icP}^{1T} B_{ac}^P, \end{aligned} \tag{8}$$

$$\begin{aligned} t_{ij}^{ab} \Delta_{ij}^{ab} = {} & I_{ijab}^{(2i)} + \mathcal{P}_{ab}^- \Bigg\{ \sum_P M_{ia}^P M_{jb}^P + \sum_c t_{ij}^{ac} F_{bc}^{(2)} \\ & + \mathcal{P}_{ij}^- \left[\sum_{kc} \left(\frac{1}{2} I_{kcib}^{(1i)} + \sum_P M_{ki}^P M_{cb}^P \right) t_{jk}^{ac} \right] \Bigg\} \\ & - \mathcal{P}_{ij}^- \sum_k t_{ik}^{ab} F_{jk}^{(2)} \\ & + \frac{1}{2} \sum_{kl} \left(I_{ijkl}^{(3i)} + \mathcal{P}_{ij}^- \sum_P M_{ki}^P M_{lj}^P \right) t_{kl}^{ab}, \end{aligned} \tag{9}$$

$$I_{kcib}^{(1i)} = \sum_{ld} \left(\sum_P B_{kd}^P B_{lc}^P \right) t_{il}^{bd}, \tag{10}$$

$$I_{ijab}^{(2i)} = -\frac{1}{2} \sum_{cd} \left(\mathcal{P}_{ab}^- \sum_P M_{da}^P M_{cb}^P \right) t_{ij}^{cd}, \tag{11}$$

$$I_{ijkl}^{(3i)} = \frac{1}{2} \sum_{cd} \left(\mathcal{P}_{cd}^- \sum_P B_{kc}^P B_{ld}^P \right) t_{ij}^{cd}. \tag{12}$$

$I^{(1i)}$, $I^{(2i)}$, and $I^{(3i)}$ intermediates are computed on the fly using a three-tensor contraction procedure[46] and are immediately added to the result. Expressions for the new M-intermediates and the revised $F^{(2)}$-intermediates are given in Table II.

The notation for M-intermediates is as follows: upper index 1, 2, or 3 denotes the terms containing B_{oo}^P, B_{ov}^P or B_{vv}^P, respectively. M^{2T} is used for the terms that include a contraction between B_{ov}^P and a T-amplitude. M^{2TT} denotes the terms that include the contraction of M^{2T} with a T-amplitude.

All $M^{(xT)}$, M, and $F^{(2)}$ intermediates in Table II are recomputed at each CCSD iteration by using the most current t_i^a and t_{ij}^{ab} amplitudes. The intermediates are discarded after having been used to compute updated T-amplitudes. After CCSD equations have been solved, intermediates $F^{(2)}$, M, $I^{(2b)}$, and $I^{(4b)}$ are computed using the converged T-amplitudes and stored to be reused in property and EOM calculations.

Total storage requirements[47] for the amplitude update procedure in CD/RI-CCSD, including all necessary integrals, intermediates, and two copies of T amplitudes, are as follows:

$$\frac{3}{2} O^2 M + 6 O V M + \frac{3}{2} V^2 M + \frac{3}{4} O^4 + \frac{7}{2} O^2 V^2. \tag{13}$$

To illustrate the difference in storage requirements, consider a calculation of closed-shell naphthalene using the cc-pVTZ/rimp2-cc-pVTZ basis set. There are 68 electrons, 412 basis functions ($O = 34$, $V = 378$), and 1050 auxiliary basis functions ($M = 1050$). Whereas conventional CCSD calculation using Eq. (6) requires 10917 Mwords (85 GB), CD/RI-CCSD using Eq. (13) requires 846 Mwords (6.6 GB). Thus, for this calculation the data set is almost 13 times smaller in the case of RI-CCSD.

The number of floating point operations scales as $\mathcal{O}(N^6)$ for both CCSD and CD/RI-CCSD. The most significant contraction in CCSD, $\sum_{cd} \langle ab||cd \rangle \tilde{t}_{ij}^{cd}$, and its CD/RI-CCSD counterpart, Eq. (11), take the same number of flops. In the latter case, the intermediate $\mathcal{P}_{ab}^- \sum_P M_{da}^P M_{cb}^P$ is formed on the fly thus reducing overall memory requirements. The CD/RI-CCSD equations involve fewer O^3V^3-type contractions, leading to a smaller prefactor ($4O^3V^3$ vs. $\frac{29}{4} O^3V^3$ in conventional CCSD). While this improvement is offset by the increased number and cost of $\mathcal{O}(N^5)$ steps, in practical applications CD/RI-CCSD are superior in terms of floating point

TABLE II. Intermediates used in CD/RI-CCSD calculations, including modified $F^{(2)}$, $I^{(2b)}$, $I^{(4b)}$, and estimates to store and compute them (closed-shell case).

Equation	Memory	Flops
$M_{iaP}^{1T} = \sum_j B_{ij}^P t_j^a$	OVM	
$M_{iaP}^{2T} = \sum_{jb} B_{jb}^P t_{ij}^{ab}$	OVM	
$M_{iaP}^{2TT} = \sum_j \left(M_{ji}^P - B_{ji}^P \right) t_j^a$	OVM	
$M_{iaP}^{3T} = \sum_b B_{ab}^P t_i^b$	OVM	
$M_P^{2T} = \sum_{jb} B_{jb}^P t_j^b$		
$M_{ij}^P = \sum_a B_{ia}^P t_j^a + B_{ij}^P$	O^2M	
$M_{ia}^P = M_{iaP}^{3T} - M_{iaP}^{1T} - M_{iaP}^{2TT} + B_{ia}^P + M_{iaP}^{2T}$	OVM	
$M_{ab}^P = B_{ab}^P - \sum_j B_{ja}^P t_j^b$	V^2M	
$F_{ij}^{(2)} = f_{ij} + \sum_a f_{ja} t_i^a + \sum_P M_{kj}^P M_P^{2T}$ $- \sum_{lP} \left(M_{kl}^P - B_{kl}^P \right) M_{lj}^P + \sum_{cP} B_{kc}^P M_{jcP}^{2T}$		
$F_{ia}^{(2)} = f_{ia} + \sum_P B_{ia}^P M_P^{2T} - \sum_{jP} B_{ja}^P \left(M_{ij}^P - B_{ij}^P \right)$		
$F_{bc}^{(2)} = f_{bc} - \sum_{kP} B_{kc}^P \left(M_{kbP}^{2T} + M_{kbP}^{3T} - M_{kbP}^{2TT} \right)$ $+ \sum_P M_{cb}^P M_P^{2T} - \sum_k f_{kc} t_k^b$		
$I_{ijka}^{(2b)} = -\mathcal{P}_{ij}^- \{ \sum_P M_{kj}^P M_{ia}^P + \sum_{lc} \left(\sum_P B_{kc}^P M_{li}^P \right) t_{jl}^{ac} \}$ $+ \sum_{cd} \left(\sum_P B_{kc}^P M_{da}^P \right) t_{ij}^{cd} - \sum_c t_{ij}^{ac} F_{kc}^{(2)}$	$\frac{3}{2} O^3 V$	$\frac{3}{2} O^4 V^2$
$I_{ijkl}^{(4b)} = \mathcal{P}_{ij}^- \sum_P M_{ki}^P M_{lj}^P + \sum_{ab} \left(\sum_P B_{ka}^P B_{lb}^P \right) t_{ij}^{ab}$	$\frac{3}{4} O^4$	$\frac{5}{8} O^4 V^2$
$I_{kcib}^{(1i)}$ (Eq. (10))		$2O^3V^3$
$I_{ijab}^{(2i)}$ (Eq. (11))		$\frac{5}{8} O^2 V^4$
$I_{ijkl}^{(3i)}$ (Eq. (12))		$\frac{5}{8} O^4 V^2$
$\sum_{kc} \left(\frac{1}{2} I_{kcib}^{(1i)} + \sum_P M_{ki}^P M_{cb}^P \right) t_{jk}^{ac}$	$2O^2V^2$	$2O^3V^3$
$\sum_{kl} \left(I_{ijkl}^{(3i)} + \mathcal{P}_{ij}^- \sum_P M_{ki}^P M_{lj}^P \right) t_{kl}^{ab}$	$\frac{3}{4} O^4$	$\frac{5}{8} O^4 V^2$

operations, memory, and I/O, as illustrated by benchmark calculations in Sec. IV.

C. EOM-EE/SF-CCSD equations

In the EOM-CCSD framework, the target excited-state wave functions are written as follows:[45,48]

$$|\Psi_R\rangle = Re^{\hat{T}}|\Phi_0\rangle, \tag{14}$$

$$\langle\Psi_L| = \langle\Phi_0|e^{-\hat{T}}L^{\dagger}. \tag{15}$$

The operators R and L are linear excitation operators:

$$R = R_0 + R_1 + R_2 + \cdots, \tag{16}$$

$$R_n = \frac{1}{n!^2}\sum r^{abc\cdots}_{ijk\ldots}a^{\dagger}ib^{\dagger}jc^{\dagger}k\cdots. \tag{17}$$

In EOM-EE operators, R_n are spin-conserving ($M_s = 0$ operators), whereas in EOM-SF they involve changing the spin of an electron ($M_s = -1$). The spin-orbital form of the EOM-CCSD equations is the same in EOM-EE and EOM-SF.[45,49]

