JCTC Journal of Chemical Theory and Computation

Article

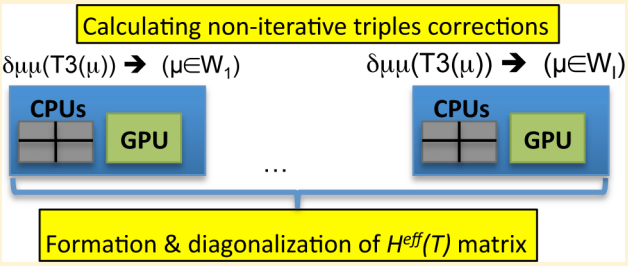
pubs.acs.org/JCTC

Noniterative Multireference Coupled Cluster Methods on Heterogeneous CPU−GPU Systems

Kiran Bhaskaran-Nair, Wenjing Ma, Sriram Krishnamoorthy, Oreste Villa, Hubertus J. J. van Dam, Edoardo Aprà, and Karol Kowalski*

William R. Wiley Environmental Molecular Sciences Laboratory, Battelle, Pacific Northwest National Laboratory, K8-91, P.O. Box 999, Richland, Washington 99352, United States

ABSTRACT: A novel parallel algorithm for noniterative multireference coupled cluster (MRCC) theories, which merges recently introduced reference-level parallelism (RLP) [Bhaskaran-Nair, K.; Brabec, J.; Aprà, E.; van Dam, H. J. J.; Pittner, J.; Kowalski, K. *J. Chem. Phys.* **2012**, *137*, 094112] with the possibility of accelerating numerical calculations using graphics processing units (GPUs) is presented. We discuss the performance of this approach applied to the MRCCSD(T) method (iterative singles and doubles and perturbative triples), where the corrections due to triples are added to the diagonal elements of the MRCCSD effective Hamiltonian matrix. The performance of the combined RLP/GPU algorithm is illustrated on the example of the Brillouin−Wigner (BW) and Mukherjee (Mk) state-specific MRCCSD(T) formulations.



1. INTRODUCTION

The widespread use of accurate yet very expensive multireference coupled cluster methods (MRCC)[1,2] is contingent upon the availability of codes that can take advantage of emerging heterogeneous computer architectures. Of special interest is the utilization of graphics processing units (GPUs), which has been shown to significantly enhance the performance of several electronic structure methods.[3−21] Due to the complex structure of the algebraic equations and an intensive communication pattern, the use of GPU-based architectures for CC calculations is very challenging.[22,23] Recently, the GPU implementations of the noniterative CCSD(T) methods have been successfully tested on selected closed- and open-shell molecular systems.[23] It was demonstrated that the performance of the GPU implementation of the CCSD(T) method (characterized by N^7 scaling, where N symbolically designates system size) is critically dependent on the parameters which define the block structure of all tensors used in the CC calculations. It was demonstrated that the proper choice of these parameters leads to nearly 9-fold speedup in double precision calculations, which is in line with the speedups reported in other GPU implementations based on double precision arithmetic.[8]

In this paper, we report on the GPU implementation of the noniterative approaches accounting for the triple excitations in correcting the energies of the MRCC models with singles and doubles (MRCCSD) in NWChem.[24] These methods will be generally termed as MRCCSD(T) methods. For its simplicity and formal similarity to the single reference CCSD(T) correction, we will entirely focus on the class of triples corrections to the effective Hamiltonian matrix introduced by Balková and Bartlett[25] (for similar developments, see refs 26−30). This approach can be universally applied to any type of Hilbert space MRCC formulation based on the Jeziorski−Monkhorst MRCC ansatz for the electronic wave function.[31] Due to the inherent numerical complexity, only diagonal triples corrections to the effective Hamiltonian matrix are considered in this paper. Other strategies for correcting MRCCSD energies are amply discussed in the literature, including multireference extensions of the Method of Moments of Coupled Cluster equations,[32,33] Λ-MRCCSD(T) method,[34] and USS approaches.[35,36] We illustrate the performance of the MRCCSD(T) corrections in the context of two state-selective methods utilizing the Jeziorski−Monkhorst ansatz: the Brillouin−Wigner (BW-MRCCSD)[37−41] and Mukherjee's MRCCSD (Mk-MRCCSD)[28,42−56] formulations. The state-specific methods have been introduced to address the intruder-state problem which was the major stumbling block for widespread use of state-universal MRCC methods based on the concept of the complete model space. Additionally, the state-specific formulations offer more optimization parameters (cluster amplitudes) per state compared to state-universal MRCC methods.

This paper is organized as follows: In section 2, we briefly discuss the theoretical details of the MRCCSD(T) formalism, while in section 3 we give a detailed description of our parallel GPU implementation. In section 4, we apply our formalism for calculating the doubly excited singlet state of the pentacene molecule.

Received: December 27, 2012

   dx.doi.org/10.1021/ct301130u | *J. Chem. Theory Comput.* XXXX, XXX, XXX−XXX

2. THEORY

2.1. Iterative State Selective MRCCSD Approaches. In contrast to the reference function $|\Phi\rangle$ used in the single reference CC methods, the multireference methods use the concept of a multidimensional model space $\mathcal{M}_0$, spanned by M Slater determinants $\{|\Phi_\mu\rangle\}_{\mu=1}^{M}$, which can provide a zeroth-order approximation to the quasidegenerate wave function $|\Psi\rangle$, i.e.,

$$|\Psi\rangle \simeq |\tilde{\Psi}\rangle = \sum_{\mu=1}^{M} c_\mu |\Phi_\mu\rangle \tag{1}$$

In the Jeziorski–Monkhorst (JM) ansatz,[31] the electronic wave function is parametrized as

$$|\Psi\rangle = \sum_{\mu=1}^{M} c_\mu e^{T^{(\mu)}} |\Phi_\mu\rangle \tag{2}$$

where the $T^{(\mu)}$ are the reference-specific cluster operators. For the complete model space (CMS), which is defined by determinants obtained from all possible distributions of the active electrons among the active spin–orbitals, the intermediate normalization condition (see ref 31) requires that the cluster operators do not produce any excitations within the model space, i.e.,

$$\langle \Phi_\nu | T^{(\mu)} | \Phi_\mu \rangle = 0, \forall_{\mu,\nu} \tag{3}$$

In the MRCC with singles and doubles (MRCCSD) approximation, the cluster operators are represented as a sum of singly ($T_1^{(\mu)}$) and doubly ($T_2^{(\mu)}$) excited clusters

$$T^{(\mu)} \simeq T_1^{(\mu)} + T_2^{(\mu)} \tag{4}$$

where each component of $T^{(\mu)}$ is defined by the so-called cluster amplitudes $t_{a(\mu)\ldots}^{i(\mu)\ldots}(\mu)$

$$T_1^{(\mu)} = \sum_{i(\mu),a(\mu)} t_{a(\mu)}^{i(\mu)}(\mu)\, X_{a(\mu)}^{+} X_{i(\mu)} \tag{5}$$

$$T_2^{(\mu)} = \sum_{i(\mu)<j(\mu),a(\mu)<b(\mu)} t_{a(\mu)b(\mu)}^{i(\mu)j(\mu)}(\mu)\, X_{a(\mu)}^{+} X_{b(\mu)}^{+} X_{j(\mu)} X_{i(\mu)} \tag{6}$$

