Theor Chim Acta (1995) 92: 305–313

Theoretica Chimica Acta

An efficient improvement of the string-based direct selected CI algorithm

Àngels Povill, Jaime Rubio

Departament de Química Física, Universitat de Barcelona, Marti i Franquès 1, E-08028 Barcelona, Spain

Received November 15, 1994/Final revision received October 4, 1995/Accepted November 4, 1995

Summary. A modified version of the Direct Selected Configuration Interaction Using Strings (DISCIUS) code is presented. From a series of test calculations it is shown that the string-based direct-CI algorithm is also very efficient in larger examples than the ones we showed in our former version. The performance of this new code with respect to the computer time is reasonable as well as its storage requirements. Hence, we present a new tool able to treat more than 10^6 determinants of a variational space that does not have a defined structure, i.e. a selected or any way truncated CI.

Key words: Discius algorithm – Configuration interaction

1 Introduction

Recent advances in quantum chemistry methodology together with the development of vector processors have made possible to solve the electronic Schrödinger equation exactly for a limited basis set in a number of cases. However, because the dimension of the Full Configuration Interaction (FCI) hamiltonian matrix grows enormously with the number of orbitals and electrons, it is only possible to perform such calculations for very small molecules and very small orbital spaces. In fact, for the NH_3 molecule with a $3s2p1d$ basis set for the N and a $2s1p$ one for the H, the FCI problem deals with 8 electrons in 28 orbitals leading to a FCI matrix of dimension 2.09×10^8 determinants in C_s symmetry.

A way to circumvent the factorial growing of the FCI expansion is to use approximate methods among which the Multireference CI (MRCI) ones have proven to be very efficient. In the MRCI approaches either starting from a preliminary Complete Active Space (CAS) [1–3] calculation or by iterative selection of the reference space [4–9], the bottleneck lies on the computation and storage of the $\boldsymbol{H}$ matrix. This problem has also been solved by using direct-CI procedures [10–12] and the use of either externally [13, 14] or internally [15, 16] contracted CI techniques. However, these MRCI techniques from selected multireference spaces do also exhibit a strong dependence on the number of electrons and orbitals. In a recent paper a direct selected CI algorithm has been presented [17]. This algorithm is based on a string formulation and hence exhibits an efficient vectorization capability. A closely related algorithm has also been suggested by Caballol and Malrieu [18]. The approach of these authors is based on a hole–particle formalism and is particularly well-suited when the number of orbitals and electrons is large.

In this work we will present a new version of the string-based direct CI algorithm already presented in [17]. Since we want to treat selected variational spaces, it would be more convenient to be able to accurately deal with large systems than very accurately with small ones. We will show that the most time-consuming step can be modified in such a way that the dependence is on some function of the number of strings appearing in the selected CI space rather than on a function of the number of orbitals and electrons. This permits to handle with larger cases as it will be shown.

2 The DISCIUS algorithm

As in any direct-CI procedure we wish to solve the matrix equation

$$\boldsymbol{H}\boldsymbol{c}_i = E_i\boldsymbol{c}_i \tag{1}$$

without explicitly computing and storing the $\boldsymbol{H}$ matrix. The usual procedure is the iterative diagonalization of Davidson [19] and consists in starting from a normalized trial vector $\boldsymbol{c}^{(n)}$ which enables the computation of the residual vector $\boldsymbol{\sigma}^{(n)}$ as

$$\sigma_I^{(n)} = \sum_J H_{IJ} c_J^{(n)} \tag{2}$$

and the energy (at the nth iteration) as

$$E^{(n)} = \boldsymbol{c}^{(n)} \cdot \boldsymbol{\sigma}^{(n)}. \tag{3}$$

The most time-consuming step is precisely the construction of the $\boldsymbol{\sigma}$ vector for the new basis-vector at each iteration of the diagonalization procedure. If the CI space considered has a well-defined, usually very compact, structure as in the CAS, RAS, or FCI ones (where Restricted Active Space (RAS)), the residual vector can be computed very efficiently. However, when the CI space has a general not predefined structure the above algorithms cannot be used directly. Two alternative procedures have been recently reported [17, 18] to deal with selected CI spaces in a direct way. The main difference between them lies on the addressing used to allocate the selected determinants. Thus, the SCIEL program of Caballol and Malrieu [18] uses a hole–particle formalism whereas the Direct Selected CI Using Strings (DISCIUS) algorithm proposed by Povill et al. [17] is string based.

