

Communication: An improved linear scaling perturbative triples correction for the domain based local pair-natural orbital based singles and doubles coupled cluster method [DLPNO-CCSD(T)]

Cite as: J. Chem. Phys. **148**, 011101 (2018); <https://doi.org/10.1063/1.5011798>

Submitted: 03 November 2017 • Accepted: 22 December 2017 • Published Online: 04 January 2018

Yang Guo, Christoph Riplinger, Ute Becker, et al.



View Online



Export Citation



CrossMark

ARTICLES YOU MAY BE INTERESTED IN

[A new near-linear scaling, efficient and accurate, open-shell domain-based local pair natural orbital coupled cluster singles and doubles theory](#)

The Journal of Chemical Physics **146**, 164105 (2017); <https://doi.org/10.1063/1.4981521>

[Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen](#)

The Journal of Chemical Physics **90**, 1007 (1989); <https://doi.org/10.1063/1.456153>

[Density-functional thermochemistry. III. The role of exact exchange](#)

The Journal of Chemical Physics **98**, 5648 (1993); <https://doi.org/10.1063/1.464913>

Lock-in Amplifiers
up to 600 MHz



Zurich
Instruments



Communication: An improved linear scaling perturbative triples correction for the domain based local pair-natural orbital based singles and doubles coupled cluster method [DLPNO-CCSD(T)]

Yang Guo,¹ Christoph Riplinger,¹ Ute Becker,¹ Dimitrios G. Liakos,¹ Yury Minenkov,² Luigi Cavallo,² and Frank Neese^{1,a)}

¹Max Planck Institut für Chemische Energiekonversion, Stiftstr. 34-36, D-45470 Mülheim an der Ruhr, Germany

²King Abdullah University of Science and Technology (KAUST), Physical Science and Engineering Division (PSE), KAUST Catalysis Center (KCC), 23955-6900 Thuwal, Saudi Arabia

(Received 3 November 2017; accepted 22 December 2017; published online 4 January 2018)

In this communication, an improved perturbative triples correction (T) algorithm for domain based local pair-natural orbital singles and doubles coupled cluster (DLPNO-CCSD) theory is reported. In our previous implementation, the semi-canonical approximation was used and linear scaling was achieved for both the DLPNO-CCSD and (T) parts of the calculation. In this work, we refer to this previous method as DLPNO-CCSD(T_0) to emphasize the semi-canonical approximation. It is well-established that the DLPNO-CCSD method can predict very accurate absolute and relative energies with respect to the parent canonical CCSD method. However, the (T_0) approximation may introduce significant errors in absolute energies as the triples correction grows up in magnitude. In the majority of cases, the relative energies from (T_0) are as accurate as the canonical (T) results of themselves. Unfortunately, in rare cases and in particular for small gap systems, the (T_0) approximation breaks down and relative energies show large deviations from the parent canonical CCSD(T) results. To address this problem, an iterative (T) algorithm based on the previous DLPNO-CCSD(T_0) algorithm has been implemented [abbreviated here as DLPNO-CCSD(T)]. Using triples natural orbitals to represent the virtual spaces for triples amplitudes, storage bottlenecks are avoided. Various carefully designed approximations ease the computational burden such that overall, the increase in the DLPNO-(T) calculation time over DLPNO-(T_0) only amounts to a factor of about two (depending on the basis set). Benchmark calculations for the GMTKN30 database show that compared to DLPNO-CCSD(T_0), the errors in absolute energies are greatly reduced and relative energies are moderately improved. The particularly problematic case of cumulene chains of increasing lengths is also successfully addressed by DLPNO-CCSD(T). *Published by AIP Publishing.* <https://doi.org/10.1063/1.5011798>

