

# Implementation and Use of a Direct, Partially Integral-Driven Non-Dyson Propagator Method for Molecular Ionization

LORIANO STORCHI,<sup>1,2</sup> GIUSEPPE VITILLARO,<sup>2</sup> FRANCESCO TARANTELLI<sup>1,3</sup>

<sup>1</sup>*Department of Chemistry, University of Perugia, via Elce di Sotto, 8, I-06123 Perugia, Italy*

<sup>2</sup>*Istituto di Scienze e Tecnologie Molecolari, CNR, via Elce di Sotto, 8, I-06123 Perugia, Italy*

<sup>3</sup>*C.N.R. Istituto di Scienze e Tecnologie Molecolari, via Elce di Sotto, 8, I-06123 Perugia, Italy*

*Received 19 May 2008; Revised 7 July 2008; Accepted 17 July 2008*

*DOI 10.1002/jcc.21104*

*Published online in Wiley InterScience (www.interscience.wiley.com).*

**Abstract:** The Green's function ADC(3) scheme has been for many years a successful method to predict theoretically the ionization (and electron affinity) spectrum of molecules. However, a dramatic enhancement of the method's power has come only recently, with the development of an approximation method to the one-particle Green's function which does not make direct use of the Dyson equation. In the present work, we present an efficient computer implementation of this novel approach, with first comparative tests demonstrating its enormous computational advantage over the conventional approach.

© 2008 Wiley Periodicals, Inc. J Comput Chem 00: 000–000, 2008

**Key words:** Green's functions; ADC; integral-driven; single ionization; non-Dyson

## Introduction

The Green's function formalism<sup>1–4</sup> constitutes a powerful and elegant theoretical tool for investigating properties and excitation processes in many-particle systems. Green's functions provide direct access to important physical quantities such as, for example, ionization energies and spectral intensities,<sup>5</sup> without the need to resort to separate approximate solutions of the Schrödinger equation for the ground state and the ionic states. The method thus provides one-step access to vast portions of the ionization/excitation spectra of molecules and incorporates implicitly from the outset a balanced account of both the ground and ionic correlation, which is difficult to achieve with conventional wavefunction approaches. Another inherent advantage of the Green's function approach, essential for the treatment of large systems, is the natural provision of size-consistent approximation schemes, displaying the correct scaling behavior with respect to the number of electrons.

In the case of the one-particle Green's function, one may resort to the well-known Dyson equation relating the Green's function to the so-called self-energy, which is an effective energy-dependent one-particle potential. The solution of the Dyson equation can be cast as an Hermitian matrix eigenvalue problem.<sup>6</sup> In practical applications one, of course, uses an approximation of the self-energy and hence of this Hermitian matrix. Various approximation schemes have been proposed over many years to evaluate the self-energy and Green's func-

tions in general (see for example refs. 7–10). Important class of approximation schemes are the diagrammatic methods. Here, one makes use of the Feynman diagrams to represent the perturbation series of the Green's function or propagator under consideration. Among the diagrammatic methods the Algebraic Diagrammatic Construction (ADC)<sup>6,11–14</sup> has proven to be of particular success for the treatment of finite electronic systems. This scheme, which provides access to the entire energy range of the valence-shell ionization, reformulates the diagrammatic perturbation expansion for the Green's function in a simple algebraic form, representing infinite partial summations of certain types of Feynman diagrams. The  $n$ th-order scheme, ADC( $n$ ), is complete through order  $n$  perturbation theory, i.e., it includes all Feynman diagrams up to  $n$ th order as well as higher order contributions in an appropriate manner. The method is quite general and applies to any Green's function or single component of it.

The essential numerical tasks associated with the computation of the Green's function in the ADC scheme involve the evaluation of matrix elements and the diagonalization of Hermitian matrices defined in the space of a special class of ionic configurations. In realistic applications, the source of problems is that one has to cope with the size of the configuration space. Depending on the size of the molecule, the orbital basis set, and the approximation scheme used, the configuration space can

*Correspondence to:* L. Storchi; e-mail: redo@thch.unipg.it

become extremely large preventing the determination of the relevant eigenvalues and eigenvectors from the corresponding secular matrix with reasonable expense. It is the purpose of the present article to describe how, building on some recent significant advances in the formal development of the theory, the so-called non-Dyson scheme,<sup>15,16</sup> it is possible to efficiently minimize the computational burden of Green's function investigations. To this end, we have implemented and illustrate here a new computer program for the calculation of ionization spectra that (a) makes use of the non-Dyson scheme and (b) adopts a partially integral-driven approach for the handling of the two-electron integrals. Preliminarily, in the next two sections, we recall the principal aspects of the third-order ADC scheme and its non-Dyson version.