By introducing a similarity transformed Hamiltonian $\bar{H}$:

$$\bar{H} \equiv e^{-\hat{T}}He^{\hat{T}}, \tag{18}$$

the energy and CCSD amplitude equations become

$$E_{CC} = \langle\Phi_0|\bar{H}|\Phi_0\rangle, \tag{19}$$

$$0 = \langle\Phi_i^a|\bar{H} - E_{CC}|\Phi_0\rangle, \tag{20}$$

$$0 = \langle\Phi_{ij}^{ab}|\bar{H} - E_{CC}|\Phi_0\rangle, \tag{21}$$

$$\cdots$$

where E_{CC} is the coupled-cluster energy for the reference state. Usually both T and R are truncated at the same level, which is the single (S) and double (D) excitations in this work. Thus, in the basis of the reference (O), S, and D, we have

$$\begin{pmatrix} 0 & \bar{H}_{OS} & \bar{H}_{OD} \\ 0 & \bar{H}_{SS} - E_{CC} & \bar{H}_{SD} \\ 0 & \bar{H}_{DS} & \bar{H}_{DD} - E_{CC} \end{pmatrix} \begin{pmatrix} R_0 \\ R_1 \\ R_2 \end{pmatrix} = \omega \begin{pmatrix} R_0 \\ R_1 \\ R_2 \end{pmatrix}, \tag{22}$$

where on the left-hand side E_{CC} only appears for the diagonal elements in the diagonal blocks and $\omega = E - E_{CC}$. Because the right eigenvectors do not form an orthonormal set, $R_0 = r_0\hat{1}$ can be present in the target excited states:

$$\begin{aligned} r_0 &= \frac{1}{\omega}(\bar{H}_{OS}R_1 + \bar{H}_{OD}R_2) \\ &= \frac{1}{\omega}\left\{\sum_{ia}F^{(2)}_{ia}r^a_i + \frac{1}{4}\sum_{ijab}\langle ij||ab\rangle r^{ab}_{ij}\right\}. \end{aligned} \tag{23}$$

Equation (22) is solved by using the generalized Davidson iterative diagonalization procedure,[45] which involves the calculation of the following σ-vectors:

$$\begin{aligned} \sigma^a_i &= ([\bar{H}_{SS} - E_{CC}]R_1)^a_i + (\bar{H}_{SD}R_2)^a_i \\ &= \sum_b F^{(2)}_{ab}r^b_i - \sum_j F^{(2)}_{ij}r^a_j - \sum_{jb} I^{(1)}_{ibja}r^b_j + \sum_{jb} F^{(2)}_{jb}r^{ab}_{ij} \\ &\quad - \frac{1}{2}\sum_{jkb} I^{(6)}_{jkib}r^{ab}_{jk} - \frac{1}{2}\sum_{jbc} I^{(7)}_{jabc}r^{bc}_{ij}, \end{aligned} \tag{24}$$

$$\begin{aligned} \sigma^{ab}_{ij} &= (\bar{H}_{DS}R_1)^{ab}_{ij} - ([\bar{H}_{DD} - E_{CC}]R_2)^{ab}_{ij} \\ &= -\mathcal{P}^-_{ab}\sum_k I^{(2)}_{ijkb}r^a_k - \mathcal{P}^-_{ij}\sum_c I^{(3)}_{jcab}r^c_i + \mathcal{P}^-_{ij}\sum_l T^{(1)}_{il}t^{ab}_{jl} \\ &\quad + \mathcal{P}^-_{ab}\sum_d T^{(2)}_{ad}t^{bd}_{ij} + \mathcal{P}^-_{ij}\sum_k r^{ab}_{jk}F^{(2)}_{ik} \\ &\quad + \mathcal{P}^-_{ab}\sum_c r^{ac}_{ij}F^{(2)}_{bc} + \mathcal{P}^-_{ij}\mathcal{P}^-_{ab}\sum_{kc} I^{(1)}_{ickb}r^{ac}_{jk} \\ &\quad + \frac{1}{2}\sum_{kl} r^{ab}_{kl}I^{(4)}_{ijkl} + \frac{1}{2}\sum_{cd} r^{cd}_{ij}I^{(5)}_{abcd} \\ &\quad + \mathcal{P}^-_{ij}\sum_l T^{(3)}_{il}t^{ab}_{jl} + \mathcal{P}^-_{ab}\sum_d T^{(4)}_{ad}t^{bd}_{ij}. \end{aligned} \tag{25}$$

The I, F, and T intermediates used in Eqs. (24) and (25) are collected in Tables I and III. Total storage requirements for computing a σ-vector, including a set of T, R, σ amplitudes and all the integrals and intermediates, are as follows:

$$\frac{9}{8}O^4 + \frac{9}{2}O^3V + 6O^2V^2 + \frac{9}{2}OV^3 + \frac{9}{8}V^4. \tag{26}$$

Note that multiple sets of R and σ amplitudes are required in the Davidson procedure for finding the excitation energies.

D. CD/RI implementation of EOM-EE/SF-CCSD

Following the same procedure as in the derivation of the CD/RI-CCSD equations, we arrive at

$$r_0 = \frac{1}{\omega}\left[\sum_{ia}F^{(2)}_{ia}r^a_i + \frac{1}{2}\sum_{ijab}\left(\sum_P B^P_{ia}B^P_{jb}\right)r^{ab}_{ij}\right], \tag{27}$$

$$\begin{aligned} \sigma^a_i &= \sum_b r^b_i F^{(2)}_{ab} - \sum_j r^a_j F^{(2)}_{ij} - \sum_{jP} M^P_{ji}\left(M^{2R}_{jaP} + M^{3R}_{jaP} - M^{2RT}_{jaP}\right) \\ &\quad + \sum_P M^{2R}_P M^P_{ia} - \sum_{kc}\left(\sum_{jP} B^P_{jc}M^{2R}_{kjP}\right)t^{ac}_{ik} \\ &\quad + \sum_{jb} r^{ab}_{ij}F^{(2)}_{jb} + \sum_{cP} M^{2R}_{icP}M^P_{ca}, \end{aligned} \tag{28}$$

TABLE III. I and T intermediates for EOM-CCSD[45] and estimated cost to store and compute them (closed-shell case).

Equation	Memory	Flops
$I^{(1)}_{ickb} = \langle ic\|\|kb\rangle - \sum_d \langle kb\|\|cd\rangle t^d_i - \sum_{ld} \langle kl\|\|cd\rangle t^{bd}_{il} + \sum_l \left[\sum_d \langle kl\|\|cd\rangle t^d_i - \langle kl\|\|ic\rangle\right] t^b_l$	$2O^2V^2$	$3O^3V^3$
$I^{(2)}_{ijkb} = \langle ij\|\|kb\rangle - \sum_l I^{(4)}_{ijkl} t^b_l + \frac{1}{2}\sum_{cd}\langle kb\|\|cd\rangle \tilde{t}^{cd}_{ij} + \sum_d \left[\sum_{lc}\langle kl\|\|cd\rangle t^c_l\right] t^{bd}_{ij} - \sum_c t^{bc}_{ij} f_{kc}$		
$\quad -\mathcal{P}^-_{ij}\left\{\sum_c \left[\langle kb\|\|jc\rangle - \sum_{ld}\langle kl\|\|cd\rangle t^{bd}_{jl}\right] t^c_i + \sum_{lc}\langle kl\|\|jc\rangle t^{bc}_{il}\right\}$	$\frac{3}{2}O^3V$	$\frac{21}{4}O^3V^3$
$I^{(3)}_{jcab} = \langle jc\|\|ab\rangle - \sum_d I^{(5)}_{abcd} t^d_j + \frac{1}{2}\sum_{kl}\langle kl\|\|jc\rangle t^{ab}_{kl} - \sum_l \left(\sum_{kd}\langle lk\|\|cd\rangle t^d_k\right) t^{ab}_{jl} - \sum_l t^{ab}_{jl} f_{lc}$		
$\quad +\mathcal{P}^-_{ab}\left\{\sum_k \left(\langle ka\|\|jc\rangle - \frac{1}{2}\sum_l \langle kl\|\|jc\rangle t^a_l - \sum_{ld}\langle kl\|\|cd\rangle t^{ad}_{jl}\right) t^b_k - \sum_{ld}\langle lb\|\|cd\rangle t^{ad}_{jl}\right\}$	$\frac{3}{2}OV^3$	$\frac{33}{4}O^3V^3$
$I^{(4)}_{ijkl} = \langle ij\|\|kl\rangle + \frac{1}{2}\sum_{ab}\langle kl\|\|ab\rangle \tilde{t}^{ab}_{ij} + \mathcal{P}^-_{ij}\sum_a \langle kl\|\|ia\rangle t^a_j$	$\frac{3}{4}O^4$	$\frac{5}{8}O^4V^2$
$I^{(5)}_{abcd} = \langle ab\|\|cd\rangle + \frac{1}{2}\sum_{kl}\langle kl\|\|cd\rangle \tilde{t}^{ab}_{kl} - \mathcal{P}^-_{ab}\sum_k \langle kb\|\|cd\rangle t^a_k$	$\frac{3}{4}V^4$	$\frac{5}{8}O^2V^4$
$I^{(6)}_{klic} = \langle kl\|\|ic\rangle - \sum_d t^d_i \langle kl\|\|cd\rangle$	$\frac{3}{2}O^3V$	
$I^{(7)}_{kacd} = \langle ka\|\|cd\rangle - \sum_l t^a_l \langle kl\|\|cd\rangle$	$\frac{3}{2}OV^3$	
$T^{(1)}_{ij} = \sum_{kc} r^c_k I^{(6)}_{jkic}$		
$T^{(2)}_{ab} = \sum_{kc} r^c_k I^{(7)}_{kabc}$		
$T^{(3)}_{ij} = \frac{1}{2}\sum_{kab}\langle jk\|\|ab\rangle r^{ab}_{ik}$		
$T^{(4)}_{ab} = \frac{1}{2}\sum_{ijc}\langle ij\|\|bc\rangle r^{ac}_{ij}$		

$$
\begin{aligned}
\sigma^{ab}_{ij} &= I^{(1i)}_{ijab} + \mathcal{P}^-_{ab}\left\{\sum_c \left(T^{(2)}_{ac} + T^{(4)}_{ac}\right) t^{bc}_{ij} + \sum_c r^{ac}_{ij} F^{(2)}_{bc} \right.\\
&\quad \left. - \sum_k I^{(2b)}_{ijkb} r^a_k\right\} + \frac{1}{2}\sum_{kl} I^{(4b)}_{ijkl} r^{ab}_{kl}\\
&\quad + \mathcal{P}^-_{ij}\mathcal{P}^-_{ab}\left\{\sum_{kd} t^{ad}_{jk}\left(\sum_P M^P_{db} M^{2R}_{kiP} + I^{(2i)}_{kidb}\right)\right.\\
&\quad \left. - \sum_P M^P_{ja}\left(\sum_c M^P_{db} r^c_i + M^{2R}_{ibP}\right) + I^{(3i)}_{ijab}\right\}\\
&\quad + \mathcal{P}^-_{ij}\left\{\sum_k r^{ab}_{jk} F^{(2)}_{ik} + \sum_k t^{ab}_{jk}\left(\sum_c F^{(2)}_{kc} r^c_i + T^{(1)}_{ik} + T^{(3)}_{ik}\right)\right\}\\
&\quad - \frac{1}{2}\sum_{kl} t^{ab}_{kl}\left(\mathcal{P}^-_{ij}\mathcal{P}^-_{kl}\sum_P M^{2R}_{liP} M^P_{kj} - \frac{1}{2} I^{(4i)}_{ijkl}\right), \qquad (29)
\end{aligned}
$$

$$
I^{(1i)}_{ijab} = \frac{1}{2}\sum_{cd}\left(\mathcal{P}^-_{ab}\sum_P M^P_{ca} M^P_{db}\right) r^{cd}_{ij}, \qquad (30)
$$

$$
I^{(2i)}_{kidb} = \sum_{lc}\left(\sum_P B^P_{ld} B^P_{kc}\right) r^{bc}_{il}, \qquad (31)
$$

$$
I^{(3i)}_{ijab} = \sum_{kc}\left(\sum_P M^P_{ki} M^P_{cb}\right) r^{ac}_{jk}, \qquad (32)
$$

$$
I^{(4i)}_{ijkl} = \sum_{cd}\left(\mathcal{P}^-_{cd}\sum_P B^P_{kc} B^P_{ld}\right) r^{cd}_{ij}. \qquad (33)
$$

