JCTC Journal of Chemical Theory and Computation

Article

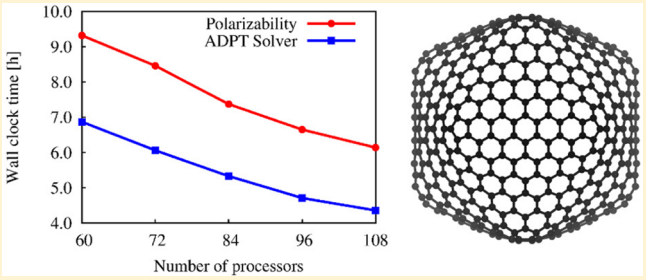
pubs.acs.org/JCTC

Robust and Efficient Auxiliary Density Perturbation Theory Calculations

Daniel Mejía-Rodríguez,* Rogelio Isaac Delgado Venegas, Patrizia Calaminici, and Andreas M. Köster

Departamento de Química, Cinvestav, Avenida Instituto Politénico Nacional 2508, A.P. 14-740 México D.F. 07000, México

ABSTRACT: A new iterative solver for the recently developed time-dependent auxiliary density perturbation theory is presented. It is based on the Eirola−Nevanlinna algorithm for large nonsymmetric linear equation systems. The new methodology is validated by static and dynamic polarizability calculations of small molecules. Comparison between the analytic and iterative solutions of the response equation system shows excellent agreement for the calculated static and dynamic polarizabilities. The new iterative solver reduces the formal scaling from $O(N^4)$ to $O(N^3)$. Furthermore, the observed computational scaling for linear alkane chains is $N^{1.6}$. This subquadratic behavior is possible in systems with a few hundred atoms because of the very small prefactors of the $O(N^3)$ and $O(N^2)$ steps remaining in the iterative solver. To demonstrate the potential of this new methodology, static polarizabilities of giant fullerenes up to C_{960}, with more than 14,000 basis functions, are calculated.



1. INTRODUCTION

Many molecular properties are obtained as (higher order) perturbations of molecular energy with respect to corresponding perturbation parameters. Unfortunately, in most cases, the perturbation problem has to be solved iteratively due to the large size of the equation systems involved. In the framework of Kohn−Sham density functional theory (DFT),[1,2] the coupled-perturbed Kohn−Sham (CPKS) method[3−6] is usually used for this purpose. Recent efforts to reduce the computational complexity of CPKS methods have been carried out by several authors,[7−15] including extensions to periodic systems.[16] For a comprehensive review of these methods, see refs 17 and 18. In a different series of efforts, noniterative algorithms for Kohn−Sham perturbation theory calculations have been developed.[19−27] In the framework of auxiliary density functional theory (ADFT),[28] static and dynamic perturbed density matrices[25,26] can be efficiently obtained directly from the corresponding perturbed approximated density. Because ADFT is based on variational fitting of the Coulomb potential[29,30] and uses the resulting approximated density for the calculation of exchange-correlation energy and its derivatives,[28] it is well suited for the calculation of large molecules with hundreds to thousands of atoms. From this use of the approximated density, an alternative formulation of the CPKS method is possible. The resulting perturbation equation system possesses only the dimension of the number of auxiliary functions[25,26] used to expand the approximated density. Therefore, a noniterative solution for the perturbation equation system becomes feasible. Corresponding to ADFT, this new methodology was named auxiliary density perturbation theory (ADPT).[25,26,31] To date, ADPT has been successfully applied for the calculation of static and dynamic dipole−dipole polarizabilities,[23,26] static dipole−quadrupole polarizabilities,[24] static and dynamic first hyperpolarizabilities,[31] and Fukui functions.[32,33]

The current implementation of ADPT in deMon2k[34] builds and inverts the response matrix, **R**. The computational bottleneck of this approach is not the $O(N^3)$ matrix inversion but rather the $O(N^4)$ building of **R**. To avoid this bottleneck, one may choose to solve the ADPT equation system by an iterative procedure. Such an approach has been taken in deMon2k for the solution of the Coulomb fitting equation system, which also has the dimension of the number of auxiliary functions, resulting in a density fitting algorithm useful for systems with more than 30,000 auxiliary functions.[35] However, different than the symmetric positive-definite Coulomb fitting equation system, the response equation system is nonsymmetric and indefinite. Traditional algorithms developed for these type of equations systems are so-called Krylov projection type methods[36] and include BiCG, BiCGSTAB,[37] CGS,[38] QMR,[39] and GMRES.[40] However, some of them lack the desirable global residual minimization properties or require the storage of a large number of subspace basis vectors. As an alternative, a new class of efficient nonsymmetric iterative solvers has been introduced on the basis of the Eirola−Nevanlinna (EN) algorithm.[41,42] In a recent study,[43] it has been shown that these EN algorithms perform equally well as the more traditional approaches. Furthermore, in the framework of the ADPT equation systems for multiple perturbations, it is beneficial that the EN algorithm generates, and improves, an approximated inverse coefficient matrix, which can be used for the following perturbations.

Received: November 27, 2014
Published: March 10, 2015

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493−1500

In this contribution, we present a robust iterative solver for the ADPT equation system on the basis of the EN algorithm. It reduces the formal scaling of the perturbation theory algorithm by 1 order of magnitude. For this purpose, we first discuss the ADPT methodology and its current implementation in deMon2k. Then, we develop the iterative ADPT solver based on the EN algorithm. Finally, validation and benchmark calculations on test systems will be discussed to show the advantages of the developed algorithm.

2. THEORY

Throughout the manuscript, perturbed quantities are denoted by a λ superscript enclosed in parentheses, small Greek letters (μ, ν) are used to symbolize atomic orbitals, and small Latin letters with a bar ($\bar{k}$, $\bar{l}$) are used to denote auxiliary functions. The Coulomb operator $1/|\mathbf{r}_1 - \mathbf{r}_2|$ is represented as a double bar $\|$, and the identity matrix is represented as $\mathbf{E}$.

2.1. Auxiliary Density Perturbation Theory. According to McWeeny's self-consistent perturbation method,[44] an element of the dynamic first-order perturbed density matrix is given by

$$\mathbf{P}^{(\lambda)}_{\mu\nu}(\omega) = 2\sum_{i}^{\text{occ}}\sum_{a}^{\text{uno}} \frac{\mathcal{K}^{(\lambda)}_{ia}(\omega)}{\epsilon_i - \epsilon_a - \omega} c_{\mu i} c_{\nu a} + 2\sum_{i}^{\text{occ}}\sum_{a}^{\text{uno}} \frac{\mathcal{K}^{(\lambda)}_{ia}(\omega)}{\epsilon_i - \epsilon_a + \omega} c_{\mu a} c_{\nu i} \tag{1}$$

where $\mathcal{K}^{(\lambda)}(\omega)$ is the molecular orbital representation of the perturbed Kohn–Sham matrix whose elements are given by

$$\mathcal{K}^{(\lambda)}_{ia}(\omega) = \sum_{\mu,\nu} c_{\mu i} K^{(\lambda)}_{\mu\nu}(\omega) c_{\nu a} \tag{2}$$

with

$$K^{(\lambda)}_{\mu\nu}(\omega) = H^{(\lambda)}_{\mu\nu}(\omega) + \sum_{\bar{k}} \langle \mu\nu \| \bar{k} \rangle (x^{(\lambda)}_{\bar{k}}(\omega) + z^{(\lambda)}_{\bar{k}}(\omega)) \tag{3}$$