Although the characteristics of the DISCIUS program have been previously reported [17], it is worth to recall some specific aspects. The first is that in DISCIUS, as in other direct CI algorithms such as those of Olsen et al. [20], Knowles et al. [21] or Zarrabian et al. [22], the Slater determinants are constructed as products of α- and β-strings, allowing an easy decomposition in $\alpha\alpha$, $\beta\beta$ and $\alpha\beta$ operations. A second point is that the list of determinants has been ordered in a particular way to allow an efficient handling of the truncated space. Finally, the use of some auxiliary arrays allows an efficient localization of a given determinant in the general list of determinants. Following Olsen et al. [20], the residual vector of the DISCIUS code is written in three parts, $\alpha\alpha$-, $\beta\beta$- and $\alpha\beta$-excitations

$$\sigma(I_\alpha, I_\beta) = \sigma_1^{\alpha\alpha}(I_\alpha, I_\beta) + \sigma_2^{\beta\beta}(I_\alpha, I_\beta) + \sigma_3^{\alpha\beta}(I_\alpha, I_\beta), \tag{4}$$

where I_α and I_β are the corresponding α- and β-strings of a determinant I. If $\{\mathrm{CI}\}$ is the subset of Slater determinants contained in the selected CI expansion, $\{I_\alpha\}$ is the set of α-strings appearing in $\{\mathrm{CI}\}$ and the same holds for $\{I_\beta\}$.

In a selected CI space, important problems arise for the construction, at each iteration step, of the σ vector, whose explicit formulae have already been reported in [17], concerning as well the $\alpha\alpha$ as the $\alpha\beta$ contributions. Since in the examples we will show, the considered molecules have $M_s = 0$, and therefore, the $\beta\beta$ contribution to the σ vector has not to be computed [20], we will omit it in our discussion. However, a completely analogous discussion to σ_1 may be done for σ_2 in cases with an even number of electrons. Let us first analyze the computation of σ_1. Here, one must generate all the single and double excitations of a given I_α string appearing in $\{I_\alpha\}$ and, because of the incompleteness of the $\{\mathrm{CI}\}$ space, ask if the resulting J_α string also belongs to $\{I_\alpha\}$. If n_α and m represent the number of α electrons and of orbitals, respectively, the number of single excitations of a given I_α string grows as $[(n_\alpha)(m - n_\alpha)]$ and the number of double excitations as

$$\left[\binom{n_\alpha}{2}\binom{m-n_\alpha}{2}\right] \quad \text{or} \quad [(n_\alpha)(n_\alpha - 1)(m - n_\alpha)(m - n_\alpha - 1)/4].$$

The dependence on the number of single and double excitations, approximately $[(n_\alpha)^2(m - n_\alpha)^2]$, is unavoidable in a CAS (or FCI) space because all the possible strings are present. The systematic generation of single and double excitations at each iteration is an efficient procedure in these very special kind of CI spaces with a reduced number of active electrons and orbitals. However, the effect of the selection is to reduce the number of active strings, and consequently, only a small part of single and double excitations on a given string generates strings belonging to $\{I_\alpha\}$. Hereafter, we will refer to an excitation connecting two strings that belong to our space as an effective excitation.