In the above summations, $i(\mu)$,$j(\mu)$ ($a(\mu)$,$b(\mu)$) indices correspond to occupied (unoccupied) spin–orbitals in reference $|\Phi_\mu\rangle$. As a consequence of the condition in eq 3, the summations in eqs 5 and 6 exclude the case when all indices correspond to active spin–orbitals. In the state-selective methods, the working equations for the cluster amplitudes (or the sufficiency conditions) are obtained by substituting the JM representation of wave function 2 into the Schrödinger equation

$$H \sum_{\mu}^{M} c_\mu e^{T^{(\mu)}} |\Phi_\mu\rangle = E \sum_{\mu=1}^{M} c_\mu e^{T^{(\mu)}} |\Phi_\mu\rangle \tag{7}$$

Due to the known overcompleteness problem of the state-selective approaches, several types of sufficiency conditions have been discussed in the literature.[37−39,42,43,45] The BW-MRCCSD and Mk-MRCCSD sufficiency conditions take the following form:

$$(E - H_{\mu\mu}^{\mathrm{eff}}) \langle \Phi_\theta^{(\mu)} | e^{T^{(\mu)}} | \Phi_\mu \rangle - \langle \Phi_\theta^{(\mu)} | H_N(\mu) e^{T^{(\mu)}} | \Phi_\mu \rangle_{\mathrm{C+DC,L}} = 0, \forall_\mu \quad \text{(BW-MRCCSD)} \tag{8}$$

$$\langle \Phi_\theta^{(\mu)} | (H e^{T^{(\mu)}})_{\mathrm{C}} | \Phi_\mu \rangle c_\mu + \sum_{\nu\neq\mu} \langle \Phi_\theta^{(\mu)} | e^{-T^{(\mu)}} e^{T^{(\nu)}} | \Phi_\mu \rangle H_{\mu\nu}^{\mathrm{eff}} c_\nu = 0, \forall_\mu \quad \text{(Mk-MRCCSD)} \tag{9}$$

where the cluster operators are given by eq 4, and $\langle \Phi_\theta^{(\mu)}|$ are excited configurations corresponding to the excitations used to define cluster operators $T^{(\mu)}$. The subscript C + DC,L designates all connected diagrams and all linked but disconnected diagrams. In contrast to the Mk-MRCC approach, the BW-MRCC formalism contains disconnected contributions. The diagonalization of the effective Hamiltonian matrix, defined for the CMS by the matrix elements $H_{\nu\mu}^{\mathrm{eff}}$,

$$H_{\nu\mu}^{\mathrm{eff}} = \langle \Phi_\nu | (H e^{T^{(\mu)}})_{\mathrm{C}} | \Phi_\mu \rangle \tag{10}$$

leads to energies (eigenvalues) and c_μ coefficients (components of the corresponding eigenvectors). The equations for the cluster amplitudes can be cast in the following algebraic form

$$\mathbf{R}^{(\mu)} = \mathbf{F}^{(\mu)}(T^{(\mu)}) + \mathbf{G}^{(\mu)}(T^{(1)}, \ldots, T^{(\mu)}, \ldots, T^{(M)}) = \mathbf{0}, \forall_{\mu=1,\ldots,M} \tag{11}$$

where the $\mathbf{F}^{(\mu)}(T^{(\mu)})$ part represents the so-called direct terms ($\langle \Phi_\theta^{(\mu)} | (H e^{T_{(\mu)}})_{\mathrm{C}} | \Phi_\mu \rangle$) and the $\mathbf{G}^{(\mu)}(T^{(1)},\ldots,T^{(\mu)},\ldots,T^{(M)})$ parts represent the so-called renormalization terms. While the direct terms depend only on the cluster operator corresponding to a given reference, the coupling term may involve all possible cluster operators (this can be directly seen in the Mk-MRCCSD approach; in the BW-MRCCSD this dependence is implicit through the energy obtained from the diagonalization of the effective Hamiltonian). The algebraic form of the coupling terms in the Mk-MRCCSD approach has been derived by Evangelista et al. in ref 45.

2.2. Noniterative Corrections Due to Triples to the MRCCSD Energies. The inclusion of higher-order clusters is a natural way of increasing the accuracy of the MRCC methods in the presence of intruder states. It was demonstrated that the inclusion of higher-rank excitations in the cluster operators (triples) offers further refinement of the MRCCSD method in calculating spectroscopic constants and the heights of the reaction barriers. The noniterative triples contribution was also shown to alleviate problems associated with the lack of unitary invariance of the MRCC methods with respect to rotations involving active orbitals. Among several venues to correct the MRCCSD energies (for a review, see ref 30 and references therein), the conceptually simplest approach consists of adding diagrams to the diagonal elements of the MRCCSD effective Hamiltonian matrix (H^{eff}(CCSD)) in a way analogous to that of the single reference CCSD(T) approach, i.e.,

$$H_{\nu\mu}^{\mathrm{eff}}(T) = H_{\nu\mu}^{\mathrm{eff}}(\mathrm{CCSD}) + \delta_{\nu\mu}\delta_{\mu\mu}(T_3^{(\mu)}) \tag{12}$$

$$\delta_{\mu\mu}(T_3^{(\mu)}) = E_{\mathrm{T}}^{[4]}(\mu) + E_{\mathrm{ST}}^{[5]}(\mu) + E_{\mathrm{ST}}^{[4]}(\mu) \tag{13}$$

where $\delta_{\nu\mu}$ represents the Kronecker δ function. The $E_{\mathrm{T}}^{[4]}(\mu)$, $E_{\mathrm{ST}}^{[5]}(\mu)$, and $E_{\mathrm{ST}}^{[4]}(\mu)$ terms represent fourth and fifth order contributions, given by the expressions

$$E_{\mathrm{T}}^{[4]}(\mu) = \frac{1}{36} \sum_{abcijk} \langle (\Phi_\mu)_{ijk}^{abc} | V_{\mathrm{N}}(\mu) T_2^{(\mu)} | \Phi_\mu \rangle_{\mathrm{C}} t_{ijk}^{abc}(\mu) \tag{14}$$