The perturbed fitting coefficients, $\mathbf{x}^{(\lambda)}(\omega)$, and the perturbed exchange-correlation coefficients, $\mathbf{z}^{(\lambda)}(\omega)$, can be obtained through the solution of the ADPT response equation system

$$\left[\frac{1}{4}\mathbf{G} - \mathbf{A}(\omega)(\mathbf{E} + \mathbf{F})\right]\mathbf{x}^{(\lambda)}(\omega) = \mathbf{R}(\omega)x^{(\lambda)}(\omega) = \mathbf{b}^{(\lambda)}(\omega) \tag{4}$$

and the relation

$$\mathbf{z}^{(\lambda)}(\omega) = \mathbf{F}\mathbf{x}^{(\lambda)}(\omega) \tag{5}$$

The elements of the Coulomb matrix $\mathbf{G}$, the frequency-dependent Coulomb response matrix $\mathbf{A}(\omega)$, and the kernel matrix $\mathbf{F}$ are given by

$$G_{\bar{k}\bar{l}} = \langle \bar{k} \| \bar{l} \rangle \tag{6}$$

$$A_{\bar{k}\bar{l}} = \sum_{i}^{\text{occ}}\sum_{a}^{\text{uno}} \langle \bar{k} \| ia \rangle \frac{\epsilon_i - \epsilon_a}{(\epsilon_i - \epsilon_a)^2 - \omega^2} \langle ia \| \bar{l} \rangle \tag{7}$$

$$F_{\bar{k}\bar{l}} = \sum_{\bar{m}} G^{-1}_{\bar{k}\bar{m}} \langle \bar{m} | f_{\text{xc}} | \bar{l} \rangle \tag{8}$$

respectively, where

$$f_{\text{xc}}[\tilde{\rho};\mathbf{r}] = \frac{\delta v_{\text{xc}}[\tilde{\rho};\mathbf{r}]}{\delta\tilde{\rho}(\mathbf{r}')}\delta(\mathbf{r} - \mathbf{r}') \tag{9}$$

is the time-independent kernel that corresponds to the time-independent exchange-correlation potential $v_{\text{xc}}[\tilde{\rho};\mathbf{r}]$. In the case of perturbation-independent basis sets, $\mathbf{b}^{(\lambda)}(\omega)$ collects terms related to operator derivatives. Further insights into the derivation of the ADPT equations are given in refs 25 and 26. The computational complexity of $\mathbf{G}$ is of $O(N^2)$, the corresponding complexity of $\mathbf{A}(\omega)$ is of $O(N^4)$, and that of $\mathbf{F}$ is of $O(N^3)$. Explicit computation of these three matrices can be avoided by an iterative procedure where only the action of $\mathbf{R}(\omega)$ on a trial vector $\mathbf{p}$ is required. We now discuss all three actions (from the matrices $\mathbf{G}$, $\mathbf{A}(\omega)$, and $\mathbf{F}$) and the corresponding computational complexity expected for each of them.

The first action is given by multiplication of the kernel matrix $\mathbf{F}$ with a trial vector $\mathbf{p}$. It can be divided into two steps. In the first step, a vector $\mathbf{f}$ with elements

$$f_{\bar{k}} = \langle \bar{k} | f_{\text{xc}} | \tilde{\sigma} \rangle \tag{10}$$

is formed. Here, $\tilde{\sigma}(\mathbf{r})$ is given by

$$\tilde{\sigma}(\mathbf{r}) = \sum_{\bar{l}} p_{\bar{l}}\, \bar{l}(\mathbf{r}) \tag{11}$$

This step involves the computation of a vector with the size of the auxiliary function set over the grid. Thus, the formal scaling is expected to be $O(N \times N_{\text{grid}})$. This scaling can be further reduced by grid screening techniques. The second step is the multiplication of the obtained $\mathbf{f}$ vector by the inverse of the Coulomb matrix (i.e., $\mathbf{f}' = \mathbf{G}^{-1}\mathbf{f}$). At this point in the calculation, $\mathbf{G}^{-1}$ is stored on a disk. Therefore, this second step involves only a matrix-vector multiplication with complexity $O(N^2)$.

The second action arises from multiplication of the Coulomb matrix $\mathbf{G}$ with the same trial vector $\mathbf{p}$. As in the previous case, at this point in the calculation, $\mathbf{G}$ is available from the disk, and the complexity is again $O(N^2)$.

The last action is given by the multiplication of $\mathbf{A}(\omega)$ with the new vector $\mathbf{u} \equiv \mathbf{p} + \mathbf{f}'$. The elements of the resulting vector $\mathbf{a}(\omega)$ are given by

$$a_{\bar{k}}(\omega) = \sum_{\bar{l}} A_{\bar{k}\bar{l}}(\omega)\, u_{\bar{l}} = \sum_{\bar{l}}\sum_{i}^{\text{occ}}\sum_{a}^{\text{uno}} \langle \bar{k} \| ia \rangle \frac{\epsilon_i - \epsilon_a}{(\epsilon_i - \epsilon_a)^2 - \omega^2} \langle ia \| \bar{l} \rangle u_{\bar{l}} \tag{12}$$

Expanding the molecular orbitals $\psi_i(\mathbf{r})$ and $\psi_a(\mathbf{r})$ in the three-center ERIs and rearranging the sums, eq 12 transforms into

$$a_{\bar{k}}(\omega) = \sum_{\sigma,\tau} \langle \bar{k} \| \sigma\tau \rangle \sum_{i}^{\text{occ}}\sum_{a}^{\text{uno}} c_{\sigma i} c_{\tau a} \frac{\epsilon_i - \epsilon_a}{(\epsilon_i - \epsilon_a)^2 - \omega^2} \times \sum_{\mu,\nu} c_{\mu i} c_{\nu a} \sum_{\bar{l}} \langle \mu\nu \| \bar{l} \rangle u_{\bar{l}} \tag{13}$$

The last sum on the right-hand side is very similar to the one used to obtain the Kohn–Sham matrix in the direct self-consistent field (SCF) procedure.[45] The formal scaling is $O(N^3)$ but can be further reduced below $O(N^2)$ with integral

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493–1500

screening and the double asymptotic expansion technique.[46] Defining a new matrix $\mathbf{Q}$ with elements

$$Q_{\mu\nu} = \sum_{\bar{l}} \langle \mu\nu \| \bar{l} \rangle u_{\bar{l}} \tag{14}$$

and substituting it into eq 13 yields

$$a_{\bar{k}}(\omega) = \sum_{\sigma,\tau} \langle \bar{k} \| \sigma\tau \rangle \sum_{i}^{\mathrm{occ}} \sum_{a}^{\mathrm{uno}} c_{\sigma i} c_{\tau a} \frac{\epsilon_i - \epsilon_a}{(\epsilon_i - \epsilon_a)^2 - \omega^2} \times \sum_{\mu,\nu} c_{\mu i} c_{\nu a} Q_{\mu\nu} \tag{15}$$

The next step is the transformation of the matrix $\mathbf{Q}$ into its molecular orbital representation, namely,

$$Q_{ia} = \sum_{\mu,\nu} c_{\mu i} c_{\nu a} Q_{\mu\nu} \tag{16}$$

which is decomposed into two $O(N^3)$ steps. BLAS[47−51] subroutines are used to obtain the maximum FLOP rate of the processors in this step. Then, an $O(N^2)$ transformation

$$Q'_{ia}(\omega) = Q_{ia} \frac{\epsilon_i - \epsilon_a}{(\epsilon_i - \epsilon_a)^2 - \omega^2} \tag{17}$$

follows. Afterward, this matrix is transformed to atomic orbital representations. Again, this step can be formulated as two matrix−matrix multiplications with formal $O(N^3)$ scaling

$$T_{\sigma\tau}(\omega) = \sum_{i}^{\mathrm{occ}} \sum_{a}^{\mathrm{uno}} c_{\sigma i} c_{\tau a} Q'_{ia}(\omega) \tag{18}$$