In order to solve this difficulty the algorithm has been changed in the following way: each $I_\alpha \in \{I_\alpha\}$ is compared with all the other $J_\alpha \in \{I_\alpha\}$ and the single and double excitations connecting I_α with J_α are computed without needing to ask if a given string belongs to $\{I_\alpha\}$. The introduction of this simple modification permits to reduce the dependence on $[(n_\alpha)^2(m - n_\alpha)^2]$ onto approximately the square of the number of α-strings in $\{\mathrm{CI}\}$, i.e. the number of elements in $\{I_\alpha\}$ set. An additional change consists on the introduction of the $I_\alpha < J_\alpha$ constraint. The separate computation of the diagonal elements ($I_\alpha = J_\alpha$) also brings an advantage since, in this case, there is no need of comparing strings and the same set of I_β strings is related with the starting and the resulting α string. Thus this operation is the only completely vectorizable.

There is another point that should be stressed. In a selected CI, all the possible α-strings of the complete space do not appear. If the selection has been cleverly made, the largest number of differences of the set of $\{I_\alpha\}$ appears in the most external orbitals. Since the orbitals of an α-string are ordered in strictly ascending order, it is very convenient to perform the comparison between two α-strings (I_α and J_α) starting from the right-hand of it (the highest occupied spin-orbital). This simple point allows us to reject a couple of strings very soon during the comparison. In other words, when we know that a string is more than a diexcitation of the other, we can reject the couple without concluding the comparison. The main implication of this is the considerable reduction (about 33%) of the total computation time of σ_1. However, it is difficult to establish a function of the dependence of the computation time of σ_1 on the number of α-strings. It is clearly smaller than the square of the number of α-strings, but it will depend on the number of electrons of the system and, more strongly, on the coupling between the set of selected α-strings. Finally, it must be pointed out that the above modification of the algorithm does not reduce the efficiency of the overall procedure in cases with few orbitals and electrons.

In this way, we have avoided the main problem in the computation of $\alpha\alpha$ and $\beta\beta$ parts of σ vector: the number of tests. This point is crucial since in contrast to the FCI, CASCI, etc, in a selected CI we do not know *a priori* if an excitation from one of the determinants of our selected space gives a determinant belonging or not to this space.

Concerning now the third contribution to σ, the $\alpha\beta$ operations (i.e. σ_3), the effective monoexcitations can be stored in a reasonable size of central storage (in our case, about 128 Mbytes). Then, the main problem is the efficiency of the vectorial part of the algorithm, i.e. the product of the coefficients by the integrals (see [17]). The vector that contains these modified bielectronic integrals is much more sparse in this contribution than the equivalent vector of σ_1 due to two facts. Firstly, we only store in it the integrals involving effective β-monoexcitations while in σ_1 we do effective $\alpha\alpha$-diexcitations. Secondly, although a component of this vector was not zero the vectorization could not be effective if the strings implied did not appear many times in the list of determinants.

In order to analyze this last point, we have counted the number of appearing strings and their frequencies in several selected CI spaces and for different kinds of problems (with different number of electrons and orbitals) and we have seen that in the best of cases a half of the appearing strings are "vectorizable" (i.e. they appear coupled with, say, 12 different I_β strings at least), but in the worst, 75% strings are not vectorizable. Table 1 shows these distributions of strings for the two examples that we have used to test the performance of our modified algorithm. The first example is for the NH_3 molecule with 8 active electrons and 28 active molecular orbitals. The dimension of the CI space is 945 503 determinants selected from the

Table 1. Distribution of strings in selected CIs for NH_3 and N_2 molecules (see the text)

Range of frequencies	Number of strings	Percentage of total appearing α-strings
NH_3 molecule		
1–1	2203	20.45
2–4	848	7.87
5–11	1534	14.24
12–24	2194	20.36
25–49	2050	19.03
50–199	444	4.12
200–5416	1502	13.95
N_2 molecule		
1–1	9095	29.03
2–4	8733	27.87
5–11	5813	18.55
12–24	3082	9.84
25–49	1792	5.72
50–199	2315	7.39
200–7872	504	1.61

1.9 million single and double excitations generated by 1324 reference determinants. This space contains 10 775 different α-strings of the 20 475 possible α-strings. The second one concerns the N_2 molecule with 10 electrons in 58 active MOs. Here, the dimension of the selected space is 718 677 selected from the 8.5 million determinants of the MRSDCI space generated by 1207 reference determinants. The number of appearing α-strings is 31 334 of a total of 4.6 million different α-strings.