$$E_{\mathrm{ST}}^{[5]}(\mu) = \sum_{ai} s_i^a(\mu) t_i^a(\mu) \tag{15}$$

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

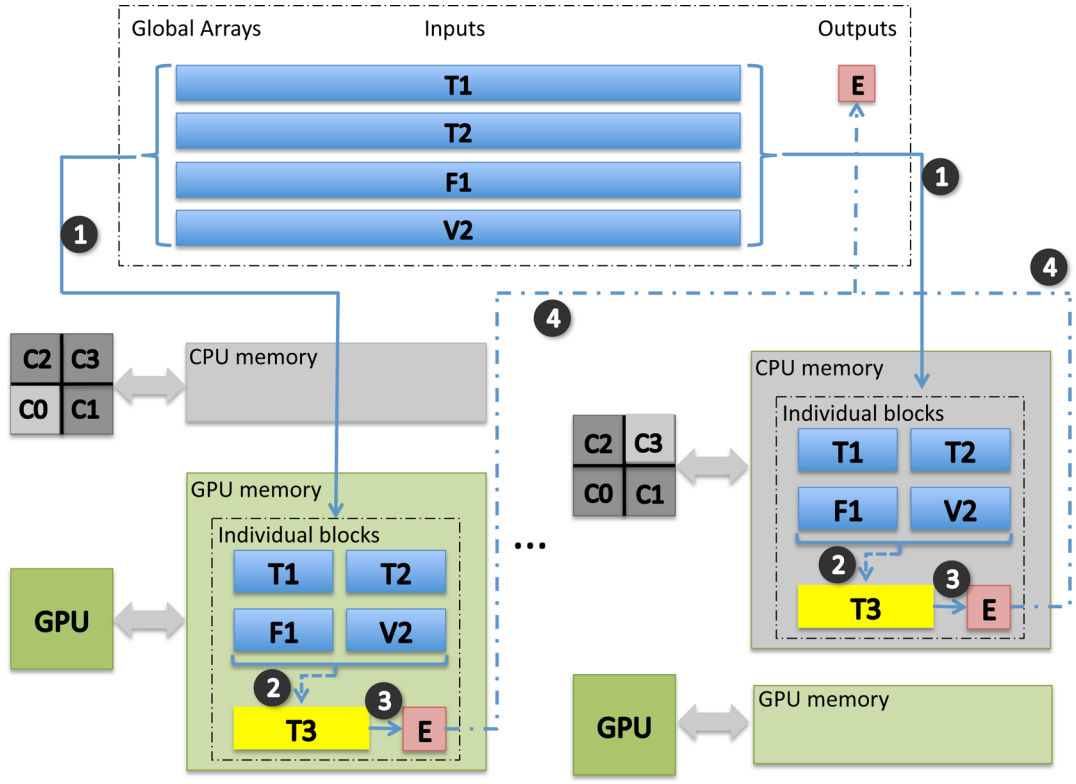


Figure 1. Execution steps involved in the hybrid (CPU and GPU) implementation of the noniterative triples correction. T_1, T_2 and T_3 tensors corresponds to reference-specific amplitudes $T_1(\mu)$, $T_2(\mu)$, and $T_3(\mu)$, respectively (see eqs 5, 6, and 18). The F_1 and V_2 tensors corresponds to one- and two-electron integrals, respectively. The individual steps are as follows: step 1, copy input blocks from global arrays to CPU/GPU local memory; step 2, contract input blocks into intermediate tensor block; step 3, reduce intermediate tensor to compute energy correction contribution; step 4, reduce final energy correction across all CPUs and GPUs.

$$E_{\mathrm{ST}}^{[4]}(\mu) = \frac{1}{4}\sum_{abcijk} f_{kc}(\mu)t_{ij}^{ab}(\mu)t_{ijk}^{abc}(\mu) \tag{16}$$

where the $s_i^a(\mu)$ intermediate is defined as

$$s_i^a(\mu) = \frac{1}{4}\sum_{bcjk} \langle bc||jk\rangle t_{ijk}^{abc}(\mu) \tag{17}$$

These corrections can be added to both BW-MRCCSD and Mk-MRCCSD effective Hamiltonians. Although several forms of the triply excited cluster amplitudes $t_{ijk}^{abc}(\mu)$ have been extensively discussed in the literature,[27−30,34] in this paper we will resort to the simplest choice defined by the formula

$$t_{ijk}^{abc}(\mu) = \frac{\langle (\Phi_\mu)_{ijk}^{abc}|V_N(\mu)T_2^{(\mu)}|\Phi_\mu\rangle_C}{D_{ijk}^{abc}(\mu)} \tag{18}$$

which is analogous to the form of the T_3 operator utilized by the CCSD(T) method. In our implementation, the $\mathbf{H}^{\mathrm{eff}}$(CCSD) operator includes up to two-body terms. Even though the noniterative corrections are added only to the diagonal elements of the MRCCSD effective Hamiltonian, the total cost of this procedure scales as $M \times N^7$, which poses a significant computational challenge. Recently, we have demonstrated on the example of a β-carotene molecule that the parallel implementations of the MRCCSD(T) methods can take advantage of 80 000 cores.[30] When integrated with parallel codes, the utilization of the GPU technology can provide a further reduction of the time to solution corresponding to noniterative methods.

3. NUMERICAL IMPLEMENTATION

3.1. Parallel Implementation. In this section, we describe our hybrid (CPU and GPU) implementation of the non-iterative triples correction. We use the terms CPU and core interchangeably to refer to a sequential processing unit. The execution steps involved at runtime are depicted in Figure 1. At the end of the iterative calculation, the T2 (reference-specific $T_2(\mu)$ amplitudes) and V2 (reference-specific two-electron integrals) tensors are stored in the distributed memories of the processors in the form of linearized global arrays.[57] Global Arrays is a library used in NWChem that aids the design of parallel applications through efficient support for global data structures as multidimensional arrays. Each global array is distributed among the local memories of the processors (CPUs). Each processor can request to get, put, or update a portion of a global array without interacting with other processors. The partitioned nature of the distribution allows each processor to directly access the portion of global array stored in its local memory. For simplicity, we depict the global arrays as being in a distinct memory domain in Figure 1.

The noniterative calculation consists of the singles and doubles contributions, followed by reduction of the resulting N^6 tensor into a scalar energy correction. The production of intermediate tensors is an N^7 operation, while the computation of the energy correction costs N^6 operations. The input tensors are initially stored in global arrays, and the output in the form of the energy scalar is produced in a global variable (single-element global array). The input tensors are logically partitioned into dense rectangular blocks. The calculations are organized as a sequence of block−block contractions, each of

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

which operates on individual *blocks* of the inputs to produce a contribution to a block of the output. All block−block contractions that together produce one block of the intermediate tensor form a *task*—the basic unit of work that is scheduled between the CPUs and GPUs. The intermediate N^6 array is the largest tensor in the MRCCSD(T) calculation. Rather than compute it in its entirety before reducing it to an energy correction scalar, we optimize its memory utilization using the following technique. Given the nature of the noniterative calculation, the intermediate tensor is computed only to determine the energy correction. We observe that the contribution of the intermediate tensor to the energy correction is done in a per element fashion. This allows us to combine the production of an individual block of the intermediate tensor and the calculation of its contribution to the energy correction. This way, the intermediate tensor is computed one block at a time, allowing the storage requirement to be greatly reduced. In particular, we store one block of the intermediate N^6 tensor per processor at any point during the calculation. Given that all tensors are stored as blocks with identical block sizes along each dimension, reducing the storage requirement allows us to increase the block size. This also improves the performance achieved for each such computation as shown in prior work[23,58,59] and confirmed in the experimental evaluation.

When a task is scheduled to be executed on a CPU or a GPU, the input data blocks are copied from the global arrays to local memory, the calculation performed locally, and the result copied back to the global array. The approach described above enables us to completely eliminate communication for the intermediate tensor T3—it is produced and reduced to an energy correction scalar in local memory without any communication. The energy corrections in the local memories of all processors are then combined to form the final energy correction. Given that the GPU cannot directly initiate communication, we employ one CPU core to drive the GPUs in a node. The CPU core driving the GPU is responsible for transferring the data blocks from the global arrays to GPU memory and launching the kernels that perform the corresponding computation on the transferred blocks. While the GPUs (together with its driving CPU) are performing the necessary calculations, other CPUs in the system also contribute to the calculation by transferring individual blocks of data from the global arrays to their local memories and performing sequential contraction calculations. Thus, all CPUs and GPUs contribute to the noniterative corrections providing the maximum possible performance. The GPUs can perform their calculations at a much greater rate than the sequential CPUs due to greater hardware resources being available. Rather than attempt to statically balance the work among the CPUs and GPUs, we employ a dynamic load balancing scheme. The work to be performed is implicitly ordered into a list of tasks that are indexed using an atomic counter. Each processor that completed the work currently assigned to it requests the next value of the atomic counter to index into this task list and identify the work to be executed. The GPU operates on the atomic counter through the driving CPU. This scheme allows individual processing elements to execute at different rates while all processing elements are being utilized without waiting for one another. The execution completes when the task list has been fully traversed.