The importance of triples correction for coupled cluster with singles and doubles (CCSD) was recognized early on.¹ The approximated iterative or non-iterative triples correction methods are more economical than full triples.^{2,3} Derived from CCSDT-1,⁴ a non-iterative method CCSD[T] was developed by Bartlett and co-workers in 1985.⁵ By adding a singles term correction to CCSD[T], Pople and co-workers proposed the CCSD(T) method in 1989.⁶ The CCSD(T) method is the most widely used wave function method for small molecules and is considered to be the “gold standard.” However, the high computational costs limit the applicability of canonical CCSD(T) to systems with not more than about 1000 basis functions on the most powerful hardware.⁷ Many groups have worked on new developments in order to be able to apply the CCSD(T) method to medium- and large-sized systems.^{8–20} The large and fast growing literature about low order or linear scaling CCSD algorithms has been extensively reviewed in earlier publications.^{11–13} In this communication, we only

discuss the studies that are focused on the (T) correction. Schütz and Werner implemented the first projected atomic orbitals (PAOs) based linear scaling local (T) method in 2000.^{8,21} In their work, different variants of local (T) were explored. Their variant using the semi-canonical (SC) approximation is the most efficient one since it neglects the couplings between different triples by the off-diagonal Fock matrix elements and hence bypasses the storage of the triples amplitudes. This SC-based local (T) method is usually referred as LCCSD(T_0). Another alternative in their paper is called LCCSD(T), in which the triples amplitudes T_{ijk}^{abc} are computed iteratively. More accurate (T) correlation energies can be recovered using converged amplitudes compared to the LCCSD(T_0) results. However, the triples amplitudes T_{ijk}^{abc} expressed in PAOs for the virtual space have to be stored either on disk or in memory and must be updated iteratively. In the LCCSD(T) method, a large amount of storage is needed to apply the method to medium-sized systems. To overcome this difficulty, orbital-specific virtual (OSV) orbitals were used to reduce the virtual space of each triple.²² Furthermore, Schmitz and Hättig introduced the Laplace transformation (LT) technique²³ and used pair natural orbitals (PNOs)²⁴

^{a)}Author to whom correspondence should be addressed: Frank.Neese@cec.mpg.de

in their recent PNO-CCSD(T) method.^{25,26} The storage bottleneck of the triples amplitudes is completely avoided by using the LT technique. The explicitly correlated method (F12) was integrated into their method as well, which can reproduce very accurate absolute energies. Alternatively, Schütz also extended their LCCSD(T) method to the CCSDT-1b method,⁵ which can avoid the storage of triple amplitudes as well.¹⁰

Besides the above mentioned direct local correlation methods, fragment-based local CCSD(T) method development is a very active area of research.^{14–20} Popular approaches include the cluster-in-molecule (CIM),¹⁵ the natural linear-scaled coupled cluster,¹⁶ the divide-expand-consolidate (DEC),²⁰ the incremental,¹⁴ and the local natural orbital (LNO)¹⁸ methods, all of which have led to low-order or linear scaling CCSD(T) methods.^{17,19,27–29} Most recently, Nagy and Kállay combined their LNO approach with LT CCSD(T).²⁹ Their new scheme is capable to compute very accurate CCSD(T) correlation energies for systems with more than ten thousand basis functions. By calculating all intermediates on the fly, there is no storage bottleneck anymore in their new method. More importantly, by using the LT technique, there is no redundant calculation in the triples amplitudes' evaluation from the overlapping fragments.

The use of pair natural orbitals (PNOs)^{30–32} was revived and introduced into local correlation methods in 2009.²⁴ Since then, a series of linear scaling CCSD(T) methods for both closed-shell and open-shell systems have been developed.^{12,13,33,34} In all these PNO-based CCSD(T) methods, the SC approximation is used. In the following, our previous domain based local pair-natural orbital (DLPNO)-CCSD(T)¹³ method is denoted as DLPNO-CCSD(T₀), whereas the DLPNO-CCSD method with the improved perturbative triples correction, described in this communication, is abbreviated as DLPNO-CCSD(T).

Several studies have shown that the local (T₀) method is a very good approximation to compute relative or reaction energies.^{12,26} However, two of us (Y.M. and L.C.) found that DLPNO-CCSD(T₀) can fail for a few cases, for example, the atomization-like energy of cumulene chains, because of the large deviations from the canonical results caused by the (T₀) corrections. To overcome this difficulty, we here present an iterative local (T) algorithm. Our strategy is similar to Schütz's LCCSD(T) method.⁸ Only a subset of triples is considered in the (T), which scales linearly with the system size. Meanwhile, by using triples natural orbitals (TNOs) instead of PAOs, the virtual spaces of surviving triples are greatly reduced. To further speed up our algorithm, three additional approximations are introduced. In the following, after a detailed description of the algorithm, a series of benchmark calculations is conducted. Finally, a range of medium-sized to large-sized molecules is studied to demonstrate the accuracy and efficiency of the new DLPNO-CCSD(T) method.

