On the Full Exploitation of Symmetry in Periodic (as well as Molecular) Self-Consistent-Field Ab initio Calculations.

Roberto Orlando,[1, a)] Marco De La Pierre,[1, 2] Claudio M. Zicovich-Wilson,[3] Alessandro Erba,[1] and Roberto Dovesi[1]

[1)] *Dipartimento di Chimica, Università di Torino and NIS, Nanostructured Interfaces and Surfaces, Centre of Excellence, Via P. Giuria 7, 10125 Torino, Italy*

[2)] *Nanochemistry Research Institute, Department of Chemistry, Curtin University, GPO Box U1987, Perth WA 6845, Australia*

[3)] *Facultad de Ciencias, Universidad Autónoma del Estado de Morelos, Av. Universidad, 1001, Col. Chamilpa, 62209 Cuernavaca (Morelos), Mexico*

Use of symmetry can dramatically reduce the computational cost (running time and memory allocation) of Self-Consistent-Field *ab initio* calculations for molecular and crystalline systems. Crucial for running time is symmetry exploitation in the evaluation of one- and two-electron integrals, diagonalization of the Fock matrix at selected points in reciprocal space, reconstruction of the density matrix. As regards memory allocation, full square matrices (overlap, Fock and density) in the Atomic Orbital (AO) basis are avoided and a direct transformation from the packed AO to the SACO (Symmetry Adapted Crystalline Orbital) basis is performed, so that the largest matrix to be handled has the size of the largest sub-block in the latter basis. Quantitative examples, referring to the implementation in the CRYSTAL code, are given for high symmetry families of compounds such as carbon fullerenes and nanotubes.

I. INTRODUCTION

In the study of the electronic structure of crystalline compounds, it is mandatory to exploit translational symmetry by using a variational basis of Bloch Functions (BFs), so that a block-diagonal structure is obtained for the (infinite) Hamiltonian matrix, each block corresponding to a **k** point in the First Brillouin Zone (BZ) (see first transformation in Figure 1). Indeed, all periodic codes are based on the use of BFs. Point symmetry, on the other hand, is totally or partially neglected in many computer codes, both periodic and molecular; only a subset of point groups (if any) is implemented, whose operators are used at some steps of the calculation: the largest point group implemented in most molecular codes is D_{2h} (the only exception is, to the authors' knowledge, the TURBOMOLE program[1]), a possible reason being that most (certainly not all!) of the molecular systems, when containing many atoms, have usually low symmetry. The low exploitation of symmetry is surprising, in particular in solid state, when considering the emphasis given in many cases to the need for "linear scaling" (a keyword in the title of a large number of papers)[2–13] of the algorithms. Why hunting for "linear scaling" and at the same time forsaking a saving factor that can be as large as 48 (cubic space groups), or 120 (fullerenes), or even more for nanotubes? A very recent publication[14] fills the gap in part. It describes the use of symmetry in the calculation of the two-electron integrals (but not in the other

steps of the calculation) in biatomic cubic compounds and carbon nanotubes, with important saving factors, along lines very similar to the ones described many years ago by one of the present authors[15] and implemented in the first release of the CRYSTAL code (CRYSTAL88).

In the present paper we discuss how symmetry permits a drastic reduction in running time and memory allocation in Self-Consistent Field (SCF) calculations of the electronic structure for both molecules and solids. This formal scheme has been implemented for the new release of the CRYSTAL code[16,17] (CRYSTAL14, see http://www.crystal.unito.it).

Symmetry can reduce running time at various steps of the SCF process:

1. Calculation of one- and two-electron integrals, whose number is reduced by up to a factor of N_{op}, the number of point symmetry operators R in the group; this part was implemented almost 30 years ago in the first version of the CRYSTAL code;[15]

2. Numerical integration of the exchange-correlation density functional in an asymmetric subset of grid points;[15]

3. Selection of a subset of **k** points of the BZ, namely the Irreducible Brillouin Zone (IBZ), at which the Fock matrix is diagonalized (this applies to periodic systems only). The number of **k** points is reduced also in this case by a factor approaching N_{op} when a dense mesh of points is used;[15]