The execution of contractions within the GPU are organized as a collection of CUDA kernels. The noniterative calculation consists of one singles and two doubles contractions, each of which includes nine-way symmetrization. We implemented one kernel for each loop index order encountered in each of these contractions, resulting in a total of 27 CUDA kernels. These were automatically generated with the optimizations that were developed earlier (ref 23 contains the details of the optimizations). The energy calculation is also implemented as a CUDA kernel. The driver CPU is responsible for invoking the relevant CUDA kernels to perform the necessary work. In particular, it detects the loop index order for the task at hand and invokes the corresponding kernel.

3.2. Reference-Level Parallelism. The reference-level parallelism (RLP), which has been described in ref 60, refers to coarse grained parallelism induced either by the structure of the MRCCSD equations or MRCCSD(T) corrections which are either an aggregate of reference-specific equations or form the set of reference-specific noniterative corrections. The key idea behind RLP is to distribute the formation of the reference-specific MRCC equations/corrections over various processor groups (PGs). By processor groups (G_i), we mean a partitioning of the whole processor domain (D) into smaller pieces, which can be symbolically expressed

$$D = \bigcup_{i=1,\ldots,I} G_i \tag{19}$$

where I is the total number of the processor groups. We will assume that the number of processors in each group (S_i) is the same and is equal to S, i.e.,

$$S_i = S = \frac{N_p}{I} (i = 1, \ldots, I) \tag{20}$$

In the above equation N_p stands for the total number of processors, which is a multiple of PGs number I. This approach employs two-level parallelism: (1) reference-level parallelism, where each set of reference-specific equations (or their aggregate) or corrections is calculated on a separate PG and (2) task-level parallelism, used to calculate a given set of reference-specific equations or a given noniterative correction.

In the simplest case, the work organization chart (symbolically designated by W) corresponds to the situation when a single PG is delegated to calculate a single set of reference-specific equations/corrections. In this case, the number of PGs coincides with the size of the model space (i.e., $I = M$). A more general situation corresponds to the case when each PG G_i forms several ($n_r(i)$) residual vectors of the MRCCSD equations or several reference-specific corrections. This can be symbolically denoted as

$$W = \bigcup_{i=1,\ldots,I} W_i(n_r(i)) \tag{21}$$

where W_i refers to the workload on the corresponding processor group G_i. In order to provide optimal load balance between the workloads on each PG, it is natural to assume that

$$n_r = n_r(i) = \frac{M}{I} (i = 1, \ldots, I) \tag{22}$$

In the following, in order to report specific configurations of the processor groups used in the calculation, we will use the PG(n_r,I,S) notation.

In the BW-MRCCSD or Mk-MRCCSD methods, where sufficiency conditions are numerically dominated by the direct terms which utilize cluster amplitudes corresponding to a given reference only, the use of the PGs enables one to significantly reduce the communication to within a processor group. Similar

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

ideas have been pursued in calculating diagonal noniterative corrections due to triples.[30] In both cases, the recent scalability tests clearly demonstrated high efficiency of the algorithms based on the processor groups. The pictorial representation of the RLP algorithm used in the context of MRCCSD(T) theories is shown in Figure 2.

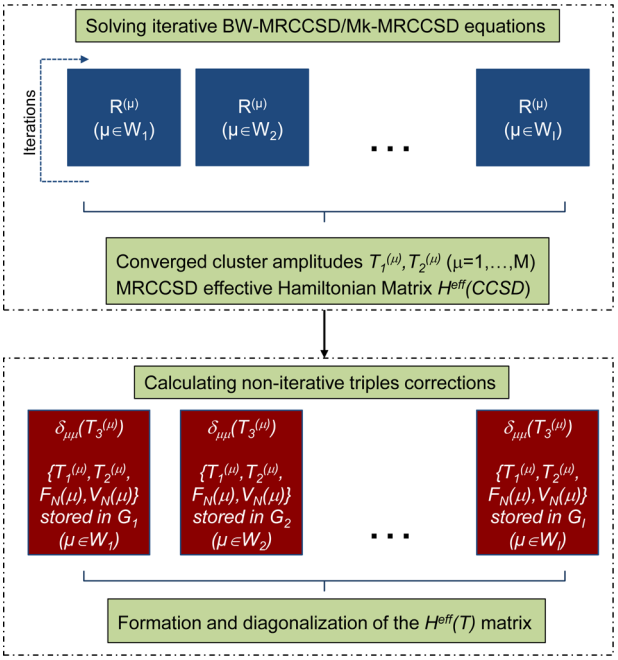


Figure 2. Graphical representation of reference level parallelism. Separate processor groups are delegated to calculate the reference-specific part of the MRCCSD equations and reference-specific noniterative corrections $\delta_{\mu\mu}(T_3^{(\mu)})$ (see eq 13).

3.3. Experimental Evaluation. We ran our experiments on a cluster consisting of dual-socket nodes constructed from 16-core AMD Interlagos processors, 64 GB of memory, and one Tesla M2090 (Fermi) GPU. The nodes are connected using a high-performance QDR Infiniband interconnect.

We evaluated the performance of our implementation on an example of the dodecane molecule. We considered several CPU−GPU configurations to evaluate the relative benefits of using the CPU vs the GPU. Memory requirements limit the number of CPUs we can use on each node to eight. Therefore, we evaluated the performance of computing MRCCSD(T) when using two, four, and eight cores per node. For the hybrid CPU−GPU execution, one of the cores drives the GPU rather than performing calculations. All experiments were performed on up to 24 nodes, the largest number of nodes that contain the Tesla M2090 GPUs in the cluster. While the implementation allows us to exploit multiple GPUs per node, this cluster configuration does not enable such an evaluation, limiting our tests to one GPU per node. We expect our implementation to scale reasonably well on multi-GPU configurations given the reduction in communication traffic resulting from our design as described in section 3.1.

We first evaluate the effectiveness of the load balancing in dynamically distributing work between the CPU and the GPU for various execution scenarios. We tracked the total number of tasks executed on the CPUs and the GPUs as we varied the node configuration and the block size. The results are shown in Table 1. The number of tasks presented were averaged over the four references that were executed. Even though each reference is independently load balanced, we observed very little variation in the CPU−GPU task distribution between the execution of the different references. The second column in the table depicts the total number of tasks—the number of blocks of the intermediate T3 tensor—for a given input size. As the block size is increased, the number of tasks decreases. However, the relationship is not always proportional. For example, we observe that the total number of tasks remains the same as we increase the block size from 18 to 20. This is due to the fact that the actual size of a block is influenced by several factors. NWChem restricts a block from spanning multiple spatial blocks, thus limiting the size of the blocks. In addition, NWChem attempts to balance the size of blocks within each spatial block so as to better balance the computational load in handling each block. We therefore anticipate the computational characteristics of executions with different block sizes to be distinct even if the number of tasks does not vary significantly.