The indices i, j, k, l are used to denote the occupied MOs, and a, b, c, d denote virtual MOs (TNOs). Note that for each ijk combination, there is a different set of quasi-canonical TNOs as denoted by a subscript a_{ijk} .¹² Using converged DLPNO-CCSD singles- and doubles-amplitudes, the energy contribution of each triple is computed by

$$E_{ijk} = (2 - \delta_{ij} - \delta_{jk}) \sum_{a_{ijk}, b_{ijk}, c_{ijk}} \hat{T}_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} \times \left[W_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} + T_i^{a_{ijk}} (j b_{ijk} | k c_{ijk}) + T_j^{b_{ijk}} (i a_{ijk} | k c_{ijk}) + T_k^{c_{ijk}} (j b_{ijk} | i a_{ijk}) \right] \times (\hat{T}_{ijk}^{abc} = 4T_{ijk}^{abc} - 2T_{ijk}^{acb} - 2T_{ijk}^{cba} - 2T_{ijk}^{bac} + T_{ijk}^{cab} + T_{ijk}^{bca}). \quad (1)$$

The $T_i^{a_{ijk}}$ in Eq. (1) is the converged singles amplitudes from the DLPNO-CCSD calculation. The number of explicitly treated index-triples ijk increases linearly with the systems size.¹³ The intermediate $W_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ has the same definition as in our previous work.¹¹ The construction of TNOs was adopted from the DLPNO-CCSD(T₀) method as well. Since localized, occupied MOs are used, the $T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ amplitudes have to be computed iteratively in order to reproduce canonical (T) results,

$$R_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} = W_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} + T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} \left(f_{a_{ijk}}^{a_{ijk}} + f_{b_{ijk}}^{b_{ijk}} + f_{c_{ijk}}^{c_{ijk}} \right) - \sum_{l, a_{ijl}, b_{ijl}, c_{ijl}} T_{ijl}^{a_{ijl} b_{ijl} c_{ijl}} f_k^l S_{a_{ijl} b_{ijl} c_{ijl}}^{a_{ijk} b_{ijk} c_{ijk}} - \sum_{l, a_{ilk}, b_{ilk}, c_{ilk}} T_{ilk}^{a_{ilk} b_{ilk} c_{ilk}} f_j^l S_{a_{ilk} b_{ilk} c_{ilk}}^{a_{ijk} b_{ijk} c_{ijk}} - \sum_{l, a_{ljk}, b_{ljk}, c_{ljk}} T_{ljk}^{a_{ljk} b_{ljk} c_{ljk}} f_i^l S_{a_{ljk} b_{ljk} c_{ljk}}^{a_{ijk} b_{ijk} c_{ijk}}, \quad (2)$$

where f_i^j is the Fock matrix elements and $S_{a_{ijl} b_{ijl} c_{ijl}}^{a_{ijk} b_{ijk} c_{ijk}}$ are overlap matrices between TNO spaces of two different triples. Note that the TNOs are chosen to be quasi-canonical, and therefore, no coupling terms arise in the virtual block. In the DLPNO-CCSD(T₀) method, the $T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ amplitudes are approximately computed by

$$W_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} + T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}} \left(f_{a_{ijk}}^{a_{ijk}} + f_{b_{ijk}}^{b_{ijk}} + f_{c_{ijk}}^{c_{ijk}} - f_i^i - f_j^j - f_k^k \right) = 0. \quad (3)$$

All intermediates and amplitudes are evaluated on the fly under the (T₀) approximation. The extension from the linear scaling (T₀) approach to (T) is straightforward. In the (T₀) calculation, only a subset of triples is considered, the number of which increases linearly with the system size. By using TNOs, the virtual space of each triple is greatly reduced. In fact, the number of TNOs per triple approaches a constant with increasing system size (on average less than 80 TNOs per triple for calculations with triple- ζ basis sets). Hence, the storage needed for all intermediates during the iteration process is not a bottleneck anymore. After the DLPNO-CCSD(T₀) calculation, the $T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ amplitudes and other intermediates can be saved on disk. Next, the $T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ amplitudes are updated according to Eq. (2) until convergence is achieved. Finally, inserting converged $T_{ijk}^{a_{ijk} b_{ijk} c_{ijk}}$ amplitudes into Eq. (1), the final local (T) correction energy is computed. Although the (T) algorithm is linear scaling, the central processing unit (CPU) time spent on the iteration step is at least five times more than the (T₀) method for medium-sized molecules. The bottleneck in the (T) calculations is the TNO space transformations between triples.