Here again $I^{(1i)}$, $I^{(2i)}$, $I^{(3i)}$, and $I^{(4i)}$ are computed using the three-tensor contraction procedure and then immediately added to the result. M, F, T, $I^{(2b)}$, and $I^{(4b)}$ intermediates in Eqs. (27)–(29) are given Tables II and IV. The intermediates from Table II are computed once, following a CD/RI-CCSD calculation, whereas the rest is recomputed at each Davidson iteration using the most current r-amplitudes.

For the computation of a σ-vector in the Davidson iterative procedure, the storage requirement for CD/RI-EOM-EE implementation becomes

$$
\frac{5}{2}O^2M + 5OVM + \frac{5}{2}V^2M + \frac{3}{2}O^4 + \frac{3}{2}O^3V + \frac{21}{4}O^2V^2. \qquad (34)
$$

For the naphthalene example from Sec. III B, the RI version of EOM-EE reduces the amount of required memory by a factor of 24 relative to the canonical implementation, that is, the conventional EOM-EE, using Eq. (26), needs 30795 Mwords (241 GB), whereas CD/RI-EOM-EE (Eq. (34)) uses 1275 Mwords (10 GB). The number of floating point operations in the σ-vector update procedure

TABLE IV. M and T intermediates for CD/RI EOM-EE-CCSD.

Equation	Memory	Flops
$M^{2R}_{iaP} = \sum_{ld} B^P_{ld} r^{ad}_{il}$	OVM	
$M^{2R}_P = \sum_{ld} B^P_{ld} r^d_l$		
$M^{3R}_{iaP} = \sum_c B^P_{ac} r^c_i$	OVM	
$M^{2R}_{ijP} = \sum_c B^P_{ic} r^c_j$	O^2M	
$M^{2R}_{abP} = \sum_k B^P_{ka} r^b_k$	V^2M	
$M^{2RT}_{iaP} = \sum_k M^{2R}_{kiP} t^a_k$	OVM	
$T^{(1)}_{ij} = \sum_P M^P_{ji} M^{2R}_P - \sum_{kP} M^{2R}_{jkP} M^P_{ki}$		
$T^{(2)}_{ab} = \sum_{kP} B^P_{kb}\left(M^{3R}_{kaP} - M^{2RT}_{kaP}\right) - \sum_P M^P_{ba} M^{2R}_P$		
$T^{(3)}_{ij} = \sum_{aP} B^P_{ja} M^{2R}_{iaP}$		
$T^{(4)}_{ab} = \sum_{iP} B^P_{ib} M^{2R}_{iaP}$		
$I^{(1i)}_{ijab}$ (Eq. (30))		$\frac{5}{8}O^2V^4$
$I^{(2i)}_{kidb}$ (Eq. (31))		$2O^3V^3$
$I^{(3i)}_{ijab}$ (Eq. (32))		$2O^3V^3$
$I^{(4i)}_{ijkl}$ (Eq. (33))		$\frac{5}{8}O^4V^2$
$\sum_{kd} t^{ad}_{jk}\left(\sum_P M^P_{db} M^{2R}_{kiP} + I^{(2i)}_{kidb}\right)$	$3O^2V^2$	$5O^3V^3$
$\sum_{kl} t^{ab}_{kl}\left(\mathcal{P}^-_{ij}\mathcal{P}^-_{kl}\sum_P M^{2R}_{liP} M^P_{kj} - \frac{1}{2} I^{(4i)}_{ijkl}\right)$	$\frac{3}{4}O^4$	$\frac{5}{8}O^4V^2$

for both conventional and RI/CD implementations scales as $\mathcal{O}(N^6)$. The cost of EOM-EE is $\frac{5}{8}O^2V^4 + \frac{3}{4}O^3V^3 + \frac{5}{8}O^4V^2$, whereas RI/CD-EOM-EE takes $\frac{5}{8}O^2V^4 + 9O^3V^3 + \frac{5}{4}O^4V^2$ operations. There is a larger number of O^3V^3 contractions in the latter case, leading to a bigger prefactor. This is the result of the on-the-fly reassembly of some fourth-order intermediates that are stored in memory in the case of conventional EOM-EE.

E. EOM-IP-CCSD and CD/RI EOM-IP-CCSD

In EOM-IP-CCSD (EOM-CCSD for ionization potentials), the operator R is not particle-conserving:

$$R_n(N-1) = \frac{1}{n!^2}\sum r^{bc\cdots}_{ijk\cdots} ib^+ jc^+ k\cdots . \tag{35}$$

In EOM-IP-CCSD, R is truncated at the two-hole-one-particle excitation level. The equations for σ-vectors are as follows:

$$\begin{aligned}
\sigma_i &= -\sum_j F^{(2)}_{ij} r_j + \sum_{jb} F^{(2)}_{jb} r^b_{ij} + \frac{1}{2}\sum_{jkb} I^{(6)}_{kjib} r^b_{jk},\\
\sigma^a_{ij} &= -\sum_k r_k I^{(2)}_{ijka} + \mathcal{P}^-_{ij}\sum_k r^a_{jk} F^{(2)}_{ik} + \sum_b r^b_{ij} F^{(2)}_{ab}\\
&\quad - \mathcal{P}^-_{ij}\sum_{kb} I^{(1)}_{jbka} r^b_{ik} + \frac{1}{2}\sum_{kl} I^{(4)}_{ijkl} r^a_{kl} + \sum_b t^{ab}_{ij} T^{(4)}_b,\\
T^{(4)}_a &= \frac{1}{2}\sum_{klb}\langle kl||ab\rangle r^b_{kl},
\end{aligned}$$

where F and I intermediates are collected in Tables I and III. Memory requirements for the σ update procedure are as follows:

$$\frac{3}{4}O^4 + 3O^3V + \frac{11}{4}O^2V^2. \tag{36}$$

This estimate excludes any three-dimensional quantities, e.g., EOM-IP amplitudes.

The CD/RI equations are derived following the same procedure as in the EOM-EE case. Specifically, we break $I^{(1)}$ and $I^{(6)}$ into the individual terms and use $I^{(2b)}$ and $I^{(4b)}$ from Table II instead of $I^{(2)}$ and $I^{(4)}$. The resulting CD/RI equations are as follows:

$$\begin{aligned}
\sigma_i &= -\sum_j F^{(2)}_{ij} r_j + \sum_{jb} F^{(2)}_{jb} r^b_{ij} - \sum_{jP} M^P_{ji}\left(\sum_{kb} B^P_{kb} r^b_{jk}\right),\\
\sigma^a_{ij} &= -\sum_k r_k I^{(2b)}_{ijka} + \mathcal{P}^-_{ij}\left[\sum_k F^{(2)}_{ik} r^a_{jk} - I^{(1i)}_{jia}\right.\\
&\quad \left. + \sum_P M^P_{ja}\left(\sum_{kb} B^P_{kb} r^b_{ik}\right) - \sum_{lc} t^{ac}_{jl} I^{(2i)}_{ilc}\right]\\
&\quad + \sum_b r^b_{ij} F^{(2)}_{ab} + \frac{1}{2}\sum_{kl} I^{(4b)}_{ijkl} r^a_{kl} + \sum_b t^{ab}_{ij} T^{(4)}_b,
\end{aligned}$$

$$\begin{aligned}
I^{(1i)}_{ija} &= \sum_{kb}\left(\sum_P M^P_{kj} M^P_{ba}\right) r^b_{ik},\\
I^{(2i)}_{ilc} &= \sum_{kb}\left(\sum_P B^P_{kc} B^P_{lb}\right) r^b_{ik},\\
T^{(4)}_b &= \sum_{kP} B^P_{kb}\left(\sum_{lc} B^P_{lc} r^c_{kl}\right).
\end{aligned}$$

$T^{(4)}$, $I^{(1i)}$, and $I^{(2i)}$ are computed using the three-tensor contraction. M-intermediates are given in Table II.

The data size for σ-vector update in CD/RI-EOM-IP is approximately:

$$\frac{3}{2}O^2M + 2OVM + \frac{3}{2}V^2M + \frac{3}{4}O^4 + \frac{3}{2}O^3V + \frac{3}{4}O^2V^2. \tag{37}$$

For the naphthalene example, memory savings achieved by using RI are limited to about 20%, that is, conventional EOM-IP using Eq. (36) requires 477 Mwords (3.7 GB), whereas CD/RI-EOM-IP, Eq. (37) needs 382 Mwords (3.0 GB). The difference in memory requirements is not as large as in the case of EOM-EE because EOM-IP does not use the OVVV and VVVV blocks of the ERIs.