### The ADC(3) Scheme

In the ADC scheme, the ionic configurations over which the Dyson equation is represented are classified in particle-hole notation with reference to the Hartree-Fock neutral ground state. We shall particularly concentrate hereafter on the third order, ADC(3), approximation,<sup>6,17,18</sup> but most considerations apply equally well to higher order schemes. In ADC(3) four classes of ionic configurations appear: one-hole ( $1h$ ) and two-hole-one-particle ( $2h1p$ ) for the  $N - 1$ -electron configurations, one-particle ( $1p$ ) and two-particle-one-hole ( $2p1h$ ) for the  $N + 1$ -electron configurations. The secular matrix is conveniently partitioned as shown in Figure 1 and its matrix elements are reported and discussed at length in, for example, ref. 17. The upper-left block (main block) represents the interaction over the  $1p$  and  $1h$  configurations and includes the energy-independent part of the self-energy,  $\Sigma(\infty)$ . This block is further coupled to the  $2p1h$  space (through the  $U^I$  block) and to the  $2h1p$  space (through the  $U^{II}$  block). The central diagonal block of the matrix,  $K^I + C^I$ , is over the  $2p1h$  space and the lower right block,  $K^{II} + C^{II}$ , over the  $2h1p$  space. The remaining blocks of the matrix vanish.

One peculiar and important feature of this representation of the Dyson equation, is the coupling between the  $N - 1$ -particle (ionization) and  $N + 1$ -particle (electron affinity) configurations. This coupling may not be weak for the low-lying poles of the Green's function and cannot be neglected, even when one is only interested, as is most usually the case, to the sole ionization spectrum. As a result, especially when using large basis sets necessary to obtain accurate results, the secular problem is largely dominated by the presence of the  $K^I + C^I$  block of the matrix. In general, and even more when higher excited configurations are required for higher-order approximation schemes, it is the explicit presence of the excited electron affinity space that induces the greatest computational burden, both in the matrix construction and, especially, in its iterative diagonalization. This is the sole aspect of ADC(3), otherwise presenting a matrix size growing linearly with the basis set and only quadratically with the number of electrons, which inhibits its applicability to large molecular systems. Years ago, a particularly simple and effective approach was presented<sup>18</sup> to approximately decouple the large  $N + 1$ -particle block from the rest of the matrix, by transforming it through a small number of block-Lanczos iterations. This "trick," however, still requires the explicit construction and multiplication of the affinity block. This block, furthermore, is

	1p/1h	2p1h	2h1p
1p/1h	$\varepsilon + \Sigma(\varepsilon)$	$(U^I)^+$	$(U^{II})^+$
2p1h	$U^I$	$K^I + C^I$	0
2h1p	$U^{II}$	0	$K^{II} + C^{II}$

**Figure 1.** Structure of the hermitian ADC matrix in the Dyson ADC(3) scheme.

in the Dyson approach needed to compute the static self-energy.<sup>13</sup> It would be highly desirable, therefore, if, starting from the standard ADC, one could devise a formalism in which the ionization and affinity block are analytically decoupled from the start, so that the evaluation of the affinity block can be altogether skipped for ionization calculations. The recently developed non-Dyson ADC(3) approach<sup>15,16,19</sup> (nD-ADC(3)) successfully addresses exactly this question.

By applying the algebraic diagrammatic construction scheme directly and separately to the advanced and retarded parts of the Green's function, rather than to the self-energy part, the novel approach sidesteps the Dyson equation and exploits *a priori* the natural decoupling of the  $N - 1$ - and  $N + 1$ -particle spaces. Very conveniently, moreover, the resulting ADC(3) secular problem turns out to be extremely similar to the standard one, deprived of the  $N + 1$ -electron configurations, at the price of course of additional terms in the matrix elements. But as we shall recall shortly, these additional terms only affect the small main block of the ADC matrix.

The non-Dyson scheme thus achieves the uniquely effective goal of providing access to theoretical ionization spectra of third order accuracy in a configuration space essentially comprising just the single excitations of the hole configurations. This makes computationally much easier, in terms of both speed and number of required iterations, the calculation of the desired ionization poles by iterative diagonalization procedures such as the block-Lanczos.<sup>18</sup> But it is very important to realize that this happens not just because the size of the ADC matrix is so drastically reduced, but also because of the consequent alteration in the structure of the eigenvalue spectrum, as we now briefly clarify.

The Lanczos method, as well as its block-Lanczos variant or preconditioned methods such as the (block-) Davidson one, are

well known iterative diagonalization methods.<sup>20–23</sup> As general subspace iteration approaches, they can in principle be used to obtain the full spectrum of a given large matrix, but in fact they have been designed and are tuned to obtain a subset of the whole eigenvalue spectrum. In the case of inner-valence ionization calculations, this useful subset rapidly becomes quite large, easily reaching hundreds if not thousands of solutions for moderately sized molecules. As the number of iterations performed increases and the size of the vector subspace grows, the approximate eigenvectors<sup>22,23</sup> converge to exact ones at different speeds: they typically converge faster the closer they lie to the outer edges of the spectrum, so that extreme eigenvalues tend to converge first (i.e., after fewer iterations) and inner ones converge last. This general trend can only to a limited extent be altered by a suitable choice of initial vectors. This represents an additional problem in the Dyson ADC approach to the ionization spectrum: the lowest ionization poles, which are typically the ones we are most interested in, because of the presence of the electron-affinity poles, lie more or less exactly in the middle of the spectrum and thus tend to converge poorly and to require a large number of iterations. Even when the affinity spectrum is reduced in size by the block-Lanczos prediagonalization,<sup>18</sup> there is a substantial number of eigenvalues which represent (inaccurate) affinity poles and thus are algebraically smaller than the onset of the ionization spectrum. By contrast, in the nD-ADC problem, the complete elimination of the affinity space brings the lowest ionization poles at the edge of the spectrum and thus significantly accelerates their convergence, well apart from and beyond the reduction of the secular problem size.