Hence, we have carefully introduced three additional approximations in order to ease the computational burden while not sacrificing any additional accuracy in the results.

Approximation 1 is based on the observation that, on average, only 20% of all triples account for more than 90% of the (T) correction. Hence, it is reasonable to focus attention on these large triples and treat them at higher accuracy, while the remaining “weak triples” are treated at lower accuracy. To this end, all surviving triples after the DLPNO-CCSD(T_0) calculations are ordered by decreasing absolute value of their energy contributions. Then, these energies are summed until 90% of the (T_0) energy is reached. These triples are denoted as “strong triples” and the remaining ijk index combinations as “weak triples.” For both strong and weak triples, smaller TNO spaces are constructed after the conventional (T_0) calculations. These are determined by two separate scaling factors ($T_{TNOscaleS}$ and $T_{TNOscaleW}$) relative to the overall TNO occupation number threshold (T_{CutTNO}) that has the default value 10^{-9} .¹³ $T_{ijk}^{a_{ijk}b_{ijk}c_{ijk}}$ with the smaller TNOs will be used in the iterative process to save computational time. Having obtained converged amplitudes in the small TNO basis, the final (T) energy is calculated as

$$E^{(T)T_{CutTNO}} \approx \sum_{i,j,k} E_{ijk}^{(T)T_{CutTNO} * T_{TNOscale}} + \Delta E_{ijk}^{(T_0)}$$

$$\left(\Delta E_{ijk}^{(T_0)} = E_{ijk}^{(T_0)T_{CutTNO}} - E_{ijk}^{(T_0)T_{CutTNO} * T_{TNOscale}} \right). \quad (4)$$

Results with different $T_{TNOscaleS}$ ($T_{TNOscaleW}$) scale factors for strong (weak) triples are given in Table S1 (see [supplementary material](#)). By setting $T_{TNOscaleS}$ as $10.0 \times T_{CutTNO}$ and $T_{TNOscaleW}$ as $100.0 \times T_{CutTNO}$, we still recover more than 99.9% of the (T) correction energies relative to the DLPNO-CCSD(T) method without the strong/weak separation and without reduced TNO spaces. From Table S1, one can see that the DLPNO-(T) algorithm is just about 2-3 times slower than its (T_0) counterpart by using the default setting for a range of medium-sized (30-50 atoms) model systems.

From Eq. (2), one notices that the couplings between triples are connected by the Fock matrix elements. For example, if $f_k^l = 0$, the contribution from $\sum_{l,a_{ijl}b_{ijl}c_{ijl}} T_{ijl}^{a_{ijl}b_{ijl}c_{ijl}}$ term is zero. In approximation 2, couplings between $f_k^l S_{a_{ijk}b_{ijk}c_{ijk}}^{a_{ijk}b_{ijk}c_{ijk}}$ and other triples are depending on the size of the off-diagonal elements of the Fock matrix. If the absolute value of f_k^l is smaller than a threshold F_{Cut} , all couplings between two triples connected by f_k^l are neglected. This approximation is commonly employed in local MP2 LMP2 implementations.^{35,36} From Table S2 (see [supplementary material](#)), one observes that the (T) correction energies converge very fast with F_{Cut} . By setting F_{Cut} to the default value 10^{-3} , the deviations with respect to the converged results are already below 1 μ hartree. Meanwhile, the speedup is about 10% relative to the calculation without approximation 2.

In order to save computer storage, the Gauss-Seidel scheme is used for the update of triples amplitudes. The (T) energies usually converge within four iterations, which is much

faster than, e.g., LMP2. Additionally, we found that the convergence speed of different triples is different. Some triples contributions converge within even fewer iterations. To further speed up the performance of the DLPNO-(T), a triple will be skipped in the next Gauss-Seidel iteration, if its energy increment with respect to the energy in the previous iteration is less than a threshold $T_{CutIter}$. Results in Table S3 (see [supplementary material](#)) show that this approximation, which represents the 3rd approximation in our algorithm, still guarantees the recovery of 99.9% of the local (T) energies by setting $T_{CutIter} = 0.001$. Meanwhile, triple-triple couplings are reduced substantially in the subsequent (T) iterations.