The number of floating point operations in the σ-vector update procedure for both implementations is $\mathcal{O}(N^5)$. The CD/RI scheme requires six $\mathcal{O}(N^5)$-type contractions, the dominant contraction being O^2V^2M. The canonical EOM-IP requires two $\mathcal{O}(N^5)$-type contractions, the dominant contraction being O^3V^2. Therefore, the σ-vector update procedure in CD/RI-EOM-IP is expected to be about three times slower than in canonical EOM-IP; however, some of this cost increase is offset by more favorable parallel scaling. Moreover, for fair comparison, the calculation of the intermediates should also be considered.

F. EOM-EA-CCSD and CD/RI EOM-EA-CCSD

In EOM-EA (EOM for electron attachment), the operator R is

$$R_n(N+1) = \frac{1}{n!^2}\sum r^{abc\cdots}_{jk\cdots} a^+ b^+ jc^+ k\cdots . \tag{38}$$

In EOM-EA-CCSD, R is truncated at the one-hole-two-particles level and the equations for the σ-vectors are as follows:

$$\begin{aligned}
\sigma^a &= \sum_c F^{(2)}_{ac} r^c + \sum_{kc} F^{(2)}_{kc} r^{ac}_k + \frac{1}{2}\sum_{kcd} I^{(7)}_{kacd} r^{cd}_k,\\
\sigma^{ab}_i &= \mathcal{P}^-_{ab}\sum_c F^{(2)}_{ac} r^{cb}_i - \sum_k F^{(2)}_{ki} r^{ab}_k - \sum_c I^{(3)}_{icab} r^c\\
&\quad + \frac{1}{2}\sum_{cd} I^{(5)}_{abcd} r^{cd}_i + \mathcal{P}^-_{ab}\sum_{kc} I^{(1)}_{kbic} r^{ac}_k - \sum_k T^{(3)}_k t^{ab}_{ik},\\
T^{(3)}_i &= \frac{1}{2}\sum_{kcd} r^{cd}_k \langle ki||cd\rangle,
\end{aligned}$$

where $F^{(2)}$ and I intermediates are given in Tables I and III. The disk requirements for the EOM-EA σ update procedure

are estimated at

$$\frac{7}{2}O^2V^2 + 3OV^3 + \frac{3}{4}V^4. \quad (39)$$

In the CD/RI scheme, we break all I-intermediates from the above equations:

$$\sigma^a = \sum_c F^{(2)}_{ac} r^c + \sum_{kc} F^{(2)}_{kc} r^{ac}_k + \sum_{dP}\left(\sum_{kc} B^P_{kc} r^{cd}_k\right) M^P_{da},$$

$$\sigma^{ab}_i = \mathcal{P}^-_{ab}\Bigg[\sum_c F^{(2)}_{ac} r^{cb}_i - I^{(1i)}_{iab} + \sum_P M^P_{ia}\left(\sum_c M^P_{cb} r^c - \sum_{kc} B^P_{kc} r^{bc}_k\right)\Bigg] - \sum_k F^{(2)}_{ik} r^{ab}_k - \sum_{kl} t^{ab}_{kl}\left[\sum_P\left(\sum_c B^P_{kc} r^c\right) M^P_{li} - I^{(2i)}_{kli}\right] + \mathcal{P}^-_{ab}\left\{\sum_{ld}\left[I^{(3i)}_{ldb} - \sum_P\left(\sum_c B^P_{lc} r^c\right) M^P_{db}\right] t^{ad}_{il}\right\} - \sum_l\left[\sum_c F^{(2)}_{lc} r^c + \sum_{dP} B^P_{ld}\left(\sum_{kc} B^P_{kc} r^{cd}_k\right)\right] t^{ab}_{il} + I^{(4i)}_{iab},$$

$$I^{(1i)}_{iab} = \sum_{kc}\left(\sum_P M^P_{ki} M^P_{cb}\right) r^{ac}_k,$$

$$I^{(2i)}_{kli} = \frac{1}{2}\sum_{cd}\left(\sum_P B^P_{kc} B^P_{ld}\right) r^{cd}_i,$$

$$I^{(3i)}_{ldb} = \sum_{kc}\left(\sum_P B^P_{kd} B^P_{lc}\right) r^{bc}_k,$$

$$I^{(4i)}_{iab} = \sum_{cd}\left(\sum_P M^P_{ca} M^P_{db}\right) r^{cd}_i.$$

$I^{(1i)}$, $I^{(2i)}$, $I^{(3i)}$, and $I^{(4i)}$ are computed using the three-tensor contraction. M-intermediates are from Table II.

The storage requirement for CD/RI-EOM-EA σ-vector update is:

$$\frac{3}{2}O^2M + 2OVM + \frac{3}{2}V^2M + \frac{3}{4}O^2V^2. \quad (40)$$

To again illustrate memory savings using the naphthalene example, conventional EOM-EA using Eq. (39) uses 20408 Mwords (159 GB), whereas CD/RI-EOM-EA Eq. (40) needs only 360 Mwords (2.8 GB).

Because the most expensive in terms of storage intermediates have been eliminated in the CD/RI implementation, the procedure requires about 57 times less memory.

Similar to EOM-IP-CCSD, the number of floating point operations for both implementations scales as $\mathcal{O}(N^5)$. The dominant $\mathcal{O}(N^5)$-type contraction, out of two in canonical EOM-EA, is the OV^4-type, whereas in CD/RI, which has ten $\mathcal{O}(N^5)$-type contractions, the dominant one is V^4M-type.

IV. BENCHMARKS

The errors introduced by the RI and CD approximations have been extensively benchmarked for quantities like total energies, molecular structures, dipole moments, and excitation energies;[7,15,18,20,36,38–40,50–54] for a recent review, see Ref. 19. Total energies have been analyzed for density functional theory,[18–20,36,50] Hartree-Fock,[15,18–20,50,51] and MP2 methods.[7,15,18–20,38,50,51] Typical errors in absolute energies are in a millihartree (mE_h) range [or 0.01 kcal/(mol-electron)] for common auxiliary basis sets[20,36,38] and for CD with a threshold of 10^{-4}.[18,19,50]

The accuracy in energy differences, such as activation energies,[16] is better by a factor of 2-3 in comparison to total energies due to error cancellation. The errors in dipole moments computed with RI[38,39] and CD[19] are below 0.01 D and are usually an order of magnitude smaller that the errors due to the incompleteness of basis sets. The RI/CD bond lengths are within 0.01 pm from the respective full calculations.[20,39,53] Aquilante *et al.*[19] have also reported vertical excitation energies (computed with CASSCF and CASPT2) that show average errors less than 0.01 eV and 0.001 eV for thresholds of 10^{-3} and 10^{-4}, respectively. The effect of the RI approximation on excitation energies within an approximate second-order coupled-cluster model, CC2, has been thoroughly investigated by Köhn and Hättig who reported errors of 0.01 eV or less.[53]

In the present paper, we focus mostly on the effect of using RI/CD representations on energy differences between different electronic states, such as electronic excitation energies and ionization/electron attachment energies. We also consider energy differences along potential energy surfaces.

We compare the timings for RI/CD versus canonical implementations and investigate the parallel performance of the code.[55] All calculations were performed on designated benchmark nodes. The hardware configuration is Xeon X5675 (2 processors, 6 cores each, 3.0 GHz, 12 Mb cache), 128 GB RAM, RAID 0 4×600 GB=2.2 TB. This configuration was referred to as Xeon-USC in our previous benchmark study.[34]

We use the following test cases:

1. Phenolate form of the anionic chromophore of the photoactive yellow protein (PYPb).[56,57] We perform CCSD calculations as well as EOM-EE/IP/EA-CCSD. We consider the energy difference between the cis- and trans-isomers and electronic energy differences between various states (electronically excited, electron-attached, and ionized states). The calculations were performed with three basis sets — 6-31+G(d,p) (test1), aug-cc-pVDZ (test2), and cc-pVTZ (test3).
2. Cluster of two methylated uracils and a water molecule (test4).[34] Energy differences between different electronic states are considered.
3. Tetramer of 4 nucleobases, AATT, from Ref. 58 (test5).
4. Oligoporphyrin dimer used in previous benchmarks[34,59] (test6).
5. Cluster of methylated uracil, mU, and water from Ref. 60. We focus on the potential energy profiles along the proton-transfer reaction coordinate.

The following thresholds were used in CCSD and EOM-CCSD calculations:[61] T-amplitudes convergence of $|T_n - T_{n-1}| \leq 10^{-4}$, energy convergence $|E_n - E_{n-1}| \leq 10^{-6}$, Davidson's procedure convergence $|R_n| \leq 10^{-5}$ (here R_n is the

TABLE V. Test systems used for benchmarks, converged CCSD correlation energies (hartree), and number of CC iterations.

Name	Molecule	Formula	Symm	N_{el}	Basis[a]	# b.f.	E_{corr}	N_{iter}
Test1	PYPb[b,c]	$C_9O_3H_7$	C_s	86	6-31+G(d,p)	263	−1.838 498	14
Test2	PYPb[b,c]	$C_9O_3H_7$	C_s	86	aug-cc-pVDZ	363	−1.955 977	14
Test2-fc	PYPb[b,d]	$C_9O_3H_7$	C_s	86	aug-cc-pVDZ	363	−1.875 348	14
Test3	PYPb[b,d]	$C_9O_3H_7$	C_s	86	cc-pVTZ	489	−2.147 251	14
Test4	$(mU)_2 \cdot H_2O$[c]	$C_{12}O_5N_4H_{18}$	C_1	158	6-31+G(d,p)	489	−3.334 026	12
Test5	AATT[d]	$C_{20}O_4N_{14}H_{22}$	C_1	386	6-311+G(d,p)	968	−5.975 572[e]	12
Test6	Porphyrin[d]	$C_{46}N_{12}H_{26}$	D_{2h}	272	cc-pVDZ	942	−7.995 344[f]	11

[a]References 64 and 65 for 6-31+G(d,p), Refs. 66 and 65 for 6-311+G(d,p), and Ref. 67 for cc-pVDZ, cc-pVTZ and aug-cc-pVDZ.
[b]PYPb: anti-syn conformation.
[c]Core electrons active.
[d]Core electrons frozen.
[e]Calculation using conventional CCSD with RI integrals (direct_ri=true, rimp2-cc-pVDZ[38] auxiliary basis set).
[f]From Ref. 59.