Using eqs 16, 17, and 18, eq 15 transforms into

$$a_{\bar{k}}(\omega) = \sum_{\sigma,\tau} \langle \bar{k} \| \sigma\tau \rangle T_{\sigma\tau}(\omega) \tag{19}$$

which represents the last step in calculating $a_{\bar{k}}(\omega)$. Like the first step, this last step is equivalent to building of the Coulomb vector $\mathbf{J}$ in the direct SCF procedure. The complexity is again $O(N^3)$ and can be further reduced below $O(N^2)$ with integral screening and the double asymptotic expansion technique.[45,46]

Overall, there are six steps with formal $O(N^3)$ complexity and four steps with formal $O(N^2)$ complexity. Thus, the formal scaling for solving the ADPT equation system decreases from $O(N^4)$ to $O(N^3)$. Two $O(N^3)$ steps can be further reduced below $O(N^2)$ by integral screening and the double asymptotic expansion technique. The remaining $O(N^3)$ steps can be hidden in systems with a few hundred atoms using specialized BLAS routines such as the ones implemented in the Intel Math Kernel Library.

Besides the reduction in computational complexity, the iterative solution of the ADPT response equation system also allows for a reduction in the memory demand. Explicit construction of $\mathbf{A}$ requires memory on the order of $N_{\mathrm{aux}} \times N_{\mathrm{occ}} \times N_{\mathrm{uno}}$, which grows cubically with system size and forces the algorithm to recalculate three-center electron repulsion integrals many times. In contrast, the iterative solution only needs N^2_{basis} matrices and N_{aux} vectors; therefore, the memory needed by the ADPT iterative solver is equivalent to the memory used during the SCF procedure. This means that ADPT response calculations are possible for all systems that pass the SCF energy calculation.

2.2. Eirola−Nevanlinna Algorithm. In the previous subsection, we showed that an iterative algorithm for the solution of the ADPT equation system reduces the computational scaling of the code by at least 1 order of magnitude compared to the direct solution. However, a prerequisite for such an approach is a robust and efficient iterative solver for the ADPT equation system. In 1989, Eirola and Nevanlinna proposed an iterative algorithm to solve nonsymmetric linear equation systems, which is accelerated via rank-one updates of an approximate inverse coefficient matrix.[41] Figure 1 shows the

1: **procedure** EN1
2: Initialization: $\mathbf{x}_0^{(\lambda)}$, $\mathbf{H}_0$ arbitrary, $\mathbf{r}_0 = \mathbf{b}^{(\lambda)} - \mathbf{R}\mathbf{x}_0^{(\lambda)}$
3: **for** $j = 0, n$ **do**
4: **if** $\|\mathbf{r}_j\|_\infty \leq \tau$ **then**
5: Exit
6: **end if**
7: $\mathbf{u}_j = \mathbf{H}_j(\mathbf{E} - \mathbf{R}\mathbf{H}_j)\mathbf{r}_j$
8: $\mathbf{v}_j = \frac{1}{|\mathbf{R}\mathbf{u}_j|^2}(\mathbf{E} - \mathbf{H}_j^T\mathbf{R}^T)\mathbf{R}\mathbf{u}_j$
9: $\mathbf{H}_{j+1} = \mathbf{H}_j + \mathbf{u}_j\mathbf{v}_j^T$
10: $\mathbf{x}_{j+1}^{(\lambda)} = \mathbf{x}_j^{(\lambda)} + \mathbf{H}_{j+1}\mathbf{r}_j$
11: $\mathbf{r}_{j+1} = \mathbf{b}^{(\lambda)} - \mathbf{R}\mathbf{x}_{j+1}^{(\lambda)}$
12: **end for**
13: **end procedure**

Figure 1. Original Eirola−Nevanlinna iterative algorithm (EN1) for solving nonsymmetric linear equation systems.

Eirola−Nevanlinna (EN) algorithm EN1 as proposed in the original paper. In terms of the ADPT response equation system, $\mathbf{H}_0$ is an arbitrary guess to the inverse response matrix $\mathbf{R}$, and $\mathbf{x}_0^{(\lambda)}$ is a guess for the perturbed fitting coefficients. In exact arithmetic, the EN algorithm yields the solution in at most n steps for an $n \times n$ linear equation system under the assumption that all $\mathbf{H}_j$ are nonsingular. As an initial guess for solving the ADPT response equation system, we use $\mathbf{G}^{-1}$ for $\mathbf{H}_0$ and set $\mathbf{x}_0^{(\lambda)}$ to zero. The disadvantage of the EN1 algorithm is that the action of $\mathbf{R}$ on different vectors has to be computed four times (Figure 1, once in line 7, twice in line 8, and once in line 11). For situations in which it is more efficient to compute a linear combination of $j + 1$ vectors instead of multiplying a vector by $\mathbf{R}$, the alternative algorithm EN2[41,52,53] (Figure 2) is more convenient. In algorithm EN2, only two actions of $\mathbf{R}$ are needed per iteration (Figure 2, lines 7 and 9 for the first action, and lines 10 and 12 for the second). The updates to the inverse coefficient matrix are not performed explicitly, instead only the actions of those updates are carried through the vectors $\mathbf{u}$ and $\mathbf{c}$ (lines 8−14). We note that other low-memory modifications of the original EN algorithm have been suggested by Gao et al.[53] and Yang and Gallivan.[42] However, the EN2 algorithm requires one less $\mathbf{R}$ action than those. Moreover, it has lower memory demands and shows the best convergence behavior for the ADPT response equation system. Therefore, we chose the EN2 algorithm. To avoid memory overflow, a restart version of the EN2 algorithm that drops all previous $\mathbf{c}$ and $\mathbf{u}$ vectors after m steps was implemented in deMon2k. All ADPT calculations, up to systems containing almost 1000 atoms, have converged

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493−1500

1: **procedure** EN2
2: Initialization: $\mathbf{x}_0^{(\lambda)}$, $\mathbf{H}_0$ arbitrary, $\mathbf{r}_0 = \mathbf{b}^{(\lambda)} - \mathbf{R}\mathbf{x}_0^{(\lambda)}$
3: **for** $j = 0, n$ **do**
4: **if** $\|\mathbf{r}_j\|_\infty \leq \tau$ **then**
5: Exit
6: **end if**
7: $\alpha_i = \mathbf{c}_i^T(\mathbf{r}_j - \mathbf{R}\mathbf{H}_0\mathbf{r}_j)$ for $i = 0, \ldots, j-1$
8: $\boldsymbol{\eta}_j = \mathbf{H}_0\mathbf{r}_j + \sum_{i=0}^{j-1} \alpha_i \mathbf{u}_i$
9: $\boldsymbol{\xi}_j = \mathbf{r}_j - \mathbf{R}\mathbf{H}_0\mathbf{r}_j - \sum_{i=0}^{j-1} \alpha_i \mathbf{c}_i$
10: $\beta_i = \mathbf{c}_i^T \mathbf{R}\mathbf{H}_0\boldsymbol{\xi}_j$ for $i = 0, \ldots, j-1$
11: $\mathbf{u}_j = \tau(\mathbf{H}_0\boldsymbol{\xi}_j - \sum_{i=0}^{j-1} \beta_i \mathbf{u}_i)$
12: $\mathbf{c}_j = \tau(\mathbf{R}\mathbf{H}_0\boldsymbol{\xi}_j - \sum_{i=0}^{j-1} \beta_i \mathbf{c}_i)$, where τ is such that $\|\mathbf{c}_j\|_2 = 1$
13: $\mathbf{x}_{j+1}^{(\lambda)} = \mathbf{x}_j^{(\lambda)} + \boldsymbol{\eta}_j + \mathbf{u}_j \mathbf{c}_j^T \boldsymbol{\xi}_j$
14: $\mathbf{r}_{j+1} = \boldsymbol{\xi}_j - \mathbf{c}_j \mathbf{c}_j^T \boldsymbol{\xi}_j$
15: **end for**
16: **end procedure**