The algorithm proposed above has been implemented in a development version of ORCA.³⁷ The closed-shell test sets in the GMTKN30 database^{38,39} are employed to show the accuracy of the new DLPNO-(T) method with aug-cc-pVDZ basis sets (Fig. 1). The canonical CCSD(T) results are used as references. The maximum absolute deviation (MAD)

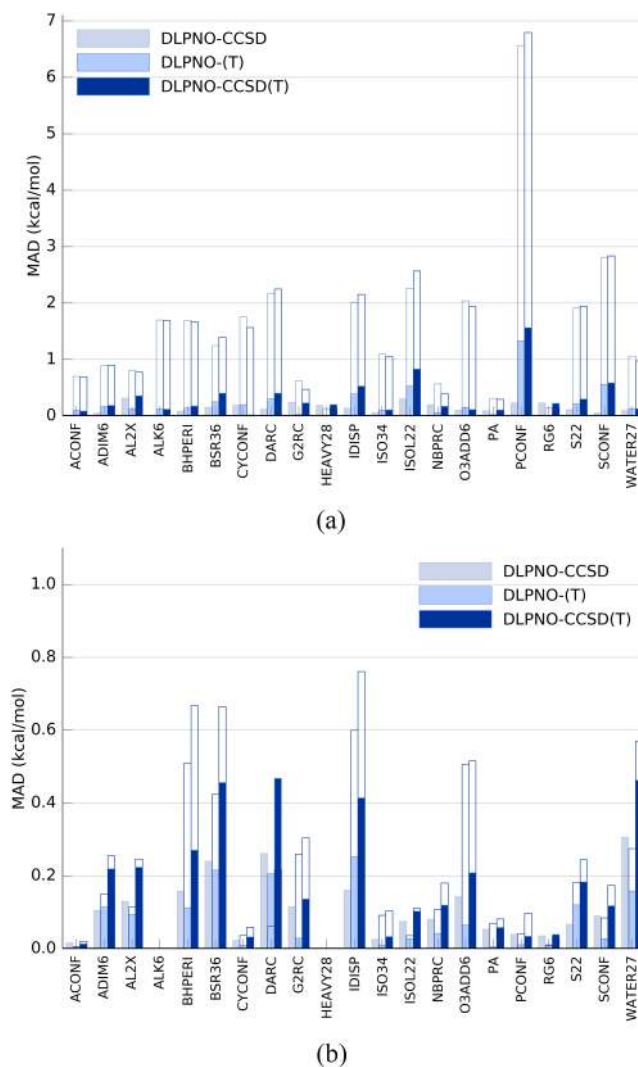


FIG. 1. Absolute (a) and relative (b) energy deviations of CCSD, (T), and total correlation energies (in kcal/mol) calculated by DLPNO-CCSD(T) with respect to the canonical CCSD(T) using aug-cc-pVDZ basis set and *TightPNO* settings for tests sets in the GMTKN30 database. The increment MAD values produced by DLPNO-(T_0) and DLPNO-CCSD(T_0) were shown as blank bars. The detailed average, MAD, and standard deviation values for each test set can be found in the [supplementary material](#) S4.

of DLPNO-(T) with respect to the canonical (T) results is less than 2 kcal/mol for all systems by using the *TightPNO* settings defined previously.¹³ By contrast, the largest MAD obtained with the previous (T_0) method is more than 6.0 kcal/mol. The deviations of CCSD and overall correlation energies with respect to the corresponding references are also shown in Fig. 1. Although, the deviations introduced by the DLPNO-CCSD approximations are smaller than (T) in some cases, the deviations between DLPNO-CCSD(T) and canonical CCSD(T) correlation energies are significantly reduced compared with DLPNO-CCSD(T_0). The relative energy deviations of DLPNO-CCSD(T) and DLPNO-CCSD(T_0) show a similar but less pronounced trend. Both DLPNO-CCSD(T_0) and DLPNO-CCSD(T) methods can reproduce the relative energies very accurately, MADs of which with respect to the canonical CCSD(T) results are below 1 kcal/mol. Nevertheless, the results computed by DLPNO-CCSD(T) are slightly better.