4. Diagonalization of the Fock matrix, by using Symmetry Adapted Crystalline Orbitals (SACOs, in
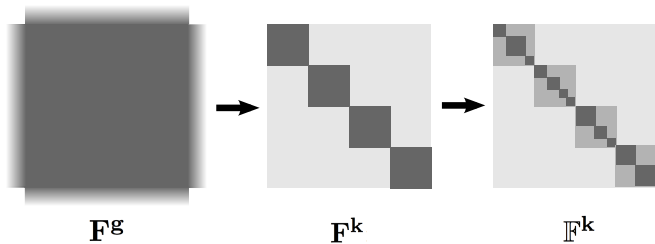
a)Electronic mail: roberto.orlando@unito.it

FIG. 1. Block-factorization of the Fock matrix. $\mathbf{F^g}$: basis of AOs (Atomic Orbitals, non-packed form; borders are blurry to indicate that such matrix is infinite in principle); $\mathbf{F^k}$: basis of BFs (Bloch Functions, i.e. Crystalline Orbitals); $\mathbb{F}^{\mathbf{k}}$: basis of SACOs (Symmetry Adapted Crystalline Orbitals).

the case of solids) or Symmetry Adapted Molecular Orbitals (SAMOs, for molecules) as a basis. SACO/SAMO bases are generated automatically in CRYSTAL from the selected basis set of Atomic Orbitals (AOs) and their symmetry, with no reference to a data-base of the Irreducible Representations (IRs) or character tables. This part of the code was implemented more than 15 years ago.[18,19] In this case saving factor in computational time is even larger than for all previous items in this list, as being roughly proportional to the third power of the ratio of the number of AOs in the basis set (N_{AO}) to the size of the largest block of the Fock matrix when represented in the SACO/SAMO basis (M_{IR});

5. Reconstruction of the density matrix, a step scaling with the third power of the basis set N_{AO}^3, as each of the N_{AO}^2 matrix elements is obtained by summing over all occupied crystalline orbitals (in the order of N_{AO}). Instead, if a density matrix is first formed in the SACO/SAMO basis for each of the N_{IR} IRs and then transformed to the AO basis, computational time gets roughly proportional to $N_{IR} \cdot M_{IR}^3 + N_{IR} \cdot M_{IR}^2 \cdot N_{op}$, with significant saving. This part of the scheme has been implemented for the present study.

Computational time is not the only issue in handling large unit cell systems, as memory can become a computational bottleneck if not properly treated at every stage of the calculation. In this respect, avoiding representation of the Fock, overlap and density matrices as full square matrices in the AO basis is crucial. As both one- and two-electron integrals are evaluated in such a basis, a set of back and forth transformations are required from the AOs to the SACOs/SAMOs bases. These expedients have been fully implemented in the CRYSTAL code for the present study, too.

In the following, performance in terms of both computational time and memory requirements, resulting from full symmetry exploitation, will be illustrated in the case of two series of highly symmetric compounds: carbon fullerenes of the (n, n) family[20] (120 symmetry operators), and carbon nanotubes of the zig-zag $(n, 0)$

family[21,22] ($8 \times n$ symmetry operators, where n up to 100 has been considered). Systems with higher periodicity are not considered here because the corresponding maximum number of point operators (48) is lower than for fullerenes and nanotubes, and then the advantages of the full exploitation of symmetry less evident. The scheme is however fully general, and in no way limited to low dimensionality systems, as documented in a second paper.[23]

The structure of the present paper is as follows: in Section II the computational scheme is shown, and the use of symmetry for reducing both the CPU cost and the memory allocation discussed. Section III provides details of the computational setup adopted for the calculations. The effectiveness of the scheme is documented in Section IV. The main conclusions are drawn in Section V.

Additional information is available in the supplementary material;[24] corresponding tables in this section are labelled with the S prefix.