For any given configuration, the GPU executes far more tasks than the CPUs. The GPU, being faster than the sequential CPU, completes execution of each task faster and thus requests more tasks by using the global counter more frequently. For a given block size, we observe that CPUs execute an increasing fraction of the tasks as we utilize more CPUs in the system. When utilizing one CPU and one GPU, the GPU executes over an order of magnitude more tasks than those executed by the CPU. This translates to a speedup of close to an order of magnitude reported in the literature when performing a calculation on the GPU as compared to the sequential CPU. In most cases, the number of tasks executed by the CPU goes up by a factor of up to 5 as we increase the number of CPUs executing tasks from 1 to 7. We observed that larger block sizes benefit execution on the GPU by achieving greater improvements in performance, as compared to the relative improve-

Table 1. Average of Tasks Executed Per Reference on the CPUs and the GPU for Each Node Configuration and Block Size

		node configuration					
		1CPU+1GPU		3CPUs+1GPU		7CPUs+1GPU	
		CPU	GPU	CPU	GPU	CPU	GPU
block size	total tasks	tasks	tasks	tasks	tasks	tasks	tasks
12	191756	15097	176659	36437	155319	65307	126449
14	129846	8934	120912	21977	107869	39987	89859
16	104688	6719	97970	16541	88148	30708	73980
18	15948	919	15029	2301	13647	4510	11438
20	15948	944	15004	2367	13581	4554	11394
22	8956	515	8441	1220	7736	2382	6574

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

ments on the CPU. In other words, the speedup achieved on the GPU as compared to sequential CPU execution often increases with block size. For a given node configuration, this effect often results in a larger fraction of tasks being executed by the GPU as we increase the block size.

Figure 3 shows the speedup achieved by the various configurations with respect to execution times when run on

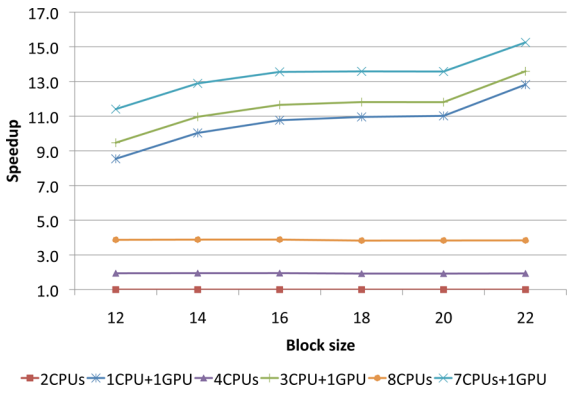


Figure 3. Speedup of the noniterative calculation for various configurations for block sizes varying from 12 through 22 with respect to the execution times using two CPUs and no GPUs.

two CPU cores and no GPUs. We expect that increasing the number of CPUs utilized increases the speedup relative to serial execution. In addition deploying GPUs speeds the calculations up further. Likewise, increasing the block sizes with a given CPU−GPU node configuration speeds the calculation up also. We see that doubling the number of CPU cores expectedly results in close to double the performance. This speedup is also independent of block sizes, demonstrating that the sequential CPU execution achieves the same floating point performance independent of block size. Unlike execution on the CPU cores, the speedups achieved when using GPUs increases with block sizes. In particular, when employing one CPU and one GPU, increasing the block size from 12 to 22 provides us a speedup factor of 1.5 without any additional hardware resources. This is due to improved floating point performance achieved by the GPU when using larger block sizes. Using one CPU core and one GPU instead of two CPU cores results in a performance improvement factor of 8.5 when using a block size of 12. Relating the execution time on two CPUs with that of one CPU core and one GPU, we can determine that a single GPU achieves a speedup of 16 over a single CPU core. Given this factor, we would predict, in the absence of superlinear effects, a speedup of 11.5 when employing seven CPU cores and one GPU. We, in fact, achieve a speedup of 11.4. Similarly, with a block size of 22, we determine the speedup achieved by a single GPU as compared to a single CPU core to be 24.6. When using seven CPU cores and one GPU, this should result in a speedup of 15.8. We observe a speedup of 15.3. This close match between the anticipated and observed speedups shows that additional cores are effectively utilized as they are added to the execution.

4. APPLICATION OF THE MRCCSD(T) FORMALISM: EXCITED STATES OF PENTACENE

The pentacene molecule has received much attention because of its high charge-carrier (hole) mobility in films and molecular crystals and its role as a component of light harvesting systems.[61−63] In the context of solar energy conversion, particularly interesting are the studies of singlet fission (SF) processes in pentacene assemblies, where a singlet excited pentacene interacts with the neighboring pentacene molecules to produce a pair of triplet excited states.[64−66] In this process the energy of an absorbed photon is transformed into the formation of multiple electron−hole pairs. Theoretical studies of SF processes are facing several challenges mainly associated with the presence of multielectron excitations in the relevant singlet excited states. By their nature, these states are multiconfigurational and require appropriate parametrization of the excited-state wave function. For example, in the recent calculations performed for the pentacene dimer using multireference many-body perturbation theory (MRMBPT) the role of doubly excited singlet states as an initial state for SF process has been confirmed.[66]

We believe that the MRCC methodologies discussed in previous sections may be useful in properly describing these states. Due to the numerical overhead of the MRCCSD(T) approaches, in this paper we focus on the lowest-lying doubly excited state (D state) of the pentacene molecule described by relatively small basis sets (6-31G[67] and cc-pVDZ[68]). Our main goal is not to provide high precision estimates of the vertical excitation energy (VEE) corresponding to the D state but rather to discuss the relations between VEEs for the D state obtained with the MRCC and EOMCC methods. In ref 69, we focused on the EOMCC calculations for the singly excited 1^1B_{2u} and 1^1B_{3u} states, which demonstrated the importance of the three-body correlation effects in single reference EOMCC theories. The present analysis is a natural extension of these studies.

As suggested in ref 65, the most important part of the dynamic correlation effects for the D state can be described by double excitation from HOMO to LUMO, HOMO−1 to LUMO, and HOMO to LUMO+1 orbitals. For this reason, we used in our MRCC calculations (4,4) CMS containing these Slater determinants. Although, MRMBPT calculations reported in ref 65 used large (12,12) CMS. However, the iterative multireference approaches can provide highly accurate estimates of energy or energy differences using much smaller model spaces compared to perturbative MR formulations (see for example ref 70). We re-examined the correctness of the (4,4) CMS by performing a series of calculations using EOMCC with a singles and doubles (EOMCCSD)[71−73] approach and two types of completely renormalized EOMCCSD with perturbative triples methods (CR-EOMCCSD(T) approach of ref 74 and its reduced variant CR-EOMCCSD(T)$^{(II)}$ (r-CR-EOMCCSD(T)$^{(II)}$) of ref 75). Moreover, in contrast to the 6-31G basis set, the EOMCCSD calculations employing the cc-pVDZ basis set indicate that the D state wave function in addition to the leading configurations from the (4,4) CMS contains singly excited configurations not included in the (4,4) CMS, which corresponds to the $2b_{1u} \rightarrow 4b_{1u}$ and $2b_{1u} \rightarrow 4b_{1u}$ excitations, which are about 2.5−3 times smaller (in the absolute values) than the amplitude corresponding to the leading double excitation from HOMO ($3b_{2g}$) to LUMO ($4b_{1u}$).

The numerical complexity of the EOMCCSD and CR-EOMCCSD(T) methods is proportional to N^6 and N^7 (N designates system size) per root, respectively. This should be compared with the $M \times N^6$ scaling of the BW/Mk-MRCCSD approaches and $M \times N^7$ scaling of the BW/Mk-MRCCSD(T) methods. The MRCC methods are more expensive than the

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

corresponding second order MRMBPT (MRMBPT(2)), which scales as $M \times N^4$. This scaling does not include the cost of four-index transformation, which is proportional to N^5. One should stress that only a small portion of two-electron integrals is needed to perform MRMBPT(2) calculations.