As discussed before, the SC approximation can introduce large deviations in special cases. For the atomization-like reaction of cumulene ($C_nH_4 \rightarrow nC(S=0) + 2H_2$), the deviation in reaction energies between DLPNO-CCSD(T_0) and canonical CCSD(T) can exceed 11.0 kcal/mol with *TightPNO* settings and cc-pVDZ basis set for $C_{16}H_4$. (The error introduced by DLPNO-CCSD part is only 1.06 kcal/mol.) With the improved triples correction, this deviation is reduced down to 2.23 kcal/mol. A detailed analysis of these results will be presented elsewhere.

In order to document the scaling behavior of the present algorithm with respect to the system size, calculations on a series of linear alkane chains are presented in Fig. 2. Since

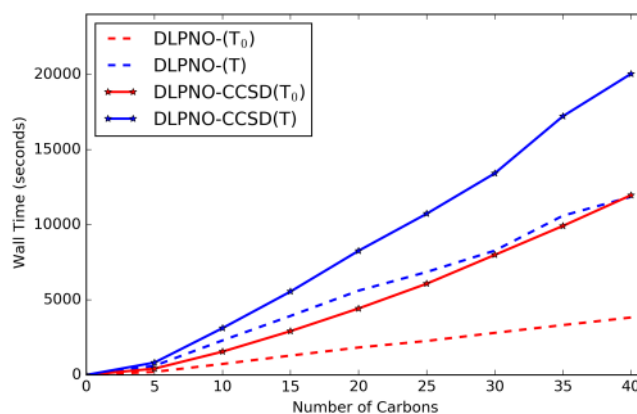


FIG. 2. The scaling behavior of DLPNO-CCSD(T) and DLPNO-CCSD(T_0) for linear alkane chains with def2-TZVP basis set. The timings for the (T)/(T_0) steps and the overall DLPNO-CCSD(T)/(T_0) calculations are plotted separately. All calculations run on cluster with two Intel(R) Xeon(R) E5-2670 processors. Only one core is used for each calculation.

the number of triples increases linearly with the number of atoms, the wall times for both methods scale linearly. However, the (T) algorithm takes about 2-3 times longer than the (T_0) algorithm. In the LT-based local (T) methods,^{25,29} usually 3-4 quadrature points are needed. Therefore, the CPU time of both Hättig's and Kallay's methods are about 3-4 times slower than their corresponding (T_0) methods. Thus, the formal prefactor of our DLPNO-CCSD(T) algorithm is comparable to or even smaller than the LT-based local CCSD(T) methods.

Finally, using the present method, a few medium-sized molecules are studied (Table I). The penicillin molecule with the def2-TZVP basis set is the largest canonical CCSD(T)

TABLE I. The accuracy and efficiency comparison for three medium-sized molecules between DLPNO-CCSD(T) and DLPNO-CCSD(T_0) with different basis set.

	Diclofenac			Penicillin			Vancomycin		
	def2-SVP	def2-TZVP	def2-QZVPP	def2-SVP	def2-TZVP	def2-QZVPP	def2-SVP	def2-TZVP	def2-QZVPP
No. basis function	329	667	1439	430	858	1921	1797	3593	8033
Accuracy									
DLPNO-(T_0)	90.32%	91.10%	-0.178 999	90.46%	91.39%	-0.219 025	-0.525 794	-0.821 973	-0.919 853
DLPNO-(T)	96.09%	96.11%	-0.188 199	95.75%	96.00%	-0.229 296	-0.555 634	-0.862 413	-0.962 056
Canonical (T) (hartree)	-0.115 737	-0.176 850	...	-0.140 410	-0.214 880
DLPNO-CCSD(T_0)	99.60%	99.59%	-4.851 412	99.60%	99.58%	-3.688 547	-15.988 954	-19.422 886	-20.804 963
DLPNO-CCSD(T)	99.84%	99.84%	-4.861 683	99.80%	99.80%	-3.697 747	-16.018 653	-19.463 164	-20.847 166
Canonical CCSD(T) (hartree)	-2.834 544	-3.459 576	...	-3.753 983	-4.550 853
Wall time (using 4 cores)									
DLPNO-(T_0) (min)	5.3	41.8	474.3	5.3	34.2	353.0	44.9	218.9 ^a	2393.5 ^a
DLPNO-CCSD(T_0) (min)	10.2	71.8	1203.0	10.3	64.5	614.7	147.7	662.4 ^a	8182.9 ^a
Ratio of DLPNO-(T)/DLPNO-(T_0)	4.6	3.2	1.6	4.0	3.2	1.5	4.3	2.7 ^a	1.5 ^a
Ratio of DLPNO-CCSD(T)/DLPNO-CCSD(T_0)	2.9	2.3	1.2	2.5	2.2	1.3	2.0	1.6 ^a	1.2 ^a
Hard disk requirement									
Additional space for intermediates in DLPNO-(T) (GB)	22.49	74.41	143.11	21.05	68.10	132.49	113.86	342.94	645.66