Figure 2. Modified Eirola–Nevanlinna iterative algorithm (EN2) for solving nonsymmetric linear equation systems.

within 9 iterations reaching the default tolerance criteria, τ, of 10^{-5}. Therefore, m is set to 15 as a balance between memory requirement and flexibility for more difficult cases. Our studies on the restart behavior of the EN2 algorithm for ADPT indicate a small increase in the number of iteration cycles needed for convergence. In all cases, calculations with the restarted EN2 algorithm converged to the same result as the nonrestarted ones. Besides the operations detailed in the previous section, all the extra operations needed by the iterative algorithm are either matrix–vector or vector–vector multiplications with complexity $O(N^2)$ at most.

3. VALIDATION

To validate our new iterative procedure, we compared static and dynamic polarizabilities of small molecules obtained with the direct and iterative solver. The experimental geometries of the molecules were used for this comparison.[54] We used DFT-optimized valence triple-ζ plus polarization (TZVP)[55] basis sets that are augmented by field-induced polarization (FIP) functions as proposed by Zeiss et al.[56] These TZVP-FIP1 basis sets are described in detail in refs 57 and 58. To avoid contamination of the valence basis set with the diffuse FIP functions, spherical basis functions are used in all calculations. The GEN-A2* auxiliary function set,[59] which contains *s*-, *p*-, *d*-, *f*-, and *g*-type auxiliary functions, is used. For the local density approximation (LDA), the Dirac exchange[60] in combination with the VWN correlation[61] is employed. The exchange–correlation energies and potentials are numerically integrated on an adaptive grid with 10^{-5} au grid accuracy.[62,63] The same grid is used for the exchange–correlation kernel calculation. The reported mean polarizabilities are calculated from the diagonal elements of the polarizability tensor by

$$\bar{\alpha}(\omega) = \frac{1}{3}\left(\alpha(\omega)_{xx} + \alpha(\omega)_{yy} + \alpha(\omega)_{zz}\right) \tag{20}$$

The corresponding polarizability anisotropies are calculated in the principal axes system of the polarizability tensor according to

$$|\Delta\alpha(\omega)|^2 = \frac{1}{2}\Big[(\alpha_{xx}(\omega) - \alpha_{yy}(\omega))^2 + (\alpha_{xx}(\omega) - \alpha_{zz}(\omega))^2 + (\alpha_{yy}(\omega) - \alpha_{zz}(\omega))^2\Big] \tag{21}$$

Table 1 shows the comparison of static polarizabilities and polarizability anisotropies (in au) obtained from the direct and iterative solutions of the ADPT equations. Also, experimental results are included as reference. The comparison between ADPT and experiment has been previously discussed[25,26] and therefore will not be discussed further here. Instead, we focus on the comparison of the two different ADPT equation system solvers. The agreement between direct and iterative solutions is almost perfect with the largest absolute deviations of 0.01 au for the average polarizability and 0.02 au for the polarizability anisotropy. In both cases, the mean absolute deviation is below 0.01%. Further tightening of the tolerance criteria for the residual vector does not change these results. Dynamic polarizabilities show practically the same behavior (Table 2) with the largest absolute deviations of 0.01 au for the average values and 0.02 au for the corresponding anisotropies. In the case of open-shell molecules, NO and O_2, the restricted open-shell Kohn–Sham (ROKS) algorithm was employed. These results demonstrate that the iterative solver does not alter the accuracy obtained by ADPT for both static and dynamic polarizabilities. Furthermore, the iterative solver took no more than 7 iterations to achieve the convergence criteria. This

Table 1. Comparison of Static LDA Polarizabilities and Polarizability Anisotropies of Small Molecules Obtained with the Direct and Iterative Solver for the ADPT Equation System with Available Experimental Data Listed for Comparison[a]

	direct		iterative		experimental	
molecule	$\bar{\alpha}$	$\|\Delta\alpha\|$	$\bar{\alpha}$	$\|\Delta\alpha\|$	$\bar{\alpha}$	$\|\Delta\alpha\|$
HF	5.89	1.13	5.89	1.13	5.40[c]	1.35[d]
CH_4	17.38		17.38		17.27[e]	
C_2H_2	23.40	12.52	23.39	12.52	22.68[e]	11.83[f]
CH_3F	17.69	1.49[b]	17.69	1.49[b]	17.32[e]	1.41[g]
HCl	18.02[b]	1.96	18.02[b]	1.96	17.54[h]	1.47[I]
H_2S	24.72	0.57[b]	24.72	0.57[b]	24.66[j]	0.67[k]
CH_2F_2	18.46	1.92[b]	18.46	1.92[b]	18.20[e]	1.70[I]
OCS	34.10	25.25	34.10	25.25	34.33[I]	26.26[f]
SO_2	25.65	13.26	25.64	13.26	25.49[l]	12.98[l]
CHF_3	19.65	1.41[b]	19.65	1.41[b]	18.69[e]	1.46[g]
CF_4	19.93		19.93		19.53[e]	
CS_2	54.27	54.95	54.26	54.93	55.38[e]	57.38[f]

[a]For all molecules, experimental geometries are used. All values are in au. [b]Calculated dynamic values at experimental λ. [c]Static value from refractive index dispersion (ref 73). [d]Static value from molecular beam electric resonance (ref 74.). [e]Static value from refractive index dispersion (ref 75). [f]Deduced from static estimates of ref 76. [g]Dynamic values at λ = 632.8 nm (ref 77). [h]Depolarized light scattering at λ = 632.8 nm (ref 78). [I]Static value from molecular beam electric resonance (ref 75). [j]Extrapolated static value from dispersion dynamic mean polarizability (ref 79). [k]Dynamic value at λ = 632.8 nm from Kerr effect (ref 81). [l]Static value from refractive index and Rayleigh scattering dispersion (ref 79).