### nD-ADC(3) Working Equations

Because in the normal ADC(3) matrix the  $2h1p$  and  $2p1h$  spaces are coupled only indirectly to each other through the main ( $1h - 1p$ ) block (see Fig. 1), it turns out that the only differences between the Dyson and non-Dyson schemes lie in the expression of the main block of matrix elements between  $1h$  configurations. In particular, in the nD-ADC(3) matrix elements, additional second and third order terms appear. In fact, as discussed by Schirmer et al.,<sup>15</sup> the ADC equations are not unique, and they have been reported in two different formulations.<sup>15,16</sup> The two versions lead to the same results to within sixth-order terms and extensive numerical comparison shows that the resulting ionization energies never differ by more than 0.01 eV.<sup>16</sup> We have implemented in our code both sets of formulas (which are computationally essentially equivalent) but, for simplicity, we report and discuss here only the set of equations of ref. 15. As for the general ADC(3) case, the nD-ADC formulas refer to a system having a nondegenerate neutral ground state, which we assume to have as usual a closed-shell Hartree-Fock representation. The ADC problem is, of course, based on a spin-free Hamiltonian and thus factorizes when expressed over spin-adapted configurations. Since the matrix elements involve just  $1h$  configurations, we need only consider that some integrals vanish because of spin. Taking this into account, it is immediate to cast the spin-orbital equations of ref. 15 in spin-free form over doublet spin-adapted configurations involving only spatial orbitals. For the remaining spin-adapted formulas, which we use in our imple-

mentation, see ref. 17. We use real orbitals with indices  $i, j, k, l, \dots$  to denote holes and  $a, b, c, d, \dots$  for particles, which indices  $p, q, r, \dots$  refer generically to either.  $(pq|rs)$  denotes a two-electron repulsion integral in “charge-cloud” notation,  $\epsilon_p$  are the orbital energies and  $\epsilon_{pqrs} = \epsilon_p + \epsilon_q - \epsilon_r - \epsilon_s$ . The second order non-Dyson term of a matrix element between configurations with holes in orbital  $i$  and  $j$  thus reads

$$C_{ij}^{(2)} = \frac{1}{2} \sum_{a,b,k} (ai|bk)((ak|bj) - 2(aj|bk)) \frac{\epsilon_{abki} + \epsilon_{abkj}}{\epsilon_{abki}\epsilon_{abkj}} \quad (1)$$

The non-Dyson third order contribution is conveniently divided in four terms:

$$C_{ij}^{(3)} = C_{ij}^{(3a)} + C_{ij}^{(3b)} + C_{ij}^{(3c)} + C_{ij}^{(3d)} \quad (2)$$

whose spin-free expressions are as follows:

$$C_{ij}^{(3a)} = \sum_{\substack{a,b,c,d \\ k}} (ab|cd)(ai|ck)(2(bj|dk) - (bk|dj)) \frac{1}{\epsilon_{acik}\epsilon_{bdjk}} \quad (3)$$

$$C_{ij}^{(3b)} = \sum_{\substack{a,b,c \\ k,l}} \{(bk|cl)((ak|bi) - 2(ai|bk))((al|cj) - 2(aj|cl)) + (bc|kl)[(aj|cl)((ak|bi) - 2(ai|bk)) + (al|cj)((ai|bk) - 2(ak|bi))]\} \frac{1}{\epsilon_{abik}\epsilon_{acjl}} \quad (4)$$

$$C_{ij}^{(3c)} = \sum_{\substack{a,b \\ k,l,m}} (al|bm)(ak|bj)(2(im|kl) - (il|km)) \frac{1}{\epsilon_{ablm}\epsilon_{abkj}} + (i \leftrightarrow j) \quad (5)$$

$$C_{ij}^{(3d)} = \sum_{\substack{a,b,c \\ k,l}} \{(ak|ci)((bj|cl) - 2(bl|cj))((al|bk) - 2(ak|bl)) + (ac|ik)[(bj|cl)((ak|bl) - 2(al|bk)) + (bl|cj)((al|bk) - 2(ak|bl))]\} \frac{1}{\epsilon_{abkl}\epsilon_{bcjl}} + (i \leftrightarrow j) \quad (6)$$

### Computational Aspects of the ND-ADC(3) Code

Even with the great advantages offered by the non-Dyson simplification, it is generally not practical to carry out a full diagonalization of the nD-ADC(3) matrix because of its dimensions, although it is often necessary to extract with sufficient accuracy

large portions of the spectrum, which may be extremely dense (as happens for example for inner valence ionization). Thus the solution of the ADC eigenvalue problem is a major computational burden of the whole procedure. For this, we resort to iterative methods such as Lanczos<sup>20,22,23</sup> Davidson<sup>21–23</sup> or, more appropriately, their block-variants.<sup>18,22,23</sup> The block-Lanczos method, in particular, is especially well-suited to extract quite effectively all the most useful spectral information from Green's function calculations.<sup>18,24,25</sup> It is one of a quite broad class of subspace iterations approaches,<sup>22,23</sup> attempting to produce some approximate eigenvalues using a projection of the whole matrix, in a subspace as small as is convenient. The subspace is iteratively enlarged until the desired eigensolutions are accurate enough. An invariable feature of all these methods is that, during the diagonalization procedure, the matrix must be used many times in matrix–vector or matrix–matrix multiplications, which are needed for the iterative refinement of an initial (trial) set of vectors. Indeed the most expensive computational task during the whole block-Lanczos procedure is this matrix multiplication.