^aUsing 8 cores.

calculation that we can afford (24 days on 6 cores). One can see that the DLPNO-(T) recovers more than 95% of conventional (T) corrections, and the overall DLPNO-CCSD(T) method recovers 99.8% of the total correlation energies. Our method presents similar accuracy as the results reported by Hättig.²⁵ The computational cost comparison for these systems between DLPNO-CCSD(T) and DLPNO-CCSD(T₀) is given in Table I as well. The (T) step in the DLPNO-CCSD(T) calculation is always less than 4 times slower than its (T₀) counterpart. As the basis set is enlarged, the (T) calculations become more efficient. With the def2-SVP basis set, the (T) costs about 4 times as much as (T₀), whereas for the def2-QZVPP basis set the increase in computational time only costs a factor of about 0.5. In the DLPNO-(T) algorithm, the computing time of the (T₀) step is much more influenced by the basis set size than the iteration step. The computing time of the iteration step is only affected by the number of TNOs, which only slightly increases with the basis set size. Thus, as the size of the basis set enlarges, the iteration step does not dominate the (T) calculation anymore. The computational cost comparison between the overall DLPNO-CCSD(T) and DLPNO-CCSD(T₀) calculation is also given in Table I. As can be readily seen, the overall DLPNO-CCSD(T) calculation is only slightly more expensive than the DLPNO-CCSD(T₀) calculation (with an overhead of only 20%-30% for the calculations using def2-QZVPP).

In this work, an improved (T) correction, based on our previous DLPNO-CCSD(T₀) method, is reported. The accuracy of the new method was confirmed by studying absolute and relative energies over test sets in the GMTKN30 database. The new (T) algorithm scales linearly with the system size, as does the previous (T₀) algorithm. The increase of computer time of computing (T) relative to (T₀) is reasonable, but certainly not negligible. It is about an additional factor of 4 for double- ζ basis sets, a factor of 2-3 for triple- ζ basis sets, and only a factor of 0.5 for quadruple- ζ basis sets. The accuracy is definitely improved significantly for absolute energies. For relative energies, the gain in accuracy is modest, but the problematic cases for which the (T₀) approximation failed are well treated with the improved (T) scheme. The promising combination of current (T) with the DLPNO-CCSD-F12 method will be reported in due course. The extension of the DLPNO-(T) from the closed-shell to the open-shell case is undergoing and will be reported in due course as well.

See [supplementary material](#) for detailed results under the three approximations and results of test sets in GMTKN30.

F.N. and Y.G. gratefully acknowledge financial support by the Max Planck Society and the cluster of excellence

(RESOLV, University of Bochum, No. EXC 1069). Y.G. is thankful to Peter Pinski for helpful discussion.