This leads to the conclusion that to treat selected spaces no advantage can be reached from vectorization in σ_3 contribution and, consequently, we have modified the construction of σ_3 in such a way that we only multiply a coefficient by an integral if the involved α- and β-monoexcitation are both effective, i.e. if J_α and I_α belong to $\{I_\alpha\}$, J_β and I_β belong to $\{I_\beta\}$ and the determinants $J_\alpha J_\beta$ and $I_\alpha I_\beta$ belong to selected $\{CI\}$ space. In other words, given a monoexcitation ij and an α-string I_α, that lead to a stored J_α, we store all the coefficients of the determinants $J_\alpha J_\beta$ appearing in the list. Then, a loop over $I_\alpha I_\beta$ is made and, for each I_β, a loop over all effective monoexcitations kl on it. The multiplication of the $(ij/kl)''$ integral by the coefficient of $J_\alpha J_\beta$ is only made if the last exists. This change has allowed us to decrease the time of computation of σ_3 dramatically in the cases more unfavourable to vectorization, without loss of efficiency in the situations where the vectorization could be *a priori* effective (those of 50% of vectorizable strings). In other words, the gain obtained from vectorization in the last is compensated by the avoiding of useless products.

The present algorithms to construct σ_1 and σ_3 contributions, including the modifications above explained, are represented in Figs. 1 and 2, respectively. Of course, there are additional loops to take profit of the symmetry of the system, but

```
- Loop over $I_\alpha \in \{I_\alpha\}$
    • Initialization of $F(J_\alpha)$
    • Special case of $J_\alpha = I_\alpha$
    - Loop over $J_\alpha \in \{I_\alpha\}/J_\alpha > I_\alpha$
        • Determination of connexion between $I_\alpha$ and $J_\alpha$
        * If is a monoexcitation $|J_\alpha\rangle = \mathrm{sign}(il) i_\alpha^+ l_\alpha |I_\alpha\rangle$
            • $F(J_\alpha) = \mathrm{sign}(il)\left[\sum_j^m (ij|jl)'' - 2\sum_m^{occ}[(im|ml)'' - (il|mm)'']\right]$
        *
        * If is a diexcitation $|J_\alpha\rangle = \mathrm{sign}(ikjl) i_\alpha^+ k_\alpha^+ j_\alpha l_\alpha |I_\alpha\rangle$
            • $F(J_\alpha) = -2\,\mathrm{sign}(ikjl)[(ij|kl)'' - (il|kj)'']$
        *
    - End loop $J_\alpha$
    - Loop over $I_\beta \in \{I_\beta\}/|I_\alpha I_\beta\rangle \in \{CI\}$
        - Loop over $J_\alpha \in \{I_\alpha\}/|J_\alpha I_\beta\rangle \in \{CI\}$ and $J_\alpha > I_\alpha$
            • $\sigma(I_\alpha, I_\beta) = \sigma(I_\alpha, I_\beta) + F(J_\alpha) \times c(J_\alpha, I_\beta)$
            • $\sigma(J_\alpha, I_\beta) = \sigma(J_\alpha, I_\beta) + F(J_\alpha) \times c(I_\alpha, I_\beta)$
        - End loop $J_\alpha$
    - End loop $I_\beta$
- End loop $I_\alpha$
```