Davidson residual), and threshold for subspace expansion in Davidson's procedure $|R_n| > 10^{-5}$. Table V lists parameters for different benchmark examples. All electrons were active in test1; test2 was executed with and without frozen core; core electrons were frozen in test3–test6. In some cases, we also employed Frozen Natural Orbitals (FNO) approximation.[62] All Cartesian geometries and relevant energies are given in the supplementary material.[63] All calculations were performed using regular (i.e., employing non-decomposed ERI) Hartree–Fock procedure. Correlation energies in Table V are for canonical calculations (using full ERI) except for test5.

Table VI presents the comparison of the canonical CCSD calculation and RI/CD approximations. We note that the errors in total CCSD energies for RI and CD/10^{-3} approximations are comparable (and are in a millihartree range). However, the rank of CD/10^{-3} is often less than that of RI giving

TABLE VI. CCSD errors and wall times (s) using 12 cores.

Name	Method	Rank	CCSD error	CCSD wall	Ratio[a]	CD wall time[b]
Test1	Full			1894		
	RI/rimp2-aug-cc-pVDZ	1099	8.10×10^{-4}	1771	0.94	
	CD/10^{-1}	135	1.49×10^{-1}	1500	0.79	85
	CD/10^{-2}	505	1.71×10^{-3}	1591	0.84	530
	CD/10^{-3}	715	2.74×10^{-4}	1642	0.87	978
	CD/10^{-4}	1065	8.27×10^{-6}	1773	0.94	2035
Test2	Full			9490		
	RI/rimp2-aug-cc-pVDZ	1099	7.8×10^{-4}	5175	0.55	
	CD/10^{-3}	804	2.1×10^{-3}	4750	0.50	2345
Test2-fc	Full			2870		
	RI/rimp2-aug-cc-pVDZ	1099	1.1×10^{-3}	2847	0.99	
	CD/10^{-3}	804	4.2×10^{-4}	2626	0.91	2348
Test3	Full			21257		
	RI/rimp2-cc-pVTZ	1256	1.4×10^{-3}	9209	0.43	
	CD/10^{-3}	1629	1.9×10^{-4}	10367	0.49	15491
Test4	Full			110379		
	RI/rimp2-aug-cc-pVDZ	2067	1.6×10^{-3}	85425	0.77	
	CD/10^{-3}	1335	6.4×10^{-4}	88198	0.80	10301
	CD/10^{-3}/FNO[c]	1335		25464	0.15	10392
Test5	Full/rimp2-cc-pVTZ/FNO[d,e,f]			726.4 h		
	RI/rimp2-cc-pVTZ/FNO[d]	3738		699.2 h	0.96	
	CD/10^{-2}/FNO[d]	1688		443.6 h	0.61	15.4 h
Test6	Full		4.7×10^{-3}	211.0 h		
	RI/rimp2-cc-pVDZ	3612	4.7×10^{-3}	194.9 h	0.92	
	RI/rimp2-cc-pVDZ/FNO[g]	3612		63.3 h	0.30	

[a]Ratio = time (RI/CD)/time(full) for CCSD iterations.
[b]Wall time for Cholesky decomposition (CD) procedure.
[c]Using FNO (see Ref. 62), 99.50% occupation threshold and frozen core (350 active orbitals). CCSD converges in 11 iterations.
[d]Using FNO, 99.50% occupation threshold and frozen core (649 active orbitals). CCSD converges in 12 iterations.
[e]From Ref. 34.
[f]Canonical CCSD calculations using RI integrals (direct_ri=true).
[g]Using FNO, 99.50% occupation threshold and frozen core (754 active orbitals). CCSD converges in 16 iterations. Memory settings: test1 — 20 GB, test2 — 50 GB, test3 — 80 GB, test4 — 48 GB, test5 — 100 GB, and test6 — 100 GB.

TABLE VII. Wall time per CCSD iteration (s) using 80 GB RAM.

Job	1 core	4 cores	8 cores	12 cores
Full	46405	14278 (×3.25)	9506 (×4.88)	7973 (×5.82)
RI/rimp2-aug-cc-pVDZ	39347	10283 (×3.83)	5539 (×7.10)	4342 (×9.06)
CD/10^{-3}	37330	9889 (×3.78)	4973 (×7.51)	4185 (×8.92)

rise to a more significant speed-up (the situation is reversed in for test3 which uses the cc-pVTZ basis). We also note that CD/10^{-4} leads to the rank comparable to the size of the auxiliary basis in RI (1065 versus 1099), but yields two orders of magnitude more accurate total energies (error 8.27×10^{-6} versus 8.10×10^{-4} hartree).

Overall, RI/CD CCSD calculations are 10%–60% faster than the canonical implementation. We observe a more significant speed-up for larger calculations, e.g., compare test4, test3, and test2 versus test1, likely because the I/O penalties are more pronounced for larger jobs. For the same molecule, we observe more significant speed-up in larger bases (compare test3 versus test2 and test1), because larger bases have more linear dependencies. Using test4 as an example, we observe that combination of CD with FNO approximation leads to a very impressive speed-up, i.e., CD/10^{-3}/FNO calculation takes only 15% of the time of the full CCSD calculation.

Finally, let us consider two large examples, i.e., a nucleobase tetramer (AATT)[34,58] (test5, C_1 symmetry, 966 basis functions, and 38 core orbitals frozen) and the oligoporphyrin dimer (test6, D_{2h} symmetry, 942 basis functions, and 58 core orbitals frozen)[34,59] and compare them to the canonical calculations.[34] In test5, we also employ FNO approximation (279 out of 830 virtuals frozen, total 649 active orbitals). First, we note significant reduction in disk requirements for both examples (e.g., 382 GB versus 2.8 TB for AATT). For test6, the first RI/CD CCSD iteration is more than twice faster than in the canonical implementation (6.33 h for RI CCSD versus 13.2 h for the canonical CCSD[34]). However, we observe a slowdown of the subsequent iterations due to the increasing number of T-amplitudes that need to be handled by the DIIS procedure. The average time per iteration for oligoporphyrin is 12 h (194.9 h total time, 16 iterations), although the first iteration is two times faster (6.33 h). We also computed oligoporphyrin in combination with FNO (130 out of 749 virtual orbitals frozen, 754 active orbitals) with time for first iteration 2.35 h and the average time per iteration 3.95 h. For AATT, we observe a similar speed-up of RI CCSD iterations, the first RI CCSD iteration takes 39 h (to be compared to 60 h in the canonical implementation[34]). AATT computed with CD/10^{-2} yields a rank of 1688 and the first CCSD iteration takes 28.25 h, to be compared with 60 h in the canonical implementation and 39 h with RI/rimp2-cc-pVTZ.

A more detailed breakdown of timings for CCSD calculations is given in Table S1 of the supplementary material.[63] As expected, the evaluation of Eq. (11) takes a significant fraction of time, especially in larger bases (it scales as O^2V^4). Evaluation of Eq. (9), which contains one O^3V^3 (third term) and one O^4V^2 (the last term) contractions, is also significant and becomes dominant in an electron-rich case, test6.

The time used for the decomposition and integral transformation steps is also shown in Table VI. Since the present implementation of the decomposition algorithm does not use point group symmetry, the timings for test1–test3 are relatively large. We note that for test4 (C_1), the time of decomposition with a threshold of 10^{-3} is about 12% of the total time for CCSD iterations.

We investigate parallel performance using test4 (Table VII). The parallel scaling is improved, e.g., the canonical implementation shows a factor of 6 speed-up on 12 cores, whereas the RI/CD code is accelerated by a factor of 9. Thus, the speed-up relative to the canonical CCSD code becomes more pronounced on 12 cores, e.g., on a single CPU, RI/CD calculation is about 20% faster, whereas on 12 cores, it is almost a factor of two faster than the canonical code. This improvement in the parallel performance is due to the significant reduction of the amount of data handled in the CCSD calculations.

Table VIII shows EOM energies and timings for test1 and test2; the results for test2-fc and test3 are presented in Table IX. We note that RI and CD/10^{-3} give comparable errors in excitation, attachment, and ionization energies, i.e., less than 0.01–0.001 eV. These errors are consistent with those reported for the CASSCF and CASPT2 methods.[19] The errors in the energies are systematically reduced with the Cholesky threshold decrease from 10^{-2} to 10^{-4} for all methods. We observe that a threshold of 10^{-2} yields errors of ~0.03 eV, which is acceptable in many situations and is less than error bars of EOM-CCSD.

For test1, the timings for RI/CD EOM methods are slower than of the canonical implementation due to the increased number of contractions, as explained in Sec. III. However, in a larger basis (aug-cc-pVDZ versus 6-31+G*), the gap shrinks for EOM-IP (total RI/CD EOM time is almost the same as of the canonical calculation), and RI/CD EOM-EE shows 60%–70% speed-up. Further increase of the basis (to cc-pVTZ) leads to an additional speed-up, i.e., RI/CD EOM-EE calculations for test3 take 25% of the full EOM-EE time. This is because the increased number of time-determining contractions in RI/CD EOM-EE (7 for RI/CD EOM-EE versus 3 N^6 operations in canonical EOM) is offset by the significantly reduced disk and memory usage by RI/CD EOM that reduces I/O penalties and improves parallel scaling. For example, for test5 (AATT) EOM-EE-CCSD calculations (with frozen core and FNO) the estimated disk usage is 7.2 TB, whereas for the corresponding RI/rimp2-cc-pVTZ calculation it is only 590 GB. For test3, we observe that canonical

TABLE VIII. EOM-CCSD energies for the 2 lowest states in each irrep and errors in energy differences (eV), and wall times for EOM (s) using 12 cores.