The results of our EOMCC studies for states of ^1A_g symmetry in 6-31G and cc-pVDZ basis sets are shown in Tables 2 and 3. In both cases, we used the pentacene geometry of ref 69. Due to its double excited character, the D state appears in all EOMCCSD calculations as a highly excited ^1A_g state. For example, for the 6-31G basis set the D state corresponds to the sixth root of EOMCCSD equations, while for the cc-pVDZ it was found as the 10th root of the EOMCCSD eigenvalue problem. The CR-EOMCCSD(T) triples correction for both basis sets employed reduces the EOMCCSD excitation energy of D state by more than 2.0 eV. One should notice that while the excitation energies of low-lying singly excited states for both basis sets considered here are similar (for example, the CR-EOMCCSD(T) excitation energies for the 2^1A_g state in 6-31G and cc-pVDZ basis sets differ only by 0.07 eV), the analogous difference for the D state is much bigger (1.68 eV). The excitation energies of the D state are also strongly dependent on the geometry employed and the form of the noniterative corrections due to triples. For the pentacene geometry used in ref 65, the 6-31G CR-EOMCCSD-(T) and r-CR-EOMCCSD(T)$^{(II)}$ results for the 2^1A_g (S) and 3^1A_g (D) states are 4.52 and 5.01 eV and 4.62 and 4.71 eV, respectively, which provide a consistent description of singly excited states while producing serious discrepancies for the D state. The above-mentioned problems of the single-reference EOMCC methods when applied to the D state of pentacene should be attributed to the highly multiconfigurational structure of the doubly excited state of pentacene.

The results of our MRCC calculations are shown in Table 4. Due to the high numerical overhead, the BW-MRCCSD energies were converged to a threshold of 10^{-4}. Again, one can observe a strong effect of triples on the calculated VEEs. For example, the (T) correction lowers the BW-MRCCSD VEE by 0.48 eV. A similar lowering in VEE can be observed in the case of BW-MRCCSD theory with *a posteriori* corrected cluster amplitudes (BW-MRCCSD ap.).[76] This leads to an even bigger (1.23 eV) lowering of the BW-MRCCSD ap. VEE by the BW-MRCCSD(T) ap. approach (defined as a standard BW-MRCCSD(T) approach utilizing *a posteriori* corrected cluster amplitudes). The 2.44 eV excitation energy of the D state is the closest one to the MRMBPT VEE value of 1.95 eV reported in ref 65.

One should notice that the MRCC VEEs are significantly lower compared to the EOMCC ones obtained in the cc-pVDZ basis set. In order to get more accurate estimates of the D state VEE, we are planning, in the future, to use larger model spaces to include the most important singly excited configurations mentioned earlier, and to employ basis sets of triple ζ quality.

The problems experienced by the CR-EOMCSD(T) formulations in describing the doubly excited state of pentacene can be understood in light of the recent MRCC studies of the doubly excited 2^1A_1 state of ozone,[77] where MRCCSD(T) approaches outperformed the CR-EOMCCSD(T) method in accuracy of the 2^1A_1 excitation energy. While the experimental value is located around 4 eV, the calculated BW-MRCCSD(T) value is 4.2 eV. This should be compared with the 5.97 eV produced by the CR-EOMCCSD(T) approach (EOMCCSD and EOMCCSDT values are 9.82 and 5.09 eV, respectively). We think that the accuracy of the CR-EOMCCSD(T) methods in describing doubly excited states deteriorates with the system size growth, and genuine MRCC methods are needed to re-establish the desired level of accuracy. A similar situation was observed for singly excited states, which for small molecular

Table 2. Excitation Energies (in eV) of 10 Lowest-Lying Excited States of A_g Symmetry of Pentacene Obtained with Various EOMCC Methods for the 6-31G Basis Set[a]

state	EOMCCSD	CR-EOMCCSD(T)
2^1A_g(Sd)	4.70	4.44
3^1A_g(D)	7.14	4.98
4^1A_g(Sd)	6.11	5.79
5^1A_g(S)	7.13	6.53
6^1A_g(Sd)	6.88	6.60
7^1A_g(S)	6.90	6.66
8^1A_g(Sd)	7.85	7.45
9^1A_g(Sd)	8.60	8.20
10^1A_g(Sd)	8.80	8.30
11^1A_g(Sd)	8.82	8.59

[a]All core electrons were kept frozen in the EOMCC calculations. Letters S or D indicate states of singly or doubly excited character. The (Sd) acronym stands for the states of mixed singly(dominant)/doubly excited character. The ordering of states corresponds to the increasing values of the CR-EOMCCSD(T) excitation energies.

Table 3. Excitation Energies (in eV) of 10 Lowest-Lying Excited States of A_g Symmetry of Pentacene Obtained with Various EOMCC Methods for the cc-pVDZ Basis Set[a]

state	EOMCCSD	CR-EOMCCSD(T)
2^1A_g(S)	4.65	4.37
3^1A_g(S)	5.97	5.59
4^1A_g(S)	6.53	6.22
5^1A_g(S)	6.79	6.54
6^1A_g(S)	6.82	6.57
7^1A_g(D)	8.82	6.66
8^1A_g(Sd)	7.67	7.27
9^1A_g(Sd)	8.54	7.43
10^1A_g(Sd)	8.35	8.06
11^1A_g(S)	8.64	8.25

[a]All core electrons were kept frozen in the EOMCC calculations. Letters S or D indicate states of singly or doubly excited character. The (Sd) acronym stands for the states of mixed singly(dominant)/doubly excited character. The ordering of states corresponds to the increasing values of the CR-EOMCCSD(T) excitation energies.

Table 4. BW-MRCC Excitation Energies (ω_D) Corresponding to the D State of Pentacene Described by the cc-pVDZ Basis Set[a]

	BW-MRCCSD	BW-MRCCSD(T)	BW-MRCCSD ap.	BW-MRCCSD(T) ap.
E_0	−844.2465	−844.3919	−844.2623	−844.4235
E_D	−844.0967	−844.2597	−844.1274	−844.3339
ω_D	4.08	3.60	3.67	2.44

[a]All core electrons were kept frozen in the CC calculations. E_0 and E_D refer to the ground-state energy and the energy of D state, respectively. The geometry of ref 69 was used in all calculations. Total ground- (E_0) and excited-state (E_D) energies are given in Hartrees, while excitation energy ω_D is reported in eV.

dx.doi.org/10.1021/ct301130u | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

systems can be well described by the EOMCCSD approach. In order to maintain the same level of accuracy for large systems, inclusion of triple excitation in the EOMCC formalism was necessary (for detailed discusion of these problems see ref.[78]). However, one should mention that other CR-EOMCCSD(T) formulations based on the use of left eigenvectors of the similarity transformed Hamiltonian ($\bar{H}$) and employing Epstein−Nesbet perturbative partitioning of $\bar{H}$ (see the CR-EOMCC(2,3) approaches of refs 79−82) may provide more accurate estimates of excitation energies corresponding to complicated doubly excited states.