It is in general not possible to store the whole ADC matrix in memory to use it during the numerous multiplication steps. Current hard-disk sizes make it a plausible approach to store it in such nonvolatile storage devices. However, this is generally inefficient because of the substantial I/O activity involved as the matrix must be read many times during the computation. A third possible method for the computation and handling of the ADC matrix, the direct approach, is therefore to be preferred. Similar approaches to the handling of large matrices and two-electron integrals have become standard in most modern quantum chemistry software. More specifically, the scheme we adopt essentially involves recomputing the large  $2h1p/2h1p$  block of the matrix whenever it is required, while the much smaller  $1h/1h$  and  $2h1p/1h$  blocks may safely be computed and stored once for all in memory or disk. Recomputing the latter two blocks is not necessary given their size and would be more difficult as they are non-sparse and their computation is more demanding. Recomputing the small  $\Sigma(\infty)$  matrix would be particularly inconvenient as it entails a costly iterative procedure.

### Integral-Driven Equations

The computational strategy we used to avoid memory and disk limitations, both in terms of size and access speed, is the adoption of an integral-driven procedure. Such approach is also well-established in conventional wave-function methods. The basic idea behind the integral-driven approach is simple: since it is in general not possible to store in main memory all the two-electron integrals and it would of course be very inefficient to access them randomly as required on disk, we must reformulate the equations so that they represent an iteration over the integrals rather than over matrix elements. In other words, we must be able to access the integrals (or subsets thereof) in any given order and add their contributions to the appropriate matrix elements, rather than exhaust each matrix element in turn by retrieving all the appropriate integral contributions. The integral-driven reformulation requires a somewhat lengthy reanalysis of the working equations but leads to a very efficient and open-ended algorithm, essentially free from storage limitations. For

the present implementation, we only applied the integral-driven approach to the handling of those matrix element contributions involving two-electron integrals over four virtual orbitals, which is by orders of magnitude the most numerous integral class. This enables very large-scale calculations without unduly complicating the code. This approach essentially involves the sole implementation of the non-Dyson contribution  $C_{ij}^{(3a)}$ , eq. (3). An entirely similar integral-driven scheme is already adopted in our conventional Dyson ADC(3) code for the one-particle Green's function.

In summary, the integral-driven procedure works as follows. We read all the integrals into memory, except the 4-virtual ones. The integrals in memory are sorted and indexed so that they can be accessed in any order and they are used in the standard configuration-driven iteration to evaluate the matrix element blocks. A set of 4-virtual integrals is then loaded from disk together with their indices to fill a specified memory area and each integral is processed in turn according to the formulas reported in Table 1, evaluating its contribution to any matrix element that requires it. When the current batch of integrals has been entirely used, we fill again the buffer with integrals from disk and proceed likewise, until all the integrals have been processed. To make the code simpler and more readable, the core of the integral-driven code is constituted by one routine which implements eq. (3) (except of the loop over virtual indices) and is called as many times as needed for each integral, with the appropriate set and permutation of argument indices. Note that this algorithm is easily parallelizable, e.g. in a “multithreading” computational environment. Finally, as mentioned earlier, we also wrote the code implementing the formulation of the nD-ADC(3) scheme presented in ref. 16. The results we obtained in both cases agree within the expected threshold for this kind of calculations, thus representing also an independent check of the correctness of the code.

There is a final very important issue concerning the nD-ADC approach which we have neglected in the foregoing discussion: the calculation of the static self-energy matrix  $\Sigma(\infty)$ . This remains an important contribution to the main block of the ADC matrix and the accuracy of its evaluation is known to affect non-negligibly the quality of the ADC results. Because of the separation of the  $N + 1$ -particle space, in the non-Dyson method we must still calculate the  $1h/1h$  part of  $\Sigma(\infty)$ . The standard Dyson Expansion Method (DEM) for the calculation of the static self-energy is an iterative approach discussed, e.g., in ref. 13. Its fundamental drawback is that, regardless of the particle number of the configurations involved, it requires the availability of both the ionization and affinity blocks of the  $\mathbf{K} + \mathbf{C}$  matrix. It is evident, therefore, that this method of calculation of  $\Sigma(\infty)$  has a significant negative impact on the efficacy of the non-Dyson approach and, in fact, may represent a computational bottleneck. It is clearly desirable to have a non-Dyson approximation method for the static self-energy which requires only matrix elements in the  $N - 1$ -electron space. Such a method, referred to as the  $\Sigma(4+)$  scheme, has indeed been proposed and shown to lead to an approximation essentially as good as the standard DEM approximation.<sup>16</sup> Since we are still in a phase of comparative study of the non-Dyson and Dyson approaches, we have not yet implemented the  $\Sigma(4+)$  scheme in our code, so that both methods rely on an identical evaluation of the static self-energy