Fig. 1. New algorithm of the σ_1 contribution to σ

```
- Loop over $ij$
    - Loop over $I_\alpha \in \{I_\alpha\}_{\text{effective}}$
        • $|J_\alpha\rangle = \text{sign}(ij) i_\alpha^+ j_\alpha | I_\alpha\rangle$ are stored
        - Loop over $J_\beta \in \{I_\beta\} / |J_\alpha J_\beta\rangle \in \{\text{CI}\}$
            • $F(J_\beta) = \text{sign}(ij) c(J_\alpha, J_\beta)$
        - End loop $J_\beta$
        - Loop over $I_\beta \in \{I_\beta\} / |I_\alpha I_\beta\rangle \in \{\text{CI}\}$
            - Loop over $kl_{\text{effective}}$
                • $|J_\beta\rangle = \text{sign}(kl) k_\beta^+ l_\beta | I_\beta\rangle$ are stored
                * If $F(J_\beta) \neq 0$
                    • $\sigma_3(I_\alpha, I_\beta) = \sigma_3(I_\alpha, I_\beta) + 2\,\text{sign}(kl) F(J_\beta)(ij|kl)''$
                *

                - End loop $kl$
            - End loop $I_\beta$
        • Initialization of $F(J_\beta)$
    - End loop $I_\alpha$
- End loop $ij$
```

Fig. 2. New algorithm of the σ_3 contribution to σ

we have omitted them for the sake of clarity. Both algorithms exhibit some similarity to the full-CI algorithms developed by Olsen et al. [20]. However, the differences are what enable to treat the selected, and therefore arbitrary, CI spaces efficiently. Concerning to the $\alpha\alpha$ contribution (Fig. 1) the main differences arise before the product loop. As has been explained, instead of a loop over couples of monoexcitations we have a loop over the couples of α-strings that is better for truncated spaces. Besides, the initialization of the $F(J_\alpha)$ vector is only carried out over those components that have been used in the previous iteration of I_α. The design of the $\alpha\beta$ contribution of Olsen et al. contains a gather and a scatter operation that enable them to avoid the use of indirect addressing in the time consuming part of the algorithm. In our algorithm there is a similar gather operation (we have the effective monoexcitations stored) but we do not carry out the scatter operation because of the useless vectorization of this kind of spaces. As has been explained, we only do the useful products.

3 Computational details

In order to illustrate the efficiency of the new DISCIUS algorithm two tests are presented here. The first one is the well known benchmark for NH_3 reported earlier by Knowles and Handy [23] and used as a test in our previous work [17]. The second example concerns a larger problem as the N_2 molecule with a cc-pvtz basis set [24] at his equilibrium distance (2.1 a.u.). In this case there are 10 valence electrons correlated in 58 molecular orbitals. The two 1s orbitals of nitrogen atoms have been frozen at the SCF level. We have performed a valence CASSCF (10 valence electrons in 8 molecular orbitals) before starting the selection. Details about geometry and basis sets can be found in [23] and [24], respectively. The first example is representative of a situation in which few electrons are correlated in a limited space. However, the computational problems encountered when dealing with more electrons and more orbitals is very different. The second example contains all the characteristics of a more complicated case.

In both cases a multireference CI space is constructed iteratively following the CIPSI algorithm [4]. The reference space $\{G\}$ contains the most important determinants. Then a selection is carried out on the full MRSDCI space according to the first-order vector coefficients as in the three-class CIPSI algorithm [5].

4 Performance of the modified DISCIUS algorithm

In this section some comparative times of computation between the old and the present algorithms are reported. Calculations have been made on an IBM 3090/600J-VF using one processor only. The calculations have been performed with a modified version of the HONDO package coupled to the CIPSI chain [25].

The comparison for different selected CI spaces in NH_3 reported in Table 2 shows that the modified version does not decrease its efficiency in cases where some vectorization was possible. However, these selected CI spaces have a dimension of 2×10^5 more or less. Hence, the percentage of vectorizable α-strings is smaller than in the example of Table 1, of about 30%. This explains the decrease of time in computing σ_3 contribution. It must be pointed out that with the present version the CPU times of the program running in a completely scalar mode or in vectorial mode are virtually the same.