5. CONCLUSIONS

The development of efficient parallel implementations of MRCC methods capable of taking advantage of heterogeneous architectures remains one of the most important factors in enabling calculations with accurate yet expensive methodologies for large systems. The joint utilization of novel parallel tools such as processor groups and possibilities offered by GPU accelerators provides a venue to significantly extend the area of application of noniterative MRCCSD(T) approaches. Our tests clearly indicate the role of GPUs in accelerating the most numerically expensive part of the MRCCSD(T) calculations where the corrections due to triples are calculated. We demonstrated that the parallel implementation effectively load balances the work between the CPUs and GPUs, adapting to their differing computational capabilities. Our tests also showed that the implementation achieves close to anticipated speedups with increasing core counts and demonstrated that the computational resources are effectively utilized.

The developed capabilities enabled us to perform excited-state calculations for the challenging doubly excited state of pentacene. Our estimates obtained with the BW-MRCCSD(T) ap. methodology place the VEE of this state 0.5 eV above the MRMBPT predictions. We have also demonstrated that the CR-EOMCCSD(T) VEEs for the D state are located significantly above the BW-MRCCSD(T)ap. one. To obtain more accurate estimates of the D state, MRCC studies with larger basis sets and model spaces are necessary.

AUTHOR INFORMATION

Corresponding Author

*E-mail: karol.kowalski@pnnl.gov.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work has been supported by the Extreme Scale Computing Initiative (K.B.-N., S.K., O.V., H.J.J.v.D., K.K.), a Laboratory Directed Research and Development Program at Pacific Northwest National Laboratory. A large portion of the research was performed using PNNL Institutional Computing at Pacific Northwest National Laboratory and EMSL, a national scientific user facility sponsored by the Department of Energy's Office of Biological and Environmental Research and located at Pacific Northwest National Laboratory. The Pacific Northwest National Laboratory is operated for the U.S. Department of Energy by the Battelle Memorial Institute under Contract DE-AC06.76RLO-1830. Large-scale BW-MRCCSD calculations have been performed using a 2012 ASCR Leadership Computing Challenge (ALCC) award (K.B.-N., S.K., E.A., K.K.) allocation at Oak Ridge Leadership Computing Facility (OLCF).

REFERENCES

(1) Paldus, J.; Pittner, J.; Čársky, P. In *Recent Progress in Coupled Cluster Methods*; Čársky, P., Paldus, J., Pittner, J., Eds.; Springer: Berlin, 2010; pp 455−490.
(2) Lyakh, D. J.; Musiał, M.; Lotrich, V. F.; Bartlett, R. J. *Chem. Rev.* **2011**, *112*, 182−243.
(3) Owens, J. D.; Luebke, D.; Govindaraju, N.; Harris, M.; Krüger, J.; Lefohn, A. E.; Purcell, T. J. *Comput. Graph. Forum* **2007**, *26*, 80−113.
(4) Stone, J. E.; Phillips, J. C.; Freddolino, P. L.; Hardy, D. J.; Trabuco, L. G.; Schulten, K. *J. Chem. Theory Comput.* **2007**, *28*, 2618−2640.
(5) Hardy, D. J.; Stone, J. E.; Schulten, K. *Parallel Comput.* **2009**, *35*, 164−177.
(6) Stone, J. E.; Hardy, D. J.; Ufimtsev, I. S.; Schulten, K. *J. Mol. Graphics Modell.* **2010**, *29*, 116−125.
(7) Yasuda, K. *J. Comput. Chem.* **2008**, *29*, 334−342.
(8) Yasuda, K. *J. Chem. Theory Comput.* **2008**, *4*, 1230−1236.
(9) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2008**, *4*, 222−231.
(10) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2009**, *5*, 1004−1015.
(11) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2009**, *5*, 2619−2628.
(12) Isborn, C. M.; Luehr, N.; Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2011**, *7*, 1814−1823.
(13) Anderson, J. A.; Lorenz, C. D.; Travesset, A. *J. Comput. Phys.* **2008**, *227*, 5342−5359.
(14) Vogt, L.; Olivares-Amaya, R.; Kermes, S.; Shao, Y.; Amador-Bedolla, C.; Aspuru-Guzik, A. *J. Phys. Chem. A* **2008**, *112*, 2049−2057.
(15) Friedrichs, M. S.; Eastman, P.; Vaidyanathan, V.; Houston, M.; Legrand, S.; Beberg, A. L.; Ensign, D. L.; Bruns, C. M.; Pande, V. S. *J. Comput. Chem.* **2009**, *30*, 864−872.
(16) van Meel, J.; Arnold, A.; Frenkel, D.; Portegies Zwart, S.; Belleman, R. *Mol. Simulat.* **2008**, *34*, 259−266.
(17) Eastman, P.; Pande, V. S. *J. Comput. Chem.* **2010**, *31*, 1268−1272.
(18) Asadchev, A.; Allada, V.; Felder, J.; Bode, B. M.; Gordon, M. S.; Windus, T. L. *J. Chem. Theory Comput.* **2010**, *6*, 696−704.
(19) Olivares-Amaya, R.; Watson, M. A.; Edgar, R. G.; Vogt, L.; Shao, Y.; Aspuru-Guzik, A. *J. Chem. Theory Comput.* **2010**, *6*, 135−144.
(20) Levine, B. G.; LeBard, D. N.; DeVane, R.; Shinoda, W.; Kohlmeyer, A.; Klein, M. L. *J. Chem. Theory Comput.* **2011**, *7*, 4135−4145.
(21) Wu, X.; Koslowski, A.; Thiel, W. *J. Chem. Theory Comput.* **2012**, *8*, 2272−2281.
(22) DePrince, A. E.; Hammond, J. R. *J. Chem. Theory Comput.* **2011**, *7*, 1287−1295.
(23) Ma, W.; Krishnamoorthy, S.; Villa, O.; Kowalski, K. *J. Chem. Theory Comput.* **2011**, *7*, 1316−1327.
(24) Valiev, M.; Bylaska, E. J.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Van Dam, H. J. J.; Wang, D.; Nieplocha, J.; Aprà, E.; Windus, T. L.; de Jong, W. *Comput. Phys. Commun.* **2010**, *181*, 1477−1489.
(25) Balková, A.; Bartlett, R. J. *J. Chem. Phys.* **1994**, *101*, 8972−8987.
(26) Li, X.; Paldus, J. *J. Chem. Phys.* **2006**, *124*, 034112.
(27) Demel, O.; Pittner, J. *J. Chem. Phys.* **2006**, *124*, 144112.
(28) Bhaskaran-Nair, K.; Demel, O.; Pittner, J. *J. Chem. Phys.* **2008**, *129*, 184105.
(29) Bhaskaran-Nair, K.; Demel, O.; Šmydke, J.; Pittner, J. *J. Chem. Phys.* **2011**, *134*, 154106.
(30) Bhaskaran-Nair, K.; Brabec, J.; Aprà, E.; van Dam, H. J. J.; Pittner, J.; Kowalski, K. *J. Chem. Phys.* **2012**, *137*, 094112.
(31) Jeziorski, B.; Monkhorst, H. J. *Phys. Rev. A* **1981**, *24*, 1668−1681.
(32) Kowalski, K.; Piecuch, P. *J. Mol. Struct.: THEOCHEM* **2001**, *547*, 191−208.
(33) Pittner, J.; Piecuch, P. *Mol. Phys.* **2009**, *107*, 1362−3028.