**Table 1.** Integral Driven Formulas for the  $C^{(3a)}$  Matrix Elements.

Case	Index relations	Integrals	Contribution
1	$a = b = c = d$	$(aala)$	$\sum_i \frac{(ai al)(aj al)}{\epsilon_{aail} \epsilon_{aajl}}$
2	$a = b = c > d$	$(aalab), (aalba), (ablaaa), (bala)$	$\sum_i \left\{ \frac{(ai al)(aj bl) + (al bj)}{\epsilon_{aail} \epsilon_{abjl}} + \frac{(ai bl) + (al bi)(aj al)}{\epsilon_{abil} \epsilon_{aajl}} \right\}$
3	$a = b > c = d$	$(aalbb), (bb\Delta a)$	$\sum_i \left\{ \left( \frac{(ai bl)2(aj bl) - (al bj)}{\epsilon_{abil} \epsilon_{abjl}} \right) + (a \leftrightarrow b) \right\}$
4	$a = c > b = d$	$(ablab), (ablba), (balab), (balba)$	$\sum_i \left\{ \left( \frac{(ai al)(bj bl)}{\epsilon_{aail} \epsilon_{bbjl}} + \frac{(ai bl)2(al bj) - (aj bl)}{\epsilon_{abil} \epsilon_{abjl}} \right) + (a \leftrightarrow b) \right\}$
5	$a > b = c = d$	$(ablbb), (balbb), (bblab), (bblba)$	Same as Case 2, with $a \leftrightarrow b$ .
6	$a = b > c > d$	$(aalcd), (aaldc), (cd\Delta a), (dc\Delta a)$	$\sum_i \left\{ \left( \frac{(ai cl)2(aj dl) - (al dj)}{\epsilon_{acil} \epsilon_{adjl}} + \frac{(al ci)2(al dj) - (aj dl)}{\epsilon_{acil} \epsilon_{adjl}} \right) + (c \leftrightarrow d) \right\}$
7	$a = c > b > d$	$(ablad), (abl\Delta a), (balad), (bal\Delta a), (ad\Delta lab), (ad\Delta lba), (d\Delta lab), (d\Delta lba)$	$\sum_i \left\{ \left( \frac{(ai al)2(bj dl) - (bl dj)}{\epsilon_{aail} \epsilon_{bdjl}} + \frac{(ai dl)2(al bj) - (aj bl)}{\epsilon_{adil} \epsilon_{abjl}} + \frac{(al bi)2(aj dl) - (al dj)}{\epsilon_{abil} \epsilon_{adjl}} + \frac{(bi dl)(aj al)}{\epsilon_{bdil} \epsilon_{aajl}} + (b \leftrightarrow d) \right) \right\}$
8	$a > b = c > d$	$(ablbd)$ , etc.	Same as Case 7, with $a \leftrightarrow b$ .
9	$a > c > b = d$	$(ablc\Delta b)$ , etc.	Same as Case 7, with $a \leftrightarrow b$ and $d \rightarrow c$ .
10	$a > c = d > b$	$(ablc\Delta c)$ , etc.	Same as Case 6, with $a \leftrightarrow c$ and $d \rightarrow b$ .
11	$a > b > c = d$	$(ablc\Delta c)$ , etc.	Same as Case 10.
12	$a > b > c > d$	$(ablc\Delta d), (ablc\Delta c), (balc\Delta d), (balc\Delta c), (cd\Delta lab), (cd\Delta lba), (d\Delta lab), (d\Delta lba)$	$\sum_i \left\{ \left[ \left( \frac{(ai cl)2(bj dl) - (bl dj)}{\epsilon_{acil} \epsilon_{bdjl}} + (c \leftrightarrow d) \right) + (a \leftrightarrow b) \right] + [a \leftrightarrow c, b \leftrightarrow d] \right\}$
13	$a > c > d > b$		Same as Case 12.
14	$a > c > b > d$		Same as Case 12.

A 4-virtual integral  $(ablc\Delta d)$  is multiplied by the contribution appropriate to its case and the result added to the  $C_{ij}^{(3a)}$  matrix element.

via the same DEM code. Work to remove the DEM bottleneck will however be undertaken in the very near future. We mention here that a non-Dyson ADC(3) algorithm has also been reported in the framework of the relativistic four-component theory.<sup>26</sup> Besides other differences, it differs essentially from ours in that it implements directly spin-orbital formulas without spin-adaptation and it does not adopt an integral-driven approach.

To summarize, the complete flow of computation in our implementation of the nD-ADC method may be sketched as follows:

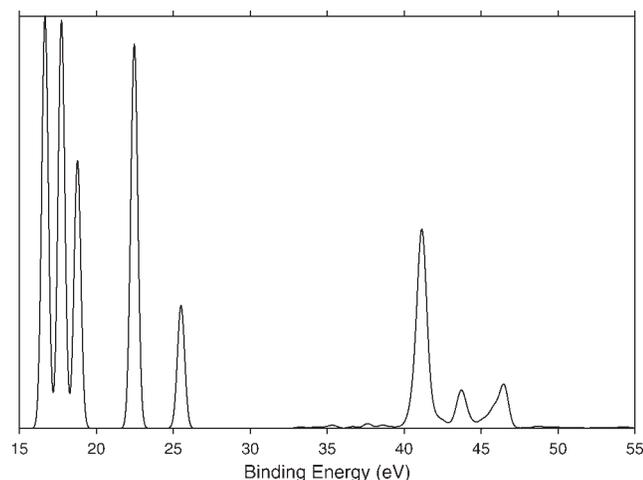
1. The  $\Sigma(\infty)$  matrix is computed (currently using the conventional DEM algorithm) and stored.
2. The  $1h/1h$  block of the  $\mathbf{K} + \mathbf{C}$  matrix, matrix, augmented with the extra non-Dyson terms in eqs. (1)–(6) are computed. In computing the latter, the 4-virtual integrals are processed list-wise from disk according to the scheme in Table 1. The  $1h/1h$  and  $\Sigma(\infty)$  matrices are added and stored.
3. The  $2h1p/1h$  block of the ADC matrix ( $U^H$ ) is computed and stored.
4. The chosen iterative diagonalizer (block-Lanczos) is started. At each iteration, the required non-zero sub-blocks of the  $2h1p/2h1p$  matrix are recomputed from the integrals.

5. After a prescribed number of iterations, the subspace projection of the ADC matrix is diagonalized and the computed eigenvalues and eigenvectors are stored for subsequent processing.

## SOME RESULTS

In this section, we show some preliminary results obtained for  $\text{CF}_4$  as a test case. Our main intention is to demonstrate to the reader the actual differences between the various approaches that have been described in this work and, in particular, some of the most evident advantages of the non-Dyson approach with respect to the Dyson one. Therefore, we performed calculations using three different method: (a) a standard ADC(3) calculation including the full  $N + 1$ -electron space, (b) an ADC(3) calculation featuring the preliminary block-Lanczos reduction of the affinity block, described in ref. 18, (c) a nD-ADC(3) calculation.

$\text{CF}_4$  represents a small and interesting test case, affording accurate calculations with extensive basis sets.<sup>25,27</sup> The molecule plays an important role in various different technologies, for example in the semiconductor industry as a plasma etching gas. Also, by industrial use it has been introduced into the earth's outer atmosphere, where it is widely believed to be involved in the depletion of the earth's ozone layer.



**Figure 2.** The single ionization spectrum of tetrafluoromethane obtained using the nD-ADC(3) method and the cc-pVTZ basis set. The spectra obtained using the three different ADC methods are essentially superimposable.

The preliminary Hartree-Fock calculation generating the molecular orbital basis for the ADC has been performed using the GAMESS-UK package.<sup>28</sup> We have used two different basis sets,<sup>29</sup> aug-cc-pVDZ and cc-pVTZ, at the tetrahedral equilibrium geometry of the molecule,<sup>30</sup> with a C–F bond length of 1.32 Å. The ionization spectrum obtained with the nD-ADC(3) method and the cc-pVTZ basis set is displayed in Figure 2. The spectra obtained by the three different ADC(3) methods are of course the same to within imperceptible deviations. The differences due to the change in basis are also essentially marginal. The spectrum reproduces the experimental results quite well, except for the interval 37.0–50.0 eV. An accurate comparison between experimental and similar theoretical results can be found in ref. 31. Even if the results of the three ADC calculations are practically indistinguishable, the computation effort required in the three cases varies very significantly, as we shall now analyze.

For the purpose of analysis, it is useful to see the entire ADC computation as divided into three main steps:

1. The computation and storage of the smaller blocks of the ADC matrix, involving the  $1h$  and  $1p$  configurations, their coupling to the respective excitations ( $U^H$ ) and the constant self-energy contribution.
2. The execution of the prescribed block-Lanczos iterations. In this step we include also, for the Dyson computation, the preliminary block-Lanczos reduction of the affinity block described in ref. 18. At the end of the iterations, this process yields a block diagonal subspace matrix that needs to be subsequently diagonalized.
3. The diagonalization of the Lanczos projected block-diagonal matrix. The final results are the (approximate) eigenvalues and the corresponding eigenvectors. The latter may be computed in full or, as is most often sufficient for the reproduction of ionization spectra<sup>18</sup> and very much less demanding, only their  $1h$  ( $1p$ ) components.

In Table 2 we report the timing on a “Intel Pentium 4 CPU 3.06 GHz” for the steps above in the three ADC methods. As the table evidences, Step 1 is clearly more expensive in the non-Dyson calculation, due to the necessary overhead to evaluate the extra non-Dyson contributions to the  $1h/1h$  block of the matrix. However, this nD-ADC(3) overhead is much exaggerated in the present tests because the code implementing it is in a very preliminary stage of development where the problem of its efficiency has not yet been studied in any detail. In addition, as explained above, the full DEM approach to the computation of the static self-energy is still used. Replacing this with the  $\Sigma(4+)$  method would significantly speed up Step 1 of the nD-ADC(3) calculation.

During the second step of the calculations, the block-Lanczos iterations are performed and here the superiority of the non-Dyson approach is evident, largely offsetting the overhead incurred in Step 1. Indeed the computation time requested is well over one order of magnitude smaller than in the Dyson cases. But the advantage of separating the ionization problem from the electron affinity one is in fact even much greater.