Method	EOM time				EOM				
	Int[a]	Iter[b]	Total[c]	Ratio[d]	calls[e]	1A′	2A′	1A″	2A″
Test1									
EOM-EE-CCSD	342	3344	3686		46	3.158 eV	4.233 eV	3.860 eV	4.171 eV
RI[f]	65	6641	6706	1.8 (2.0)	46	1.0×10^{-4}	1.0×10^{-4}	1.2×10^{-3}	1.2×10^{-3}
CD/10^{-2}	56	6093	6149	1.7 (1.8)	46	2.8×10^{-2}	3.5×10^{-2}	2.5×10^{-2}	2.0×10^{-2}
CD/10^{-3}	59	6276	6335	1.7 (1.9)	46	4.2×10^{-3}	6.7×10^{-3}	6.5×10^{-3}	3.9×10^{-3}
CD/10^{-4}	64	6588	6652	1.8 (2.0)	46	2.0×10^{-4}	6.0×10^{-4}	1.0×10^{-3}	9.0×10^{-4}
EOM-EA-CCSD	351	486	837		31	3.931 eV	4.329 eV	3.700 eV	5.268 eV
RI[f]	65	984	1049	1.3 (2.0)	31	1.0×10^{-4}	$<10^{-4}$	1.3×10^{-3}	1.0×10^{-4}
CD/10^{-2}	56	710	766	0.9 (1.5)	31	1.9×10^{-2}	9.8×10^{-3}	1.5×10^{-2}	2.5×10^{-2}
CD/10^{-3}	59	788	847	1.0 (1.6)	31	7.5×10^{-3}	5.6×10^{-3}	2.5×10^{-3}	6.6×10^{-3}
CD/10^{-4}	65	975	1040	1.2 (2.0)	31	4.0×10^{-4}	1.4×10^{-3}	3.0×10^{-4}	1.1×10^{-3}
EOM-IP-CCSD	93	46	139		26	4.328 eV	6.758 eV	2.735 eV	5.353 eV
RI[f]	65	132	197	1.4 (2.9)	26	6.0×10^{-4}	8.0×10^{-4}	1.3×10^{-3}	1.1×10^{-3}
CD/10^{-2}	57	108	165	1.2 (2.4)	26	1.3×10^{-2}	6.9×10^{-3}	2.3×10^{-3}	2.8×10^{-3}
CD/10^{-3}	60	117	177	1.3 (2.5)	26	2.0×10^{-4}	1.0×10^{-4}	2.0×10^{-4}	7.0×10^{-4}
CD/10^{-4}	65	132	197	1.4 (2.9)	26	$<10^{-4}$	$<10^{-4}$	1.0×10^{-4}	$<10^{-4}$
Test2									
EOM-EE-CCSD	20021	39991	60012		49	3.167 eV	4.148 eV	3.349 eV	5.666 eV
RI[f]	139	18522	18661	0.3 (0.5)	49	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-3}	1.1×10^{-3}
CD/10^{-3}	125	18572	18697	0.3 (0.5)	49	5.0×10^{-4}	1.2×10^{-3}	2.0×10^{-4}	$<10^{-4}$
EOM-IP-CCSD	245	70	315		26	4.467 eV	6.772 eV	2.904 eV	5.462 eV
RI[f]	165	211	376	1.2 (3.0)	26	9.0×10^{-4}	8.0×10^{-4}	1.3×10^{-3}	8.0×10^{-4}
CD/10^{-3}	159	196	355	1.1 (2.8)	26	9.0×10^{-4}	7.0×10^{-4}	9.0×10^{-4}	8.0×10^{-4}

[a]Time for calculations of the EOM-CCSD intermediates for the Davidson procedure.
[b]Time for EOM iterations.
[c]Total EOM time (intermediates + Davidson iterations).
[d]Ratio = time (RI/CD)/time (full). The first value is the ratio of total EOM times; the ratio for Davidson iterations is given in parentheses.
[e]σ-vector update calls.
[f]rimp2-aug-cc-pVDZ auxiliary basis.

EOM shows rather poor parallel scaling (CPU 102655 s, wall 93432 s, ratio=1.09), whereas for RI EOM we see more than a 10 fold CPU/wall ratio (CPU 228202 s, wall 21553 s, ratio=10.58), leading to an overall 5-fold speedup of Davidson iterations. Thus, RI/CD implementation of EOM not only extends the applicability of the method to larger systems that may not be accessible by canonical EOM-CCSD due to disk/memory bottlenecks, but also improves timings of the

TABLE IX. EOM-CCSD energies for the 2 lowest states in each irrep and errors in energy differences (eV), and wall times (s) using 12 cores.

Method	EOM time				EOM				
	Int[a]	Iter[b]	Total[c]	Ratio[d]	calls[e]	1A′	2A′	1A″	2A″
Test2-fc									
EOM-EE-CCSD	4565	6917	11482		48	3.166 eV	4.148 eV	3.344 eV	5.662 eV
RI[f]	68	9619	9687	0.8 (1.4)	48	1.0×10^{-4}	1.0×10^{-4}	1.1×10^{-3}	1.1×10^{-3}
CD/10^{-3}	62	8987	9049	1.3 (2.3)	48	5.0×10^{-4}	1.2×10^{-3}	2.0×10^{-4}	$<10^{-4}$
Test3									
EOM-EE-CCSD	47946	93432	141378		36	3.307 eV	4.333 eV	5.513 eV	5.690 eV
RI[g]	170	21552	21722	0.2 (0.2)	36	3.0×10^{-4}	1.0×10^{-4}	4.0×10^{-4}	3.0×10^{-4}
CD/10^{-3}	194	26839	27033	0.2 (0.3)	36	8.0×10^{-4}	9.0×10^{-4}	2.6×10^{-3}	2.4×10^{-3}
EOM-IP-CCSD	426	62	488		26	4.450 eV	6.795 eV	2.872 eV	5.447 eV
RI[g]	166	214	380	0.8 (3.5)	26	1.3×10^{-3}	1.3×10^{-3}	1.0×10^{-3}	8.0×10^{-4}
CD/10^{-3}	189	245	434	0.9 (4.0)	26	1.0×10^{-4}	1.0×10^{-4}	2.0×10^{-4}	3.0×10^{-4}

[a]Time for calculations of the EOM-CCSD intermediates for the Davidson procedure.
[b]Time for EOM iterations.
[c]Total EOM time (intermediates + Davidson iterations).
[d]Ratio = time (RI/CD)/time (full). The first value is the ratio of total EOM times; the ratio for Davidson iterations is given in parentheses.
[e]σ-vector update calls.
[f]rimp2-aug-cc-pVDZ auxiliary basis.
[g]rimp2-cc-pVTZ auxiliary basis.

TABLE X. EOM-IP-CCSD energies (absolute errors for RI/CD) and EOM wall times (s) for test4 (two lowest EOM roots).

Method	EOM time				EOM		
	Int[a]	Iter[b]	Total[c]	Ratio[d]	calls[e]	State 1	State 2
EOM-IP-CCSD	5088	503	5591		10	8.421 eV	8.858 eV
RI[f]	1866	576	2442	0.4	10	1.0×10^{-3}	1.0×10^{-3}
CD/10^{-3}	2083	514	2597	0.5	10	5.0×10^{-4}	2.0×10^{-4}
CD/10^{-3}/FNO[g]	541	270	811	0.2	10	1.6×10^{-2}	1.5×10^{-2}

[a]Time for calculations of the EOM-CCSD intermediates for the Davidson procedure.
[b]Time for EOM iterations.
[c]Total EOM time (intermediates + Davidson iterations).
[d]Ratio of total times: time (RI/CD)/time (full).
[e]Number of calls of σ-update procedure.
[f]rimp2-aug-cc-pVDZ auxiliary basis.
[g]Frozen core and FNO (threshold 99.50%) was used.

calculations by removing the overheads due to large size of the data.

The calculation time of the intermediates for EOM calculations is significantly reduced for all RI/CD methods, as illustrated by test2 timings revealing that the intermediates calculations (dominated by the VVVV block of the transformed integrals) take almost as much time as Davidson iterations. Thus, the overall CD/RI EOM-EE timings (Davidson iterations plus intermediates) are considerably faster (3–5 times) than those of the canonical code when only of few EOM roots are computed for large systems.

The detailed timings for RI EOM-EE-CCSD calculations are given in Table S2 of the supplementary material.[63] We observe that σ_2-vector update procedure takes most of total EOM time (96% for test3). Within it, the calculation of I^{li}_{ijab}, Eq. (30), is dominant (70% of the total EOM time for test3), as expected based on O^2V^4 operations required to evaluate this term.

Calculations of ionization energies for test4 (Table X) show the errors of the same order of the magnitude, 0.001 eV and 0.0001 eV for RI and CD/10^{-3}, respectively, as in the PYPb examples (test1 and test2). Calculations of EOM-IP σ-vectors with RI/CD are slightly slower than in the canonical calculations; however, the time required for calculation of intermediates is significantly smaller for RI/CD, resulting in more than 2-fold overall speedup. The speed-up for RI/CD EOM-IP is less than for RI/CD EOM-EE due to smaller size of the data used by EOM-IP, which does not involve VVVV and OVVV intermediates; thus, the canonical code shows much better parallel performance than EOM-EE (for full EOM-IP, CPU 587 s, wall 63 s, ratio = 9.45; for RI EOM-IP, CPU 2067 s, wall 214 s, ratio=9.70).

Using FNO (threshold 99.5%, 118 virtual orbitals frozen out of 410 total) significantly improves the total EOM timings making it more than 6 times faster than the full canonical calculation. The errors introduced by the FNO approximation are larger than those due to CD (~0.01 eV), but they are still acceptable for most applications. Thus, RI/CD in conjunction with FNO leads to significant reduction of both memory and computational cost requirements, with only minor losses in accuracy.

To quantify the errors in energy differences along potential energy surfaces, we consider two examples. We begin by considering the energy differences between two PYP isomers[57] shown in Table XI. The energy difference between two PYPb isomers (anti-syn and anti-anti) is 4.15 kcal/mol at the CCSD/6-31+G(d,p) level of theory. The errors introduced by RI and CD are 2.40×10^{-3} (rimp2-aug-cc-pVDZ), 6.42×10^{-2} (CD/10^{-2}), 2.12×10^{-2} (CD/10^{-3}), and 9.00×10^{-4} (CD/10^{-4}) kcal/mol; the errors are considerably smaller than the errors in the total CCSD correlation energy due to error cancellation. Note that even for the crudest CD threshold (10^{-2}), the error in the energy differences is quite satisfactory (~0.1 kcal/mol). The error cancellation effect is more pronounced for RI where the error in energy differences is more than 2 orders of magnitude less than the error in the total energy, whereas for CD the difference is more modest (about 1 order of magnitude). Thus, in terms of the energy differences, RI is more accurate than CD/10^{-3}, but is still slightly less accurate than CD/10^{-4}.