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493–1500

Table 2. Comparison of Dynamic LDA Polarizabilities and Polarizability Anisotropies of Small Molecules Obtained with the Direct and Iterative Solver for the ADPT Equation System with Available Experimental Data Listed for Comparison[a]

	direct		iterative		experimental	
molecule	$\bar{\alpha}$	$\lvert\Delta\alpha\rvert$	$\bar{\alpha}$	$\lvert\Delta\alpha\rvert$	$\bar{\alpha}$	$\lvert\Delta\alpha\rvert$
NH_3	15.29	2.64	15.29	2.64	14.98[b]	1.94[b]
H_2O	10.46	0.14	10.46	0.13	9.92[c]	0.66[c]
N_2	11.99	5.06	11.99	5.05	11.95[b]	4.70[b]
CO	13.70	3.34	13.70	3.35	13.34[b]	3.59[b]
NO	11.97	5.54	11.97	5.53	11.74[b]	5.70[b]
O_2	10.56	6.18	10.56	6.18	10.78[b]	7.42[b]
N_2O	19.82	19.46	19.82	19.45	20.24[b]	19.97[b]
CO_2	17.72	13.88	17.72	13.89	17.75[b]	14.17[b]
Cl_2	31.03	15.45	31.03	15.43	31.11[b]	17.54[b]
C_2H_4	29.12	12.86	29.10	12.84	28.48[b]	12.21[b]
C_2H_6	30.43	4.66	30.43	4.67	30.10[b]	5.20[b]
C_6H_6	72.68	40.57	72.67	40.56	70.18[b]	37.93[d]

[a]For all molecules, experimental geometries are used. All values are in au. [b]Depolarized light scattering at λ = 632.8 nm (ref 78). [c]Depolarized ratio from Rayleigh scattering at λ = 514.5 nm (ref 80). [d]Depolarized light scattering at λ = 632.8 nm (ref 79).

demonstrates that the chosen EN2 solver is well suited for the nonsymmetric structure of the response equation system.

4. BENCHMARKS

To test the real computational complexity of the code, static polarizability calculations for linear alkane chains (C_nH_{2n+2}) with up to 100 carbon atoms were performed. In this case, the valence double-ζ plus polarization (DZVP)[55] basis set in combination with the GEN-A2 auxiliary function set[59] was used. The calculations were performed on a single Intel Xeon X5675 @ 3.07 GHz processor with a maximum of 2 GB of allocatable memory. Figure 3 shows the wall clock time spent in the full polarizability module of deMon2k when using the direct and iterative ADPT solver. The full polarizability module performs the following steps:

1. Calculates **G** and its inverse $\mathbf{G}^{-1}$.
2. Solves fitting equation system to obtain **x**.
3. Builds the perturbation vectors $\mathbf{b}^{(\lambda)}(\omega)$.
4. Solves the response equation system to obtain $\mathbf{x}^{(\lambda)}(\omega)$.
5. Calculates $\mathbf{z}^{(\lambda)}(\omega)$ using eq 5.
6. Builds the perturbed Kohn–Sham matrix $\mathbf{K}^{(\lambda)}(\omega)$ using eq 3.
7. Builds the perturbed density matrix $\mathbf{P}^{(\lambda)}(\omega)$ using eq 1.
8. Calculates the polarizability tensor elements as $\alpha_{\lambda\eta}(\omega) = \sum_{\mu,\nu} P^{(\lambda)}_{\mu\nu}(\omega)\langle\mu|r_\eta|\nu\rangle$.

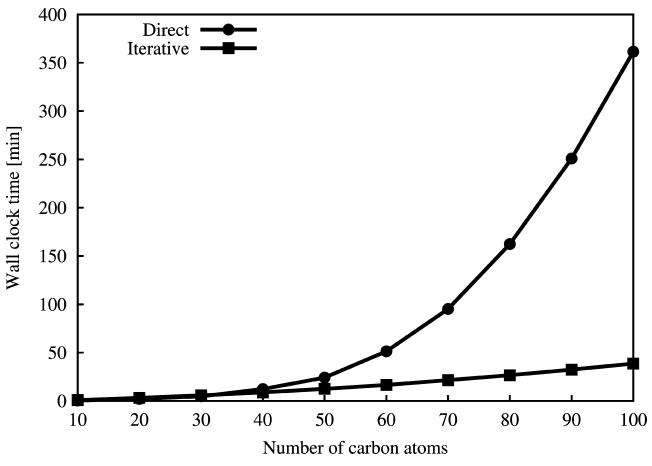


Figure 3. Wall clock time of ADPT polarizability benchmark calculations on linear alkane chains C_nH_{2n+2}.

The only difference between the direct and iterative ADPT solver arises from an intermediate step between 3 and 4, where the direct solver explicitly builds $\mathbf{A}(\omega)$ and **F** to ensemble and invert $\mathbf{R}(\omega)$, whereas the iterative solver employs the EN2 algorithm described above. As seen in Figure 3, the direct and iterative solvers show similar performance until the 30 carbon atoms chain. However, for larger chains, the $O(N^4)$ step rapidly dominates the direct approach, and the iterative solver becomes computationally advantageous. For $C_{100}H_{202}$, the iterative solver is already 1 order of magnitude faster than the direct one. Although the formal complexity of the iterative ADPT solver is $O(N^3)$, the scaling observed for this benchmark is subquadratic, namely, $N^{1.6}$. A more detailed view of the timings are shown in Table 3, where individual times for the four principal components of the iterative ADPT solver are given. Note that the $O(N^3)$ matrix multiplications, involving the molecular orbital coefficients, are buried deep down the list (column labeled as **T**) and that the calculation and contraction of ERIs is the most time-demanding step (columns **Q** and $\mathbf{a}(\omega)$). Even though the full polarizability module also has other significant contributions mainly arising from step 1, the $N^{1.6}$ scaling is a consequence of the subquadratic ERI computation and contraction. If the inversion of the **G** matrix in step 1, which involves a singular value decomposition, is taken out of the computational timings, the scaling of this benchmark is reduced to $N^{1.3}$. One possibility to avoid the $O(N^3)$ inversion of **G** is to use the Coulomb fitting solver implemented in deMon2k[35] for the step $\mathbf{f}' = \mathbf{G}^{-1}\mathbf{f}$, which is equivalent to solving the linear equation system $\mathbf{G}\mathbf{f}' = \mathbf{f}$. This possibility has yet to be explored in terms of accuracy and stability for the iterative ADPT solver. It is worth mentioning that the full serial polarizability module for the $C_{100}H_{202}$ alkane chain takes only 40 min with the iterative approach. Moreover, the iterative ADPT solver does not rely on special sparse linear algebra routines; therefore, sparsity does not have to be enforced.

Table 3. Timings for the Iterative ADPT Solver in Linear Alkane Chains[a]

molecule	**f**	**Q**	**T**	$\mathbf{a}(\omega)$	total
$C_{10}H_{22}$	9.2	17.6	0.1	21.4	47.9
$C_{20}H_{42}$	25.8	67.1	0.5	80.5	172.9
$C_{30}H_{62}$	41.5	125.9	1.6	150.5	318.3
$C_{40}H_{82}$	56.3	194.7	3.7	230.3	484.1
$C_{50}H_{102}$	73.2	273.1	7.0	324.1	677.0
$C_{60}H_{122}$	88.8	361.5	11.9	427.6	890.1
$C_{70}H_{142}$	108.1	464.2	18.6	547.2	1139.6
$C_{80}H_{162}$	120.8	575.2	27.8	674.5	1401.3
$C_{90}H_{182}$	137.7	696.1	39.0	812.8	1690.4
$C_{100}H_{202}$	153.4	824.5	52.8	964.5	2002.3

[a]All timing values are in seconds (s). The VWN/GEN-A2/DZVP level of theory was used. Column **f** corresponds to eq 10. Column **Q** corresponds to eq 14. Column **T** corresponds to eqs 16–18. Column $\mathbf{a}(\omega)$ corresponds to eq 19.