Finally, in the last step, the diagonalization of the block-diagonal subspace matrix is performed. This is the least onerous step in the calculation but here too the speed-up achieved in the non-Dyson calculation is spectacular and is basically due to elimination of the  $1p$  configuration space: the final size of the block-diagonal matrix is equal to the number of iterations performed times the number of Lanczos starting vectors. Now the latter is equal to the total number of orbitals ( $1p + 1h$ ) in the standard ADC, but only to the number of occupied orbitals in nD-ADC. This makes for at least one order of magnitude reduction in size, which translates into three orders of magnitude reduction in diagonalization time. In addition, the band of the block-diagonal matrix is also equal to the size of the main space and is thus also reduced in nD-ADC.

Turning now to a brief comparison between the timings for the two basis sets used, we mention that the number of active orbitals was 105 in the aug-cc-pVDZ calculation and 145 in the cc-pVTZ one. Although more data would be desirable for a more thorough analysis, inspection of Table 2 then shows that the time for the full ADC(3) calculation scales roughly as  $n^{5.5}$ .

**Table 2.** Computation Time (s) for the Various Steps of the three ADC Procedures Discussed in the Text.

Basis	Method	Step 1	Step 2	Step 3	Tot
aug-cc-pVDZ	ADC(3)	719.4	8168.5	371.8	9259.7
	Reduced ADC(3)	719.8	4499.5	352.4	5204.3
	nD-ADC(3)	2882.3	305.5	0.4	3188.2
cc-pVTZ	ADC(3)	2997.4	50640.0	1034.1	54671.5
	Reduced ADC(3)	2953.9	14239.8	1100.1	18293.8
	nD-ADC(3)	11555.6	819.0	0.4	12375.0

The times reported are cumulative for all the separate symmetry calculations, each involving 200 Lanczos iterations, used to obtain the spectra of Fig. 2. The time for Step 2 of the “Reduced ADC(3)” calculation includes 50 block-Lanczos iterations to truncate the affinity block, according to ref. 18.

(which is essentially the scaling of Step 2) while the affinity-reduced and the nD-ADC calculations both scale as  $n^4$ . However, this dependence is of course entirely dominated by the DEM construction of the energy-independent self-energy in Step 1 and would be very much less demanding for the nD-ADC method once the  $\Sigma(4+)$  approach is implemented. Note further that while Step 3 scales of course as  $n^3$  in the conventional methods, it is independent of the basis size in nD-ADC, as here it only depends on the number of electrons.

In the present example, for comparison purposes, we performed the same number of iterations in all three calculations. However, as we discussed earlier, the relevant rate of eigenvector convergence in the nD-ADC case is much more favorable than in the conventional calculations, meaning that a much smaller number of iterations is required to achieve convergence on the desired region of the spectrum or, if the case, a much wider spectrum can be studied. In the present case for example, converging the same number of ionization poles requires as little as one fourth of the number of Lanczos iterations.

## SUMMARY AND CONCLUSIONS

The basic idea of the non-Dyson approach is to apply the ADC procedure separately to the  $G^-(\omega)$  and  $G^+(\omega)$  parts of the one-particle GF, instead of applying it to the self energy  $\Sigma(\omega)$ . The diagrammatic perturbation expansion of the latter is in fact easier, and has for long time been the standard approach, but the final result is made computationally impractical by the fact that the  $N - 1$ -electron (ionization) and the  $N + 1$ -electron (affinity) problems are coupled. In the nD-ADC approach, the two problems are fully decoupled and can be treated separately, at the cost of some additional higher order terms in the small  $1h/1h$  ( $1p/1p$ ) block of the ADC matrix. For calculating a molecular ionization spectrum, the computational advantage of the nD-ADC approach is enormous on at least three fronts. First, the generally very much larger affinity configuration space (in ADC(3), in particular, the  $1p$  configurations and all their single excitations) disappears from the ADC eigenvalue problem. This makes the size of the ADC matrix vary as the square of the number of electrons rather than of the basis set size, a reduction of roughly two orders of magnitude for typical accurate molecular calculations. The diagonalizer (block-Lanczos in our case) iterations are correspondingly much faster by several orders of magnitude. Second, the elimination of the affinity poles automatically makes the desired ionization energies appear at the outer edge of the ADC spectrum rather than in the middle, and this dramatically speeds up the convergence rate of the diagonalizer in terms of required number of iterations. Third, the main configuration space, deprived of the  $1p$  configurations, is typically one order of magnitude smaller and so are both the bandwidth of the final Lanczos projection and its overall size for a given number of block-iterations.

There are two new main computational issues which must be successfully tackled in order to make the nD-ADC(3) approach fully efficient for application to large molecular systems. The first is that the extra third order terms that appear in the  $1h/1h$  block of the matrix involve two-electron integrals with four vir-

tual indices. As these are in general too numerous to be held in fast memory in the sorted and individually addressable fashion necessary to evaluate each matrix element in turn, we adopted an integrals driven approach, where the list of integrals is scanned (in principle, irrespective of their storage medium or computational source) and their contributions suitably scattered over the matrix elements. The second problem is presented by the evaluation of the energy-independent term of the self-energy. Adopting here the standard iterative approach where the ionization and electron affinity spaces are still coupled defeats in part the whole idea and advantage of the non-Dyson approach. Fortunately, a new procedure has recently been proposed where the decoupling of the  $N - 1$ - and  $N + 1$ -electron spaces is exploited also for the construction of  $\Sigma(\infty)$ .<sup>16</sup> Work is in progress to incorporate this approach in our program. The preliminary tests presented in this article clearly suggest that the final implementation of the non-Dyson method will truly open new and exciting perspectives for the theoretical simulation of molecular ionization spectra.