II. THE COMPUTATIONAL SCHEME

The following scheme has been implemented in the new CRYSTAL14 code.[16,17] All the presented features are in the current public version, except for the formulation of density matrix in terms of SACOs (point G. below), that is implemented in a second release of the code to be distributed in 2014. A brief outline was presented about two years ago, when the scheme was still at the project stage.[25]

CRYSTAL was designed so as to take advantage of the chemical nature and connectivity of the system. Thus, interactions are computed in real space and selected on the basis of AO overlap criteria. However, as recalled above, the SCF problem for periodic systems must be solved in the reciprocal space by representing matrices in the BF basis. For these reasons, all three types of matrices used in the SCF cycle, namely overlap ($\mathbf{S}$), Fock ($\mathbf{F}$) and density ($\mathbf{P}$), must be represented both in the AO and the BF basis along the SCF cycle. Moreover, they all can be stored either in their symmetry irreducible form or in the symmetry full form. In real space, every matrix $\mathbf{M}$ ($\mathbf{M} = \mathbf{S}$, $\mathbf{F}$, $\mathbf{P}$) is expressed in the AO basis and stored in packed arrays as a result of the application of effective integral selection (*screening*) criteria, that drastically reduce the number of matrix elements to be computed and stored (see point C. below). Hence their size scales linearly with the size of the system, instead of quadratically. In the following, such matrices are labeled with Latin characters: upper-case for full matrices ($\mathbf{M^g}$) and lower-case for matrix irreducible blocks ($\mathbf{m^g}$), with $\mathbf{g}$ denoting a real space lattice basis vector. $\mathbf{M^g}$ is obtained from $\mathbf{m^g}$ by application of the symmetry operators $\mathbf{R}$ of the point group. For molecules we can formally assume that $\mathbf{g}$ represents the null vector only. In highly symmetric cases, $\mathbf{m^g}$ is much smaller than $\mathbf{M^g}$: Table I shows that the size of $\mathbf{F^g}$ and $\mathbf{f^g}$ for C60, with accurate

TABLE I. General information about a family of (n,n) fullerenes. N_{at}^{tot} is the total number of atoms, N_{at}^{asym} the subset in the asymmetric unit; N_{AO} is the total number of atomic orbitals; $S_\mathbf{f}$ denotes the size of the asymmetric wedge of the Fock matrix, and $S_\mathbf{F}$ its expanded version obtained by application of the symmetry operators (both matrices are packed); M_{IR} is the number of rows (columns) of the largest diagonal block in the square $N_{AO} \times N_{AO}$ Fock matrix in the SAMO basis ($\mathbb{F}$); $\mathcal{R}_1$, $\mathcal{R}_2$ and $\mathcal{R}_3$ are the ratios $S_\mathbf{F}/S_\mathbf{f}$, $S_\mathbf{f}/N_{at}^{tot}$ and N_{AO}/M_{IR}, respectively. All fullerenes possess the same symmetric pattern with 120 operators and 10 IRs.

(n,n)	N_{at}^{asym}	N_{at}^{tot}	N_{AO}	$S_\mathbf{f}$	$S_\mathbf{F}$	M_{IR}	$\mathcal{R}_1$	$\mathcal{R}_2$	$\mathcal{R}_3$
(1,1)	1	60	840	1'759	169'980	38	97	29	22
(2,2)	3	240	3'360	6'707	716'130	146	107	28	23
(4,4)	10	960	13'440	25'377	2'847'690	572	112	26	23
(6,6)	21	2'160	30'240	55'661	6'362'370	1'278	114	26	24
(8,8)	36	3'840	53'760	97'559	11'260'170	2'264	115	25	24
(10,10)	55	6'000	84'000	151'071	17'541'090	3'530	116	25	24

basis set and computational conditions (see Section III),

is 169'980 and 1759, respectively. Their ratio ($\mathcal{R}_1$ in the Table) is 97, tending to 120 for larger fullerenes.