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493–1500

Another advantage of the iterative ADPT solver is the low memory demand compared to that of the direct solver. The memory needed for the iterative solver is almost the same as the that needed during the SCF in deMon2k. This makes it possible to perform parallel polarizability computations in architectures with moderate amounts of memory. To test the scalability of the parallel version of the algorithm 60, 72, 84, 96, and 108 Intel Xeon E5649 @ 2.53 GHz processors are used with 2 GB of maximum allocatable memory per core. The test system is the C_{720} fullerene optimized at the VWN/DZVP/GEN-A2 level of theory. This large fullerene, with more than 10,000 basis functions and almost 25,000 auxiliary functions, was chosen to demonstrate the low-memory demand of the iterative ADPT solver. In fact, the direct solver cannot perform this calculation due to memory requirements. Figure 4 shows

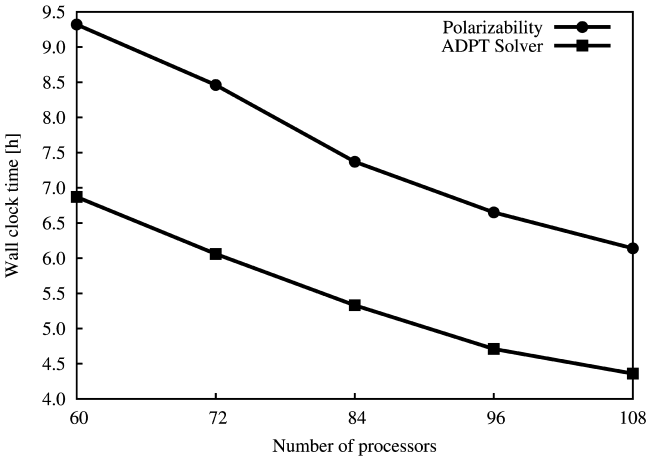


Figure 4. Scalability of the parallel deMon2k polarizability module and the ADPT iterative solver for the polarizability calculation of the C_{720} fullerene.

the wall clock timings, in hours, for the full polarizability module and the iterative ADPT solver alone. As always, no simplification due to molecular symmetry constraints were employed. Note that the total time to calculate the static polarizability of C_{720} is less than 10 h once the SCF has converged. In fact, with 108 processors, the total calculation time, including the SCF and the polarizability module, is about 10 h!

In summary, the iterative ADPT solver is well suited for serial as well as parallel calculations, reduces the formal scaling to $O(N^3)$, and achieves subquadratic behavior for systems with a few hundred atoms. As a consequence, the ADPT iterative solver allows the computation of static and dynamic polarizabilities of systems with hundreds of atoms and more than 10,000 basis functions in just a few hours.

5. APPLICATION

To demonstrate the potential of the iterative ADPT solver, static polarizabilities of fullerenes from C_{60} to C_{960} are calculated.

The VWN/DZVP/GEN-A2 methodology is selected for the giant fullerenes due to is good performance to accuracy ratio.[25,64] With this basis and auxiliary function set combination, the largest system contains 14,400 basis functions and more than 32,000 auxiliary functions. In Table 4, we list the ADPT static polarizabilities for these systems and compare

Table 4. Comparison of the C_{60} and Giant Fullerene ADPT Static Average Polarizabilities with Other Methodologies[a]

molecule	PPP[b]	SOS−RPA[c]	PBE0/SVPD[d]	VWN/DZVP[e]	VWN/DZVP[f]
C_{60}	526.4	533.1	551.1	529.0	528.4
C_{180}	1403.7	2024.5	2014.0		2014.5
C_{240}	2017.8	2915.3	2017.8	2902.3	2901.6
C_{540}	6222.0	7794.3	8395.4	8464.2	8466.0
C_{720}	9494.0	12470.9	12342.9		12489.3
C_{960}		18524.2			18538.4

[a]All ADPT computations were performed at the VWN/DZVP/GEN-A2 level of theory with fully optimized geometries. [b]From ref 65.

[c]From ref 66.

[d]From ref 67.

[e]Direct ADPT solver from ref 64.

[f]This work.

them with corresponding values from other methodologies.[65−67] In our calculations, all fullerene structures are fully optimized without symmetry restrictions at the same level of theory. The semiempirical PPP approach severely underestimates the polarizability of almost all of the fullerenes in the series. It is interesting to note that the polarizability of C_{240} from ref 67 is very small and, in fact, equal to the underestimated PPP polarizability. This is the reason for the artificial minimum in the polarizability per atom reported in ref 67. Our results, and those from the sum-over-states with random-phase approximation correction from ref 66, do not show such a minimum. Figure 5 shows the polarizability per

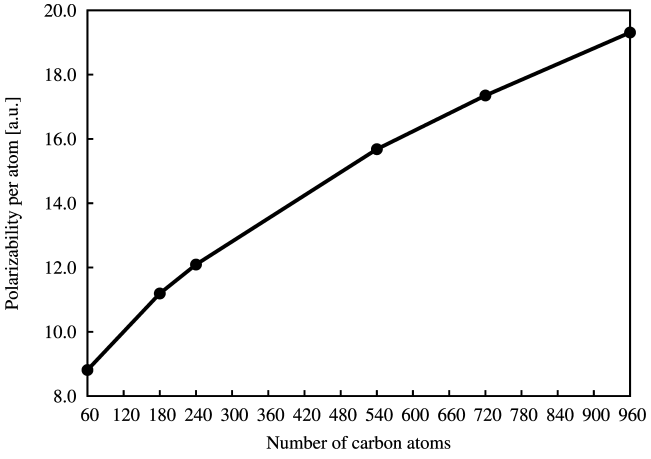


Figure 5. Polarizability per carbon atom for icosahedral fullerenes from C_{60} to C_{960} at the VWN/DZVP/GEN-A2 level of theory.

carbon atom for all studied fullerenes. The polarizability per atom grows monotonically, although it seems to be a reduction of the slope at the end of the curve in line with the reduction of the slope in a plot of the fullerene radii. The polarizability anisotropies of the fullerenes studied here must vanish by symmetry. In fact, we find that the calculated polarizability anisotropies are very small (<2.0 au). Moreover, the anisotropies from the iterative solutions are always smaller than the ones from the direct solutions previously reported in the literature.[64] This further confirms the enhanced stability of the iterative solver.

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493−1500

Dispersion of the polarizability of C_{60} was calculated and compared to the available experimental results.[68−72] Figure 6

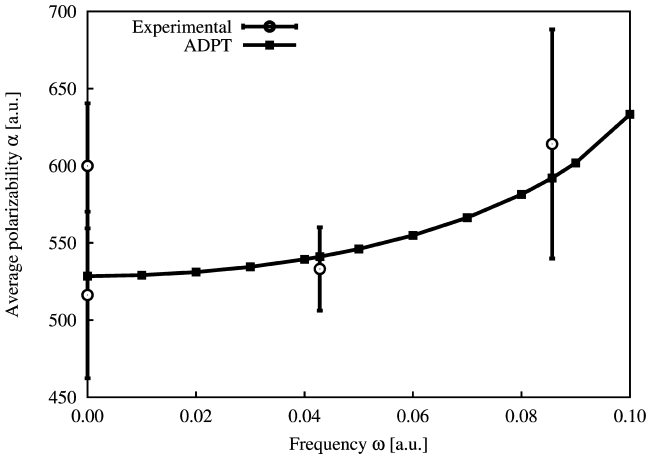


Figure 6. Comparison between the dispersion of the polarizability of the C_{60} fullerene calculated at the VWN/DZVP/GEN-A2 level of theory and experimental results.

shows that ADPT is in very good agreement with these experimental results. The most energetic wavelength used was 532 nm. This has enough energy to promote (forbidden) electronic excitations that could potentially produce numerical noise within the ADPT iterative solver. However, note that the calculated polarizability dispersion curve is smooth, which indicates that the iterative solver is quite stable even for situations where the frequency is close to an excitation (i.e., when the denominator $(\epsilon_i - \epsilon_a)^2 - \omega^2$ appearing in the formation of the vector $\mathbf{a}(\omega)$ is close to zero. For situations like these, the iterative solver could take additional steps to converge but in our experience never reaches the 15 step restart limit.