As a more challenging case, we consider scans along proton-transfer coordinate in ionized mU-H_2O cluster from Ref. 60. Figure 1 shows CCSD and EOM-IP-CCSD energies along the proton-transfer reaction coordinate computed in the 6-311+G(d,p) basis set. We note that RI features the smallest errors, both in terms of absolute values (around 10^{-4}–10^{-5} eV) and in terms of non-parallelity errors (NPEs) (4×10^{-5} and 5×10^{-5} eV for CCSD and EOM-IP-CCSD energies, respectively). This is because the auxiliary basis in RI is atom-centered and does not depend on geometry.[68] CD shows larger errors along the scan; however, the respective NPEs are small and do not exceed 0.001 eV for CD/10^{-3} and 0.0003 eV for CD/10^{-4}. We note that the range of changes in total energy

TABLE XI. Energy differences between PYPb isomers ($E_{anit-anti} - E_{anti-syn}$, kcal/mol) and the corresponding errors against full CCSD.

Method	Energy difference	Error (kcal/mol)	Error (hartree)
Full	4.1531		
RI/rimp2-aug-cc-pVDZ	4.1507	2.40×10^{-3}	3.82×10^{-6}
CD/10^{-2}	4.0889	6.42×10^{-2}	1.02×10^{-4}
CD/10^{-3}	4.1319	2.12×10^{-2}	3.37×10^{-5}
CD/10^{-4}	4.1540	9.00×10^{-4}	1.43×10^{-6}

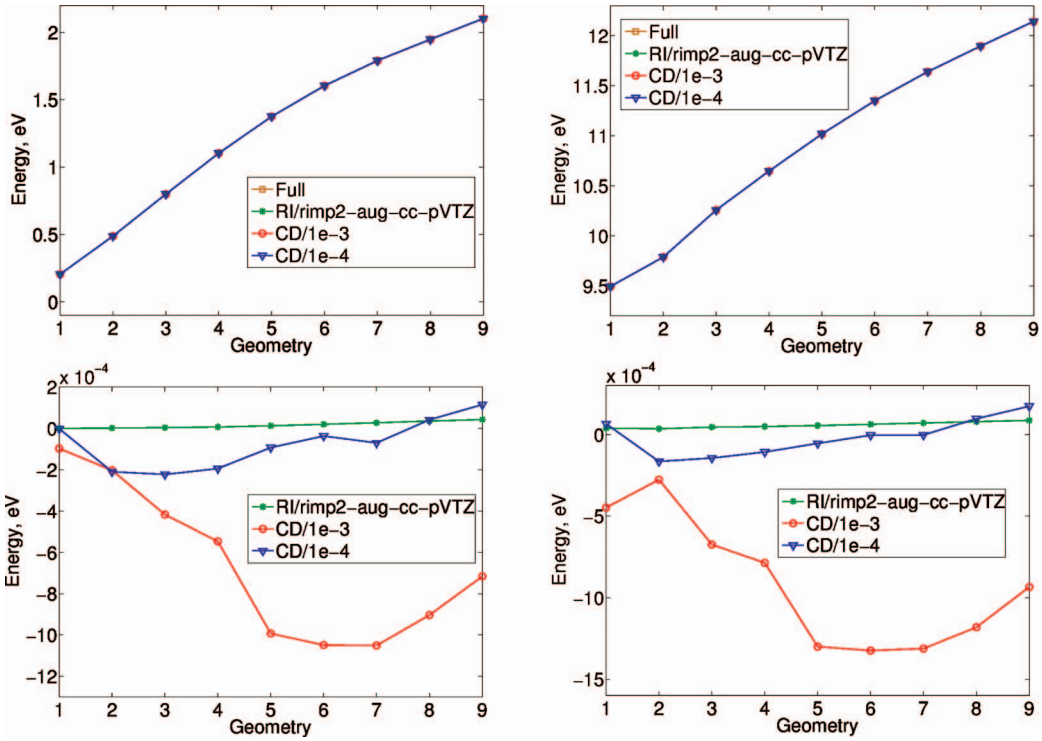

FIG. 1. Top: CCSD (left) and EOM-IP-CCSD (right) energies along the proton-transfer coordinate in mU-H_2O. Bottom: Errors of RI/rimp2-aug-cc-pVTZ and CD approximations.

along this scan is about 2 eV. Smooth behavior of the CD scans is consistent with small variations of the rank along this scan, e.g., for CD/10^{-3} and CD/10^{-4} the number of Cholesky vectors is 834±2 and 1188 ±3, respectively.

V. CONCLUSIONS

We present a new implementation of RI and Cholesky decompositions within the CCSD/EOM-CCSD suite of methods in the Q-Chem electronic structure package.[27,28] This implementation eliminates the storage of the most expensive four-index electron repulsion integrals and intermediates, such as VVVV, OVVV, and OVOV blocks of ERI, leading to a significant reduction in storage requirements and I/O overheads. The number of floating-point operations is reduced for CCSD; however, it is increased by approximately a factor of 3 in EOM calculations (σ-vectors update) because the transformed integrals and related intermediates, which are computed only once in canonical EOM, need to be reassembled at each Davidson iteration in the RI/CD implementation. However, this undesirable increase in computations is offset by significantly reduced I/O overheads. In a shared-memory parallel setting the reduction of I/O also leads to better CPU utilization and improved parallel scalability. When the calculation of the intermediates is included, the ratio between RI/CD and canonical EOM-EE timings is about 0.3–0.5 for moderate-size basis sets. The gains are more significant in large bases, e.g., a RI EOM-EE-CCSD/cc-pVTZ calculation takes only 15% of the time required for the full calculation. Additional computational savings can be achieved by combining RI/CD and FNO approaches.[62]

The accuracy of RI/CD implementations is benchmarked with an emphasis on energy differences, such as excitation energies. In agreement with previous benchmarks based on the CASSCF, CASPT2, and CC2 methods,[19,53] we observe that the errors in energy differences are smaller than the errors in total energies due to error cancellation. Typical errors in the CCSD correlation energy are less than a millihartree for the RI approximation with RI-MP2 auxiliary bases; however, the respective EOM errors are less than 0.001 eV. The accuracy of CD can be controlled by a single threshold. For a threshold of 10^{-4}, which results in a rank similar to RI, the errors in total energies are two orders of magnitude less than for RI; however, the errors in energy differences are roughly the same. This threshold is therefore recommended when high accuracy is required. We note that errors in excitation energies are quite small when using thresholds of 10^{-2} and 10^{-3} (less than 0.04 and 0.008 eV, respectively); therefore, these thresholds can be used in most calculations.

This paper presents our first step towards developing reduced-scaling CC/EOM-CC codes. While the present implementation does not reduce scaling of the calculations, it affords significant computational savings thus extending the applicability of these methods to larger systems. In order to achieve further gains, additional steps should be taken. Among promising strategies[19] are a tensor hyper-contraction approach,[21,22] local correlation schemes, and pair natural orbitals,[25,26,69,70] as well as reduced-rank representations of the CC/EOM amplitudes.[23,24,71,72]

ACKNOWLEDGMENTS

A.I.K. is grateful to Professor Juergen Gauss and Professor Andreas Köhn for stimulating discussions. Support for this work was provided through Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research and Basic Energy Sciences. A.I.K. also acknowledges support from the Humboldt Research Foundation (Bessel Award).