(34) Evangelista, F. A.; Prochnow, E.; Gauss, J.; H., F. S., III. *J. Chem. Phys.* **2010**, *132*, 074107.
(35) Kowalski, K. *J. Chem. Phys.* **2011**, *134*, 194107.
(36) Brabec, J.; van Dam, H. J. J.; Pittner, J.; Kowalski, K. *J. Chem. Phys.* **2012**, *136*, 124102.
(37) Hubač, I. In *New Methods in Quantum Theory*; Tsipis, A., Popov, V. S., Herschbach, D. R., Avery, J. S., Eds.; Kluwer: Dordrecht, The Netherlands, 1996; NATO ASI Series 3: High Technology, Vol. *8*, pp 183−202.
(38) Mášik, J.; Hubač, I. *Adv. Quantum Chem.* **1998**, *31*, 75−104.
(39) Hubač, I.; Pittner, J.; Čársky, P. *J. Chem. Phys.* **2000**, *112*, 8779−8784.
(40) Pittner, J.; Šmydke, J. *J. Chem. Phys.* **2007**, *127*, 114103.
(41) Demel, O.; Pittner, J. *J. Chem. Phys.* **2008**, *128*, 104108.
(42) Mahapatra, U. S.; Datta, B.; Mukherjee, D. *J. Chem. Phys.* **1999**, *110*, 6171−6188.
(43) Mahapatra, U. S.; Datta, B.; Mukherjee, D. *Chem. Phys. Lett.* **1999**, *299*, 42−50.
(44) Evangelista, F. A.; Allen, W. D.; Schaefer, H. F., III. *J. Chem. Phys.* **2006**, *125*, 154113.
(45) Evangelista, F. A.; Allen, W. D.; Schaefer, H. F., III. *J. Chem. Phys.* **2007**, *127*, 024102.
(46) Evangelista, F. A.; Simmonett, A. C.; Allen, W. D.; Schaefer, H. F., III.; Gauss, J. *J. Chem. Phys.* **2008**, *128*, 124104.
(47) Das, S.; Mukherjee, D.; Kállay, M. *J. Chem. Phys.* **2010**, *132*, 074103.
(48) Prochnow, E.; Evangelista, F. A.; H., F. S., III.; Allen, W. D.; Gauss, J. *J. Chem. Phys.* **2009**, *131*, 064109.
(49) Li, X.; Paldus, J. *J. Chem. Phys.* **2010**, *133*, 184106.
(50) Li, X.; Paldus, J. *Chem. Phys. Lett.* **2010**, *496*, 183.
(51) Mahapatra, U. S.; Chattopadhyay, S. *J. Chem. Phys.* **2011**, *134*, 044113.
(52) Bhaskaran-Nair, K.; Demel, O.; Pittner, J. *J. Chem. Phys.* **2010**, *132*, 154105.
(53) Datta, D.; Mukherjee, D. *J. Chem. Phys.* **2011**, *134*, 054122.
(54) Demel, O.; Kedzuch, S.; Svana, M.; Ten-no, S.; Pittner, J.; Noga, J. *Phys. Chem. Chem. Phys.* **2012**, *14*, 4753−4762.
(55) Das, S.; Pathak, S.; Datta, D.; Mukherjee, D. *J. Chem. Phys.* **2012**, 137.
(56) Brabec, J.; Bhaskaran-Nair, K.; Govind, N.; Pittner, J.; Kowalski, K. *J. Chem. Phys.* **2012**, 137.
(57) Nieplocha, J.; Palmer, B.; Tipparaju, V.; Krishnan, M.; Trease, H.; Apra, E. *Int. J. High Perform. Comput. Appl.* **2006**, *20*, 203−231.
(58) Ma, W.; Krishnamoorthy, S.; Villa, O.; Kowalski, K. *Cluster Computing (CLUSTER), 2010 IEEE International Conference on*; Institute of Electrical and Electronics Engineers: Piscataway, NJ, 2010; pp 207−216.
(59) Ma, W.; Krishnamoorthy, S.; Villa, O.; Kowalski, K.; Agrawal, G. *Cluster Comput.* **2011**, 1−25.
(60) Brabec, J.; Pittner, J.; van Dam, H. J. J.; Aprà, E.; Kowalski, K. *J. Chem. Theory Comput.* **2012**, *8*, 487−497.
(61) Lin, Y.-Y.; Gundlach, D.; Nelson, S.; Jackson, T. *IEEE Electron Device Lett.* **1997**, *18*, 606−608.
(62) Klauk, H.; Halik, M.; Zschieschang, U.; Schmid, G.; Radlik, G.; Weber, W. *J. Appl. Phys.* **2002**, *92*, 5259−5263.
(63) Kim, G.-H.; Shtein, M.; K.P., P. *Appl. Phys. Lett.* **2011**, *98*, 093303.
(64) Smith, M. B.; Michl, J. *Chem. Rev.* **2010**, *110*, 6891−6936.
(65) Zimmerman, P. M.; Zhang, Z.; Musgrave, C. B. *Nature Chem.* **2010**, *2*, 648−652.
(66) Zimmerman, P. M.; Bell, F.; Casanova, D.; Head-Gordon, M. *J. Am. Chem. Soc.* **2011**, *133*, 19944−19952.
(67) Hehre, W. J.; Ditchfield, R.; Pople, J. A. *J. Chem. Phys.* **1972**, *56*, 2257−2261.
(68) Dunning, T., Jr. *J. Chem. Phys.* **1989**, *90*, 1007−1023.
(69) Lopata, K.; Reslan, R.; Kowalska, M.; Neuhauser, D.; Govind, N.; Kowalski, K. *J. Chem. Theory Comput.* **2011**, *7*, 3686−3693.
(70) Chang, C.; Calzado, C. J.; Amor, N. B.; Marin, J. S.; Maynau, D. *J. Chem. Phys.* **2012**, *137*, 104102.
(71) Geertsen, J.; Rittby, M.; Bartlett, R. J. *Chem. Phys. Lett.* **1989**, *164*, 57−62.
(72) Comeau, D. C.; Bartlett, R. J. *Chem. Phys. Lett.* **1993**, *207*, 414−423.
(73) Stanton, J. F.; Bartlett, R. J. *J. Chem. Phys.* **1993**, *98*, 7029−7039.
(74) Kowalski, K.; Piecuch, P. *J. Chem. Phys.* **2004**, *120*, 1715−1738.
(75) Kowalski, K.; Valiev, M. *Int. J. Quantum Chem.* **2008**, *108*, 2178−2190.
(76) Pittner, J. *J. Chem. Phys.* **2003**, *118*, 10876−10889.
(77) Bhaskaran-Nair, K.; Kowalski, K. *J. Chem. Phys.* **2012**, *137*, 216101.
(78) K., K.; Olson, R.; Krishnamoorthy, S.; Tipparaju, V.; Aprà, E. *J. Chem. Theory Comput.* **2011**, *7*, 2200−2208.
(79) Włoch, M.; Lodriguito, M.; Piecuch, P.; Gour, J. *Mol. Phys.* **2006**, *104*, 2149−2172.
(80) Piecuch, P.; Włoch, M.; Lodriguito, M.; Gour, J. In *Recent Advances in the Theory of Chemical and Physical Systems, Progress in Theoretical Chemistry and Physics*; Wilson, S., Julien, J.-P., Maruani, J., Brämdas, E., Delgado-Barrio, G., Eds.; Springer: Dordrecht, The Netherlands, 2006; Vol. *15*, pp 45−106.
(81) Piecuch, P.; Gour, J.; Włoch, M. *Int. J. Quantum Chem.* **2009**, *109*, 3268−3304.
(82) Gour, J.; Piecuch, P.; Włoch, M. *Mol. Phys.* **2010**, *108*, 2633−2646.