## Acknowledgments

The authors would like to thank the FIRB project "Molecular compounds and hybrid nanostructured materials with resonant and non-resonant optical properties for photonic devices."

## References

1. Abrikosov, A. A.; Gorkov, L. P.; Dzyaloshinski, I. E. *Methods of Quantum Field Theory in Statistical Physics*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1963.
2. Migdal, A. B. *Theory of Finite Fermi Systems*; Wiley: New York, 1967.
3. Mattuck, R. D. *A Guide to Feynman Diagrams in the Many-Body Problem*; McGraw-Hill: New York, 1967.
4. Fetter, A. L.; Walecka, J. D. *Quantum Theory of Many-Particle Systems*; McGraw-Hill: New York, 1971.
5. Cederbaum, L. S. *Mol Phys* 1974, 28, 479.
6. Schirmer, J.; Cederbaum, L. S.; Walter, O. *Phys Rev A* 1983, 28, 1237.
7. Pickup, B. T.; Goscinski, O. *Mol Phys* 1973, 26, 19.
8. Cederbaum, L. S.; Domcke, W. *Adv Chem Phys* 1977, 36, 205.
9. McCurdy, C. W.; Rescigno, T. N.; Yeager, D. L. McKoy, V. In *Methods of Electronic Structure Theory*; Schaefer, H. F., Eds.; Plenum Press: New York, 1977; p. 339.
10. Ortiz, J. V. *J Chem Phys* 1998, 108, 1008.
11. Schirmer, J. *Phys Rev A* 1982, 26, 2395.
12. Schirmer, J.; Barth, A. *Z Phys A* 1984, 317, 267.
13. Schirmer, J.; Angonoa, G. *J Chem Phys* 1989, 91, 1754.
14. Tarantelli, A.; Cederbaum, L. S. *Phys Rev A* 1989, 39, 1656.
15. Schirmer, J.; Trofimov, A. B.; Stelter, G. *J Chem Phys* 1998, 109, 4734.
16. Trofimov, A. B.; Schirmer, J. *J Chem Phys* 2005, 123, 144114.
17. von Niessen, W.; Schirmer, J.; Cederbaum, L. S. *Comput Phys Rep* 1984, 1, 57.
18. Weikert, H.-G.; Meyer, H.-D.; Cederbaum, L. S.; Tarantelli, F. *J Chem Phys* 1996, 104, 7122.
19. Schirmer, J. *Phys Rev A* 1991, 43, 4647.
20. Lanczos, C. *J Res Natl Bur Stand* 1950, 45, 255.
21. Davidson, E. R. *J Comput Phys* 1975, 17, 87.
22. Parlett, B. N. *The Symmetric Eigenvalue Problem*; Prentice Hall: Englewood Cliffs, NJ, USA, 1980.

23. Saad, Y. Numerical Methods for Large Eigenvalue Problems; Manchester University Press: Manchester, UK, 1992.
24. Tarantelli, F. Chem Phys 2006, 329, 11.
25. Gottfried, F. O.; Cederbaum, L. S.; Tarantelli, F. J Chem Phys 1996, 104, 9754.
26. Pernpointer, M. J Chem Phys 2004, 121, 8782.
27. Feifel, R.; Eland, J. H. D.; Storchi, L.; Tarantelli, F. J Chem Phys 2006, 125, 194318.
28. GAMESS-UK is a package of ab initio programs written by Guest, M. F.; van Lenthe, J. H.; Kendrick, J.; Schoffel, K.; Sherwood, P. with contributions from Amos, R. D.; Buenker, R. J.; van Dam, H. J. J.; Dupuis, M.; Handy, N. C.; Hillier, I. H.; Knowles, P. J.; Bonacic-Koutecky, V.; von Niessen, W.; Harrison, R. J.; Rendell, A. P.; Saunders, V. R.; Stone, A. J.; Tozer, D. J.; de Vries, A. H. The package is derived from the original GAMESS code due to Dupuis, M.; Spangler, D.; Wendoloski, J. NRCC Software Catalog, Vol. 1, Program No. QG01 (GAMESS), 1980.
29. (a) Dunning, T. H., Jr. J Chem Phys 1989, 90, 1007; (b) Kendall, R. A.; Dunning, T. H. Jr.; Harrison, R. J. J Chem Phys 1992, 96, 6796
30. Sutton, L. E. Tables of Interatomic Distances and Configurations in Molecules and Ions; The Chemical Society: London, 1958.
31. Holland, D. M. P.; Potts, A. W.; Trofimov, A. B.; Breidbach, J.; Schirmer, J.; Feifel, R.; Richter, T.; Godehusen, K.; Martins, M.; Tutay, A.; Yalcinkaya, M.; Al-Hada, M.; Eriksson, S.; Karlsson, L. Chem Phys 2005, 308, 43.