6. CONCLUSIONS

In this paper, we propose an iterative solver for the time-dependent auxiliary density perturbation theory response equation system. The solver is based on the Eirola−Nevanlinna algorithm for nonsymmetric linear equation systems. The new algorithm for solving the response equation system reduces the formal scaling from $O(N^4)$ to $O(N^3)$. The calculated scaling for linear alkane chains, from $C_{10}H_{22}$ to $C_{100}H_{202}$, is $N^{1.6}$. This diminished scaling is possible because the $O(N^3)$ steps (i.e., matrix−matrix multiplications) are calculated using the very efficient Intel MKL BLAS routines, and the corresponding timings were overshadowed by other lower scaling steps. Furthermore, electron repulsion integral screening and double asymptotic expansion techniques allowed for very efficient ERI calculations and contractions. The reduction in the computational complexity of the new algorithm produces speeds up to 1 order of magnitude faster for the calculation of the all-electron static polarizabilities of systems with more than 100 atoms. The polarizabilities calculated with the new algorithm are indistinguishable from those obtained by the analytic solution of the response equation system in auxiliary density perturbation theory. This new iterative solver allowed us to calculate static polarizabilities of carbon fullerenes from C_{60} to C_{960} within a day. The electric properties were calculated at the optimized geometries without symmetry restrictions. The static polarizabilities obtained here confirm a monotonical increase in the polarizability per atom up to C_{960}. Adaption of the iterative solver presented here for other response properties is currently under development in our laboratory.

AUTHOR INFORMATION

Corresponding Author

*E-mail: dmejia@cinvestav.mx.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was financially supported by the CONACyT projects CB-179409 and U130726. D.M.R. and R.D.V. gratefully acknowledge CONACyT Ph.D. fellowships 47922 and 47383, respectively. C_{720} benchmark calculations were performed on the WESTGRID of Compute Canada.

REFERENCES

(1) Hohenberg, P.; Kohn, W. *Phys. Rev.* **1964**, *136*, B864−B871.
(2) Kohn, W.; Sham, L. J. *Phys. Rev.* **1965**, *140*, A1133−A1138.
(3) Fournier, R. *J. Chem. Phys.* **1990**, *92*, 5422−5429.
(4) Komornicki, A.; Fitzgerald, G. *J. Chem. Phys.* **1993**, *98*, 1398−1421.
(5) Colwell, S. M.; Murray, C. W.; Handy, N. C.; Amos, R. D. *Chem. Phys. Lett.* **1993**, *210*, 261−268.
(6) Lee, A. M.; Colwell, S. M. *J. Chem. Phys.* **1994**, *101*, 9704−9709.
(7) Ochsenfeld, C.; Head-Gordon, M. *Chem. Phys. Lett.* **1997**, *270*, 399−405.
(8) Weber, V.; Niklasson, A. M. N.; Challacombe, M. *Phys. Rev. Lett.* **2004**, *92*, 193002.
(9) Weber, V.; Niklasson, A. M.; Challacombe, M. *J. Chem. Phys.* **2005**, *123*, 044106.
(10) Niklasson, A. M.; Weber, V. *J. Chem. Phys.* **2007**, *127*, 064105.
(11) Kussmann, J.; Ochsenfeld, C. *J. Chem. Phys.* **2007**, *127*, 204103.
(12) Coriani, S.; Høst, S.; Jansik, B.; Tøgersen, L.; Olsen, J.; Jørgensen, P.; Reine, S.; Pawłowski, F.; Helgaker, T.; Sałek, P. *J. Chem. Phys.* **2007**, *126*, 154108.
(13) Beer, M.; Ochsenfeld, C. *J. Chem. Phys.* **2008**, *128*, 221102.
(14) Kjærgaard, T.; Jørgensen, P.; Olsen, J.; Coriani, S.; Helgaker, T. *J. Chem. Phys.* **2008**, *129*, 154106.
(15) Kobayashi, M.; Touma, T.; Nakai, H. *J. Chem. Phys.* **2012**, *136*, 084108.
(16) Izmaylov, A. F.; Brothers, E. N.; Scuseria, G. E. *J. Chem. Phys.* **2006**, *125*, 224105.
(17) Helgaker, T.; Coriani, S.; Jørgensen, P.; Kristensen, K.; Olsen, J.; Ruud, K. *Chem. Rev.* **2012**, *112*, 543−631.
(18) Kussmann, J.; Beer, M.; Ochsenfeld, C. *WIREs Comput. Mol. Sci.* **2013**, *3*, 614−636.
(19) Sophy, K. B.; Pal, S. *J. Chem. Phys.* **2003**, *118*, 10861−10866.
(20) Sophy, K. B.; Pal, S. *J. Mol. Struct.: THEOCHEM* **2004**, *676*, 89−95.
(21) Sophy, K. B.; Calaminici, P.; Pal, S. *J. Chem. Theory Comput.* **2007**, *3*, 716−727.
(22) Sophy, K. B.; Shedge, S. V.; Pal, S. *J. Phys. Chem. A* **2008**, *112*, 11266−11272.
(23) Shedge, S. V.; Carmona-Espíndola, J.; Pal, S.; Köster, A. M. *J. Phys. Chem. A* **2010**, *114*, 2357−2364.
(24) Shedge, S. V.; Pal, S.; Köster, A. M. *Chem. Phys. Lett.* **2011**, *510*, 185−190.
(25) Flores-Moreno, R.; Köster, A. M. *J. Chem. Phys.* **2008**, *128*, 134105.
(26) Carmona-Espíndola, J.; Flores-Moreno, R.; Köster, A. M. *J. Chem. Phys.* **2010**, *133*, 084102.
(27) Dunlap, B. I.; Schweigert, I. V. *J. Chem. Phys.* **2011**, *134*, 044122.
(28) Köster, A. M.; Reveles, J. U.; del Campo, J. M. *J. Chem. Phys.* **2004**, *121*, 3417−3424.