On the other hand, in the present version the time depends no more on the number of determinants of $\{CI\}$ set but on the square of the number of appearing α-strings multiplied by a factor related to the coupling between them as it has been explained before. In order to analyze this aspect, different natural orbital (NO) transformations have been considered. The time for computing σ in different selected CI from several sets of NO was compared with the corresponding to canonical orbitals in selected spaces with approximately the same number of active strings. The conclusions from those calculations are that the computation time when canonical orbitals are used is longer than the time when natural orbitals are used. The differences between times corresponding to different sets of natural orbitals are not meaningful. The different behaviour between spaces from canonical and natural orbitals comes from the different number of effective single and double

Table 2. Comparison of the time required to compute the σ vector in our previous and present versions of the direct selected CI algorithm for NH_3 molecule (DZP basis set). The calculations have been performed with a modified version of the HONDO package coupled to the CIPSI chain [25]. Time is in seconds on a IBM 3090/600J with VF. $\{G\}$ stands for set of generator determinants, $\{GD\}$ for set of generated determinants, $\{CI\}$ for the set of selected determinants and $N_{\{I_\alpha\}}$ for the number of appearing α-strings

				Old version			New version		
dim $\{G\}$	dim $\{GD\}$	dim $\{CI\}$	$N_{\{I_\alpha\}}$	t_{σ_1}	t_{σ_3}	t_σ	t_{σ_1}	t_{σ_3}	t_σ
41	170378	170419	4259	93	248	341	70	137	207
135	459522	203671	5251	105	301	406	64	168	232
371	897411	215260	5193	103	315	418	65	183	248

Table 3. Timing of DISCIUS algorithm for the NH_3 case. Time is in seconds on an IBM 3090/600J with VF. $\{G\}$ stands for set of generator determinants, {GD} for set of generated determinants, {CI} for the set of selected determinants and $N_{\{I_\alpha\}}$ for the number of appearing α-strings

dim $\{G\}$	dim {GD}	dim {CI}	$N_{\{I_\alpha\}}$	t_{σ_1}	t_{σ_3}	t_σ
1324	1908892	945503	10775	380	1090	1470
2565	2492630	824705	9854	315	960	1275
		1001392	10671	390	1160	1550
3941	3266674	896242	10405	350	1010	1360
		1102078	11355	465	1290	1755

Table 4. The same as Table 3 for the N_2 molecule

dim $\{G\}$	dim {GD}	dim {CI}	$N_{\{I_\alpha\}}$	t_{σ_1}	t_{σ_3}	t_σ
1207	8458554	657471	31334	545	575	1120
3339	18711084	825616	37382	685	705	1390

excitations that exists in their respective spaces, or in other words, the strings are connected in a more efficient way when we use natural orbitals and there are more null matrix elements. Unfortunately, the time we gain using NO is spent in the Davidson procedure: as it is known, the convergence of this method is better when using canonical orbitals than, practically, all other kinds of them.

In Table 3 we report results of the NH_3 case for three reference spaces and, for each of these spaces, one or two diagonalizations of selected MRSDCI spaces. In any case time ranges between 1200 and 1800 s per iteration and the σ_3-time is approximately three times the σ_1-time.

Now, let us turn our attention to Table 4 that shows results of the N_2 case. We present here two diagonalizations of selected spaces from different generator space. Here, the σ_1-time is more or less the same that the σ_3-time. This ratio between σ-times is similar to the ratio expected from a full-CI code (with $M_s = 0$) what means that, in this case, we avoid a comparable number of operations in both contributions.

Finally, we must point out that the accuracy of the present method is closely related to the selection procedure employed. The use of the CIPSI [4, 5] selection criteria permits to efficiently include all the important determinants higher than double excitations even when dealing with large systems [26–31].

5 Conclusions

A modified version of the new selected direct CI scheme has been presented and implemented. The strategy has changed in such a way that now it is able to treat any kind of selected spaces with the same level of efficiency. The time required to compute the residual vector, σ, demonstrates that the present version makes it possible to carry out variational treatments of more than a million selected determinants only using 128 Mbytes of central storage. Moreover, the structure of these variational spaces is completely general. The performance of this new code with respect to the computer time and storage requirements are reasonable.