- ¹J. Paldus, J. Čížek, and I. Shavitt, *Phys. Rev. A* **5**, 50 (1972).
- ²S. A. Kucharski and R. J. Bartlett, *Theor. Chim. Acta* **80**, 387 (1991).
- ³J. D. Watts, J. Gauss, and R. J. Bartlett, *J. Chem. Phys.* **98**, 8718 (1993).
- ⁴Y. S. Lee, S. A. Kucharski, and R. J. Bartlett, *J. Chem. Phys.* **81**, 5906 (1984).
- ⁵M. Urban, J. Noga, S. J. Cole, and R. J. Bartlett, *J. Chem. Phys.* **83**, 4041 (1985).
- ⁶K. Raghavachari, G. W. Trucks, J. A. Pople, and M. Head-Gordon, *Chem. Phys. Lett.* **157**, 479 (1989).
- ⁷T. Janowski and P. Pulay, *J. Chem. Theory Comput.* **4**, 1585 (2008).
- ⁸M. Schütz, *J. Chem. Phys.* **113**, 9986 (2000).
- ⁹M. Schütz and H.-J. Werner, *J. Chem. Phys.* **114**, 661 (2000).
- ¹⁰M. Schütz, *J. Chem. Phys.* **116**, 8772 (2002).
- ¹¹C. Riplinger and F. Neese, *J. Chem. Phys.* **138**, 034106 (2013).
- ¹²C. Riplinger, B. Sandhoefer, A. Hansen, and F. Neese, *J. Chem. Phys.* **139**, 134101 (2013).
- ¹³C. Riplinger, P. Pinski, U. Becker, E. F. Valeev *et al.*, *J. Chem. Phys.* **144**, 024109 (2016).
- ¹⁴H. Stoll, B. Paulus, and P. Fulde, *J. Chem. Phys.* **123**, 144108 (2005).
- ¹⁵S. Li, J. Shen, W. Li, and Y. Jiang, *J. Chem. Phys.* **125**, 074109 (2006).
- ¹⁶N. Flocke and R. J. Bartlett, *J. Chem. Phys.* **121**, 10935 (2004).
- ¹⁷T. F. Hughes, N. Flocke, and R. J. Bartlett, *J. Phys. Chem. A* **112**, 5994 (2008).
- ¹⁸Z. Rolik, L. Szegedy, I. Ladjánszki, B. Ladóczki *et al.*, *J. Chem. Phys.* **139**, 094105 (2013).
- ¹⁹J. J. Eriksen, P. Baudin, P. Ettenhuber, K. Kristensen *et al.*, *J. Chem. Theory Comput.* **11**, 2984 (2015).
- ²⁰M. Ziolkowski, B. Jansik, T. Kjaergaard, and P. Jorgensen, *J. Chem. Phys.* **133**, 014107 (2010).
- ²¹M. Schütz and H.-J. Werner, *Chem. Phys. Lett.* **318**, 370 (2000).
- ²²M. Schütz, J. Yang, G. K.-L. Chan, F. R. Manby *et al.*, *J. Chem. Phys.* **138**, 054109 (2013).
- ²³P. Constans, P. Y. Ayala, and G. E. Scuseria, *J. Chem. Phys.* **113**, 10451 (2000).
- ²⁴F. Neese, F. Wennmohs, and A. Hansen, *J. Chem. Phys.* **130**, 114108 (2009).
- ²⁵G. Schmitz and C. Hättig, *J. Chem. Phys.* **145**, 234107 (2016).
- ²⁶G. Schmitz and C. Hättig, *J. Chem. Theory Comput.* **13**, 2623 (2017).
- ²⁷W. Li, P. Piecuch, J. R. Gour, and S. Li, *J. Chem. Phys.* **131**, 114109 (2009).
- ²⁸J. Friedrich and M. Dolg, *J. Chem. Theory Comput.* **5**, 287 (2009).
- ²⁹P. R. Nagy and M. Kállay, *J. Chem. Phys.* **146**, 214106 (2017).
- ³⁰C. Edmiston and M. Krauss, *J. Chem. Phys.* **42**, 1119 (1965).
- ³¹W. Meyer, *J. Chem. Phys.* **58**, 1017 (1973).
- ³²R. Ahlrichs, F. Driessler, H. Lischka, V. Staemmler *et al.*, *J. Chem. Phys.* **62**, 1235 (1975).
- ³³A. Hansen, D. G. Liakos, and F. Neese, *J. Chem. Phys.* **135**, 214102 (2011).
- ³⁴M. Saitow, U. Becker, C. Riplinger, E. F. Valeev *et al.*, *J. Chem. Phys.* **146**, 164105 (2017).
- ³⁵M. Schütz, G. Hetzer, and H.-J. Werner, *J. Chem. Phys.* **111**, 5691 (1999).
- ³⁶P. Pinski, C. Riplinger, E. F. Valeev, and F. Neese, *J. Chem. Phys.* **143**, 034108 (2015).
- ³⁷F. Neese, *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **8**, e1327 (2018).
- ³⁸L. Goerigk and S. Grimme, *J. Chem. Theory Comput.* **7**, 291 (2011).
- ³⁹L. Goerigk and S. Grimme, *Phys. Chem. Chem. Phys.* **13**, 6670 (2011).