[1] J. A. Pople, "Theoretical models for chemistry," in *Energy, Structure and Reactivity: Proceedings of the 1972 Boulder Summer Research Conference on Theoretical Chemistry*, edited by D. W. Smith and W. B. McRae (Wiley, New York, 1973), pp. 51–61.
[2] T. Helgaker, P. Jørgensen, and J. Olsen, *Molecular Electronic Structure Theory* (Wiley & Sons, 2000).
[3] R. J. Bartlett, Mol. Phys. **108**, 2905 (2010).
[4] A. I. Krylov, Annu. Rev. Phys. Chem. **59**, 433 (2008).
[5] K. Sneskov and O. Christiansen, WIREs Comput. Mol. Sci. **2**, 566 (2011).
[6] R. J. Bartlett, WIREs Comput. Mol. Sci. **2**, 126 (2012).
[7] F. Weigend, M. Häser, H. Patzelt, and R. Ahlrichs, Chem. Phys. Lett. **294**, 143 (1998).
[8] J. L. Whitten, J. Chem. Phys. **58**, 4496 (1973).
[9] M. W. Feyereisen, G. Fitzgerald, and A. Komornicki, Chem. Phys. Lett. **208**, 359 (1993).
[10] O. Vahtras, J. Almlöf, and M. W. Feyereisen, Chem. Phys. Lett. **213**, 514 (1993).
[11] A. Komornicki and G. Fitzgerald, J. Chem. Phys. **98**, 1398 (1993).
[12] D. E. Bernhold and R. J. Harrison, Chem. Phys. Lett. **250**, 477 (1996).
[13] Y. Jung, A. Sodt, P. M. W. Gill, and M. Head-Gordon, Proc. Natl. Acad. Sci. U.S.A. **102**, 6692 (2005).
[14] N. H. F. Beebe and J. Linderberg, Int. J. Quantum Chem. **12**, 683 (1977).
[15] H. Koch, A. S. de Merás, and T. B. Pedersen, J. Chem. Phys. **118**, 9481 (2003).
[16] F. Aquilante, R. Lindh, and T. B. Pedersen, J. Chem. Phys. **127**, 114107 (2007).
[17] F. Aquilante, T. B. Pedersen, and R. Lindh, Theor. Chem. Acc. **124**, 1 (2009).
[18] F. Aquilante, L. Gagliardi, T. B. Pedersen, and R. Lindh, J. Chem. Phys. **130**, 154107 (2009).
[19] F. Aquilante, L. Boman, J. Boström, H. Koch, R. Lindh, A. S. de Merás, and T. B. Pedersen, "Cholesky decomposition techniques in electronic structure theory," in *Linear-Scaling Techniques in Computational Chemistry and Physics*, edited by R. Zaleśny, M. G. Papadopoulos, P. G. Mezey, and J. Leszczynski, Challenges and Advances in Computational Chemistry and Physics (Springer, 2011), pp. 301–343.
[20] F. Weigend, M. Kattannek, and R. Ahlrichs, J. Chem. Phys. **130**, 164106 (2009).
[21] R. M. Parrish, E. G. Hohenstein, T. J. Martínez, and C. D. Sherrill, J. Chem. Phys. **137**, 224106 (2012).
[22] E. G. Hohenstein, R. M. Parrish, C. D. Sherrill, and T. J. Martínez, J. Chem. Phys. **137**, 221101 (2012).
[23] F. Aquilante, T. K. Todorova, L. Gagliardi, T. B. Pedersen, and B. O. Roos, J. Chem. Phys. **131**, 034113 (2009).
[24] F. Aquilante and T. B. Pedersen, Chem. Phys. Lett. **449**, 354 (2007).
[25] C. Riplinger and F. Neese, J. Chem. Phys. **138**, 034106 (2013).
[26] D. Kats, T. Korona, and M. Schütz, J. Chem. Phys. **125**, 104106 (2006).
[27] Y. Shao, L. Fusti-Molnar, Y. Jung, J. Kussmann, C. Ochsenfeld, S. Brown, A. T. B. Gilbert, L. V. Slipchenko, S. V. Levchenko, D. P. O'Neill, R. A. Distasio, Jr., R. C. Lochan, T. Wang, G. J. O. Beran, N. A. Besley, J. M. Herbert, C. Y. Lin, T. Van Voorhis, S. H. Chien, A. Sodt, R. P. Steele, V. A. Rassolov, P. Maslen, P. P. Korambath, R. D. Adamson, B. Austin, J. Baker, E. F. C. Bird, H. Daschel, R. J. Doerksen, A. Dreuw, B. D. Dunietz, A. D. Dutoi, T. R. Furlani, S. R. Gwaltney, A. Heyden, S. Hirata, C.-P. Hsu, G. S. Kedziora, R. Z. Khalliulin, P. Klunziger, A. M. Lee, W. Z. Liang, I. Lotan, N. Nair, B. Peters, E. I. Proynov, P. A. Pieniazek, Y. M. Rhee, J. Ritchie, E. Rosta, C. D. Sherrill, A. C. Simmonett, J. E. Subotnik, H. L. Woodcock III, W. Zhang, A. T. Bell, A. K. Chakraborty, D. M. Chipman, F. J. Keil, A. Warshel, W. J. Hehre, H. F. Schaefer III, J. Kong, A. I. Krylov, P. M. W. Gill, and M. Head-Gordon, Phys. Chem. Chem. Phys. **8**, 3172 (2006).
[28] A. I. Krylov and P. M. W. Gill, WIREs Comput. Mol. Sci. **3**, 317 (2013).
[29] A. P. Rendell and T. J. Lee, J. Chem. Phys. **101**, 400 (1994).
[30] Turbomole User's Manual (version 6.4), 2012, see http://www.turbomole-gmbh.com; accessed 05/25/2013.
[31] F. Aquilante, L. de Vico, N. Ferré, G. Ghigo, P.-Å. Malmqvist, P. Neogrády, T. B. Pedersen, M. Pitonák, M. Reiher, B. Roos, L. Serrano-Andrés, M. Urban, V. Veryazov, and R. Lindh, J. Comput. Chem. **31**, 224 (2010).
[32] S. Wilson, Comput. Phys. Commun. **58**, 71 (1990).
[33] G. H. Golub and C. F. Van Loan, *Matrix Computations* (Johns Hopkins University Press, 1996).
[34] E. Epifanovsky, M. Wormit, T. Kuś, A. Landau, D. Zuev, K. Khistyaev, P. Manohar, I. Kaliman, A. Dreuw, and A. I. Krylov, J. Comput. Chem. **34**, 2293 (2013).
[35] K. Eichkorn, O. Treutler, H. Öhm, M. Häser, and R. Ahlrichs, Chem. Phys. Lett. **242**, 652 (1995).
[36] K. Eichkorn, F. Weigend, O. Treutler, and R. Ahlrichs, Theor. Chem. Acc. **97**, 119 (1997).
[37] F. Weigend, J. Comput. Chem. **29**, 167 (2008).
[38] F. Weigend, A. Köhn, and C. Hättig, J. Chem. Phys. **116**, 3175 (2002).
[39] F. Weigend, Phys. Chem. Chem. Phys. **8**, 1057 (2006).
[40] A. Hellweg, C. Hättig, S. Höfener, and W. Klopper, Theor. Chem. Acc. **117**, 587 (2007).
[41] G. D. Purvis and R. J. Bartlett, J. Chem. Phys. **76**, 1910 (1982).
[42] Throughout the paper, we adhere to the convention that *ijkl* denote occupied orbitals, *abcd* denote virtual orbitals, and *pqrs* denote orbitals that can be either occupied or virtual.
[43] In the closed-shell case, two spin cases of integrals and amplitudes are stored: $\langle\alpha\alpha||\alpha\alpha\rangle$ and $\langle\alpha\beta||\alpha\beta\rangle$; in the open-shell case there is additionally a third spin case: $\langle\beta\beta||\beta\beta\rangle$. With applicable permutational symmetry taken into consideration and an assumption of no point group symmetry in the molecule, the disk requirement for ERI objects in the closed-shell case are $\langle ij||kl\rangle : \frac{3}{8}O^4$, $\langle ij||ka\rangle : \frac{3}{2}O^3V$, $\langle ij||ab\rangle : \frac{3}{4}O^2V^2$, $\langle ia||jb\rangle : O^2V^2$, $\langle ia||bc\rangle : \frac{3}{2}OV^3$, and $\langle ab||cd\rangle : \frac{3}{8}V^4$.
[44] Usually, several T_2 vectors are stored for the DIIS algorithm.
[45] S. V. Levchenko and A. I. Krylov, J. Chem. Phys. **120**, 175 (2004).
[46] The three-tensor contraction is defined as $T_{ijk} = \sum_{abc} A_{iac} B_{jab} C_{kbc}$, where i, j, k, a, b, c designate collectively respective inner and outer indices in tensors A, B, C. To compute such a contraction, first an intermediate needs to be formed by multiplying two of the arguments, followed by a contraction with the third argument to form the result. If the size of the intermediate exceeds memory limits, it is desirable to proceed in batches by forming the intermediate by parts and contracting such that each batch is completed in core memory. This approach allows us to minimize I/O by automatically tuning the batch size to the memory limit and retain the total computational cost of the contraction.
[47] Storage requirements for the RI integrals in the closed-shell case are as follows: $B^P_{oo} : \frac{1}{2}O^2M$, $B^P_{ov} : OVM$, and $B^P_{vv} : \frac{1}{2}V^2M$.
[48] J. F. Stanton and R. J. Bartlett, J. Chem. Phys. **98**, 7029 (1993).
[49] A. I. Krylov, Acc. Chem. Res. **39**, 83 (2006).
[50] J. Boström, F. Aquilante, T. B. Pedersen, and R. Lindh, J. Chem. Theory Comput. **5**, 1545 (2009).
[51] V. P. Vysotskiy and L. S. Cederbaum, J. Chem. Phys. **132**, 044110 (2010).
[52] C. Hättig and F. Weigend, J. Chem. Phys. **113**, 5154 (2000).
[53] A. Köhn and C. Hättig, J. Chem. Phys. **119**, 5021 (2003).
[54] C. Hättig and A. Köhn, J. Chem. Phys. **117**, 6939 (2002).
[55] During the final revision stage, a small algorithmic improvement has been implemented that resulted in ∼5% speed-up in CD/RI CC/EOM calculations. Thus, the reported timings are roughly 5% slower than the current code.
[56] T. Rocha-Rinza, O. Christiansen, J. Rajput, A. Gopalan, D. B. Rahbek, L. H. Andersen, A. V. Bochenkova, A. A. Granovsky, K. B. Bravaya, A. V. Nemukhin, K. L. Christiansen, and M. B. Nielsen, J. Phys. Chem. A **113**, 9442 (2009).
[57] D. Zuev, K. B. Bravaya, T. D. Crawford, R. Lindh, and A. I. Krylov, J. Chem. Phys. **134**, 034310 (2011).
[58] K. B. Bravaya, E. Epifanovsky, and Anna I. Krylov, J. Phys. Chem. Lett. **3**, 2726 (2012).
[59] Nwchem: High-Performance Computational Chemistry Software, see http://www.nwchem-sw.org/index.php/Benchmarks (last accessed March 13, 2013).
[60] K. Khistyaev, A. Golan, K. B. Bravaya, N. Orms, A. I. Krylov, and M. Ahmed, J. Phys. Chem. A **117**, 6789 (2013).
[61] Q-Chem's keywords controlling the CC and EOM convergence: CC_T_CONV = 4, CC_E_CONV = 6, EOM_DAVIDSON_CONV = 5, EOM_DAVIDSON_THRESH = 5.

[62] A. Landau, K. Khistyaev, S. Dolgikh, and A. I. Krylov, J. Chem. Phys. **132**, 014109 (2010).
[63] See supplementary material at http://dx.doi.org/10.1063/1.4820484 for additional details on timings, Cartesian geometries, and relevant energies.
[64] W. J. Hehre, R. Ditchfield, and J. A. Pople, J. Chem. Phys. **56**, 2257 (1972).
[65] T. Clark, J. Chandrasekhar, and P. V. R. Schleyer, J. Comput. Chem. **4**, 294 (1983).
[66] R. Krishnan, J. S. Binkley, R. Seeger, and J. A. Pople, J. Chem. Phys. **72**, 650 (1980).
[67] T. H. Dunning, J. Chem. Phys. **90**, 1007 (1989).
[68] F. Weigend and M. Häser, Theor. Chim. Acta **97**, 331 (1997).
[69] J. Yang, Y. Kurashige, F. R. Manby, and G. K. L. Chan, J. Chem. Phys. **134**, 044123 (2011).
[70] J. E. Subotnik and M. Head-Gordon, J. Chem. Phys. **123**, 064108 (2005).
[71] U. Benedikt, A. A. Auer, M. Espig, and W. Hackbusch, J. Chem. Phys. **134**, 054118 (2011).
[72] F. Bell, D. Lambrecht, and M. Head-Gordon, Mol. Phys. **108**, 2759 (2010).