The coupling of this code to the CIPSI philosophy [4, 5] to select the determinants and the recent implementation of a diagonal self-consistent size-consistent dressing of the hamiltonian matrix proposed by Daudey, Heully and Malrieu [32, 33], provide us of a tool able to do chemical calculations of high accuracy as it has recently been shown in [34, 35].

Acknowledgements. Authors are indebted to DGICYT (Project PB92-0766-C02). The calculations were performed on an IBM 3090/600 VF and computation time was kindly provided by CESCA (Centre de Supercomputació de Catalunya). We also want to thank Dr. Rosa Caballol for her helpful discussions.

References

1. Roos BO, Taylor PR, Siegbahn PEM (1980) Chem Phys 48:157
2. Siegbahn PEM, Almlöf J, Heiberg A, Roos BO (1981) J Chem Phys 74:2384
3. Roos BO, (1980) Int J Quantum Chem 14:175
4. Huron B, Rancurel P, Malrieu JP (1973) J Chem Phys 75:5745
5. Evangelisti S, Daudey JP, Malrieu JP (1983) Chem Phys 75:91
6. Buenker RJ, Peyerimhoff SD (1968) Theoret Chim Acta 12:183
7. Buenker RJ, Peyerimhoff SD (1974) Theoret Chim Acta 35:33
8. Buenker RJ, Peyerimhoff SD (1975) Theoret Chim Acta 39:217
9. Siegbahn PEM (1980) J Chem Phys 72:1647
10. Siegbahn PEM (1984) Chem Phys Lett 109:417
11. Duch W (1986) Chem Phys Lett 124:442
12. Knowles PJ, Werner HJ (1988) Chem Phys Lett 145:514
13. Siegbahn PEM (1977) Chem Phys 25:197
14. Siegbahn PEM (1983) Int J Quantum Chem 23:1869
15. Meyer W (1977) In: Shaefer III HF (ed) Methods of electronic structure theory. Plenum, New York
16. Siegbhan PEM (1980) Int J Quantum Chem 18:1229
17. Povill A, Rubio J, Illas F (1992) Theoret Chim Acta 82:229
18. Caballol R, Malrieu JP (1992) Chem Phys Lett 188:543
19. Davidson ER (1975) J Comput Phys 17:87
20. Olsen J, Roos BO, Jørgensen P, Jensen HJ, (1988) J Chem Phys 89:2185
21. Knowles PJ, Handy NC (1989) Com Phys Comm 54:75
22. Zarrabian S, Sarma CS, Paldus J (1989) Chem Phys Lett 155:183
23. Knowles PJ, Handy NC (1989) J Chem Phys 91:2396
24. Dunning Jr TH (1989) J Chem Phys 90:1007
25. Dupuis M, Rys J, King HF, HONDO-76, program 338, QCPE, University of Indiana, Bloomington, IN 47401. Pseudopotential adaptation by Daudey JP and Pelissier M. General ROHF adaptation by Cabollol R and Daudey JP. CIPSI chain of programs by Pelissier M, Daudey JP, Malrieu JP, Evangelisti S, Spiegelmann F, Maynau D, Rubio J and Illas F.
26. Harrison RJ (1991) J Chem Phys 94:5021
27. Illas F, Rubio J, Ricart JM, Bagus PS (1991) J Chem Phys 95:1877
28. Illas F, Bachs M, Rubio J, Ricart JM (1984) J Chem Phys 91:5466
29. Illas F, Cañellas J, Ricart JM, Rubio J (1990) J Chem Phys 93:2603
30. Novoa JJ, Mota F, Arnau F (1990) Chem Phys Lett 165:50
31. Evangelisti S, Bendazzoli GL, Ansaloni R, Rossi E (1995) Chem Phys Lett 233:353
32. Daudey JP, Heully JL, Malrieu JP (1993) J Chem Phys 99:1240
33. Heully JL, Malrieu JP (1992) Chem Phys Lett 199:545
34. Povill A, Caballol R, Rubio J, Malrieu JP (1993) Chem Phys Lett 209:126
35. Povill A, Rubio J, Caballol R, Malrieu JP (1994) Chem Phys Lett 218:283