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493−1500

(29) Dunlap, B. I.; Conolly, J. W. D.; Sabin, J. R. *J. Chem. Phys.* **1979**, *71*, 4993−4999.
(30) Mintmire, J. W.; Dunlap, B. I. *Phys. Rev. A* **1982**, *25*, 88−95.
(31) Carmona-Espíndola, J.; Flores-Moreno, R.; Köster, A. M. *Int. J. Quantum Chem.* **2012**, *112*, 3461−3471.
(32) Flores-Moreno, R.; Melin, J.; Ortiz, J. V.; Merino, G. *J. Chem. Phys.* **2008**, *129*, 224105.
(33) Flores-Moreno, R. *J. Chem. Theory Comput.* **2010**, *6*, 48−54.
(34) Köster, A. M.; Geudtner, G.; Calaminici, P.; Casida, M. E.; Dominguez, V. D.; Flores-Moreno, R.; Gamboa, G. U.; Goursot, A.; Heine, T.; Ipatov, A.; Janetzko, F.; del Campo, J. M.; Reveles, J. U.; Vela, A.; Zuniga-Gutierrez, B.; Salahub, D. R. *deMon2k*, version 3; The deMon Developers: Cinvestav, México, D.F., 2011.
(35) Domínguez-Soria, V. D.; Geudtner, G.; Morales, J. L.; Calaminici, P.; Köster, A. M. *J. Chem. Phys.* **2009**, *131*, 124102.
(36) van der Vorst, H. A. In *SOFSEM'96: Theory and Practice of Informatics*; Jeffery, K. G., Král, J., Bartošek, M., Eds.; Lecture Notes in Computer Science; Springer: Berlin, Germany, 1996; Vol. *1175*, pp 217−234.
(37) van der Vorst, H. A. *SIAM J. Sci. Stat. Comp.* **1992**, *13*, 631−644.
(38) Sonneveld, P. *SIAM J. Sci. Stat. Comp.* **1989**, *10*, 36−52.
(39) Freund, R. W.; Nachtigal, N. M. *Numerische Mathematik* **1991**, *60*, 315−339.
(40) Saad, Y.; Schultz, M. H. *SIAM J. Sci. Stat. Comp.* **1986**, *7*, 856−869.
(41) Eirola, T.; Nevanlinna, O. *Linear Algebra and its Applications* **1989**, *121*, 511−520.
(42) Yang, U. M.; Gallivan, K. A. *Applied Numerical Mathematics* **1995**, *19*, 287−317.
(43) Zlatev, Z.; Georgiev, K. *Central European Journal of Mathematics* **2013**, *11*, 1510−1530.
(44) McWeeny, R. *Methods of Molecular Quantum Mechanics*, 2nd ed.; Academic Press: London, 1992.
(45) KÃster, A. M. *J. Chem. Phys.* **2003**, *118*, 9943.
(46) Alvarez-Ibarra, A.; Köster, A. M. *J. Chem. Phys.* **2013**, *139*, 024102.
(47) Lawson, C. L.; Hanson, R. J.; Kincaid, D.; Krogh, F. T. *ACM Trans. Math. Soft.* **1979**, *5*, 308−323.
(48) Dongarra, J.; Du Croz, J.; Hammarling, S.; Hanson, R. J. *ACM Trans. Math. Soft.* **1988**, *14*, 1−17.
(49) Dongarra, J.; Du Croz, J.; Hammarling, S.; Hanson, R. J. *ACM Trans. Math. Soft.* **1988**, *14*, 18−32.
(50) Dongarra, J.; Du Croz, J.; Hammarling, S. *ACM Trans. Math. Soft.* **1990**, *16*, 1−17.
(51) Dongarra, J.; Du Croz, J.; Hammarling, S. *ACM Trans. Math. Soft.* **1990**, *16*, 18−28.
(52) Vuik, C.; van der Vorst, H. A. *Linear Algebra and its Applications* **1992**, *160*, 131−162.
(53) Gao, W.; Xue, J.; Qu, Y. *Applied Mathematics and Computation* **1999**, *98*, 199−208.
(54) Lide, D. R. *CRC Handbook of Chemistry and Physics*; CRC Press: Boca Raton, FL, 2005; Chapter 9, pp 15−44.
(55) Godbout, N.; Salahub, D. R.; Andzelm, J.; Wimmer, E. *Can. J. Phys.* **1992**, *70*, 560−571.
(56) Zeiss, G. D.; Scott, W. R.; Suzuki, N.; Chong, D. P.; Langhoff, S. R. *Mol. Phys.* **1979**, *37*, 1543−1572.
(57) Calaminici, P.; Jug, K.; Köster, A. M. *J. Chem. Phys.* **1998**, *109*, 7756−7763.
(58) Calaminici, P.; Jug, K.; Köster, A. M.; Ingamells, V. E.; Papadopoulos, M. G. *J. Chem. Phys.* **2000**, *112*, 6301−6308.
(59) Calaminici, P.; Janetzko, F.; Köster, A. M.; Mejia-Olvera, R.; Zuniga-Gutierrez, B. *J. Chem. Phys.* **2007**, *126*, 044108.
(60) Dirac, P. A. M. *Proc. Cambridge Philos. R. Soc.* **1930**, *26*, 376−385.
(61) Vosko, S. H.; Wilk, L.; Nusair, M. *Can. J. Phys.* **1980**, *58*, 1200−1211.
(62) Krack, M.; Köster, A. M. *J. Chem. Phys.* **1998**, *108*, 3226−3234.
(63) Köster, A. M.; Flores-Moreno, R.; Reveles, J. U. *J. Chem. Phys.* **2004**, *121*, 681−690.
(64) Calaminici, P.; Carmona-Espíndola, J.; Geudtner, G.; Köster, A. M. *Int. J. Quantum Chem.* **2012**, *112*, 3252−3255.
(65) Ruiz, A.; Bretón, J.; Gomez Llorente, J. M. *J. Chem. Phys.* **2001**, *114*, 1272−1277.
(66) Zope, R. R.; Baruah, T.; Pederson, M. R.; Dunlap, B. I. *Phys. Rev. B* **2008**, *77*, 115452.
(67) Rappoport, D.; Furche, F. *J. Chem. Phys.* **2010**, *133*, 134105.
(68) Antoine, R.; Dugourd, P.; Rayane, D.; Benichou, E.; Broyer, M.; Chanderzon, F.; Guet, C. *J. Chem. Phys.* **1999**, *110*, 9771−9772.
(69) Ballard, A.; Bonin, K.; Louderback, J. *J. Chem. Phys.* **2000**, *113*, 5732−5735.
(70) Berninger, M.; Stefanov, A.; Deachapunya, S.; Arndt, M. *Phys. Rev. A* **2007**, *76*, 013607.
(71) Hackermüller, L.; Hornberger, K.; Gerlich, S.; Gring, M.; Ulbricht, H. *Appl. Phys. B: Lasers Opt.* **2007**, *89*, 469−473.
(72) Compagnon, I.; Antoine, R.; Broyer, M.; Dugourd, P.; Lermé, J.; Rayane, D. *Phys. Rev. A* **2001**, *64*, 025201.
(73) Perkins, A. J. *J. Phys. Chem.* **1964**, *68*, 654−655.
(74) Bishop, D. M.; Cheung, L. M. *J. Phys. Chem. Ref. Data* **1982**, *11*, 119.
(75) Spackman, M. A. *J. Phys. Chem.* **1989**, *93*, 7594−7603.
(76) Alms, G. R.; Burnham, A. K.; Flygare, W. H. *J. Chem. Phys.* **1975**, *63*, 3321−3326.
(77) Miller, C. K.; Orr, B. J.; Ward, J. F. *J. Chem. Phys.* **1981**, *74*, 4858−4871.
(78) Bridge, N. J.; Buckingham, A. D. *Proc. R. Soc. London, Ser. A* **1966**, *295*, 334−349.
(79) Bogaard, M. P.; Buckingham, A. D.; Pierens, R. K.; White, A. H. *J. Chem. Soc., Faraday Trans. 1* **1978**, *74*, 3008−3015.
(80) Murphy, W. F. *J. Chem. Phys.* **1977**, *67*, 5877−5882.
(81) Bogaard, M. P.; Buckingham, A. D.; Ritchie, G. L. D. *Chem. Phys. Lett.* **1982**, *90*, 183−190.

DOI: 10.1021/ct501065g
J. Chem. Theory Comput. 2015, 11, 1493−1500