In reciprocal space, $\mathbf{F}$ and $\mathbf{S}$ must both be in square form (this is a capital difference with respect to the direct space representation) to be diagonalized at each $\mathbf{k}$ point in the IBZ and yield the one-electron eigenvalues and eigenvectors (in molecular cases it can formally be assumed that there exists only the Γ point in the IBZ). The eigenvectors are also to be stored to memory as they are needed in the reconstruction of the density matrix. They are arranged in a matrix, $\mathbf{A}$, with the same size and symmetry properties related to $\mathbf{F}$. In the following, matrices in reciprocal space (including $\mathbf{A}$) are denoted as $\mathbf{M}^\mathbf{k}$ when the BFs are constructed in the basis of AOs, whereas blackboard characters ($\mathbb{M}^\mathbf{k}$) are used when the BFs are in the basis of SACOs/SAMOs so that the overlap, Fock and eigenvectors matrices take a block-diagonal form, each block being labeled by an irreducible representation index ($\mathbb{M}^{\mathbf{k},IR}$).

The reader can appreciate the advantage of using the SAMO basis by taking, again, C60 as an example (Table I): the size of $\mathbf{F}^\mathbf{k}$ is 840×840 whereas that of the largest $\mathbb{F}^{\mathbf{k},IR}$ is only 38×38, that is, the size of the largest IR matrix to diagonalize and store to memory is 489 times smaller than $\mathbf{M}^\mathbf{k}$.

Here follows a list of the steps where CRYSTAL widely benefits from deep use of the system symmetry:

A. Symmetry is used to identify relationships between atoms, shells, atom pairs, so that linear transformations along the SCF process can be performed efficiently through mapping tables. It is also used to find the IRs and the character table, as well as the transformation matrices $\mathbf{W}^\mathbf{k}$ from the AO to the SACO/SAMO basis, which are stored in a compressed form.

B. One- and two-electron integrals are evaluated with reference to the symmetry irreducible set of atoms and shell-shell couples.

C. Every element of the direct space irreducible Fock matrix wedge $\mathbf{f}^\mathbf{g}$ is formed by combination of one- and two-electron integrals with the density matrix $\mathbf{P}^{\mathbf{g}'}$:

$$\mathbf{f}_{\mu\nu}^\mathbf{g} = \sum_{\lambda,\rho,\mathbf{g}'} \mathbf{P}_{\lambda\rho}^{\mathbf{g}'} \sum_\mathbf{h} \Big[(\mu^\mathbf{0}\nu^\mathbf{g}|\lambda^\mathbf{h}\rho^{\mathbf{h}+\mathbf{g}'}) - \frac{1}{2}(\mu^\mathbf{0}\lambda^\mathbf{h}|\nu^\mathbf{g}\rho^{\mathbf{h}+\mathbf{g}'}) \Big] + H_{\mu\nu}^\mathbf{g} \quad (1)$$

where square brackets contain the Coulomb and exchange integrals and $H_{\mu\nu}^\mathbf{g}$ is the one-electron contribution to the Fock matrix. Summations in Eq. 1 are truncated by screening techniques based on overlap criteria between gaussian functions or charge distributions for multipolar expansions and on an estimate of the density matrix range in real space (see Refs. 16, 17, and 26). In the same way, the number of matrix elements to be computed is restricted to those atoms related by some degree of vicinity so that matrices in real space take a very compact structure scaling linearly with the system size, as anticipated above. In addition to this, only the irreducible wedge of the matrices is formed and only $\mathbf{f}^\mathbf{g}$ and $\mathbf{p}^\mathbf{g}$ are stored to memory. Shell-shell blocks of $\mathbf{P}^\mathbf{g}$ in Eq. 1 are obtained by rotation from $\mathbf{p}^\mathbf{g}$ when needed and this can be done with a "low-memory" approach illustrated in Refs. 27 and 28. For example, in the case of C60 (Table I), only 1759 elements of $\mathbf{f}^\mathbf{g}$ ($\mathbf{g} = \mathbf{0}$, in this case)

were computed and stored to memory instead of $840 \times 840 = 705'600$ in the full square Fock matrix $\mathbf{F^g}$, that is, 400 times less elements.

D. The exchange-correlation functional of the electron density in the unit cell is integrated numerically on a set of points belonging to the asymmetric part (that is the symmetry irreducible part) of the integration grid.

E. Transformation of the overlap and Fock matrices from real to reciprocal space ($\mathbf{m^g} \rightarrow \mathbb{M}^\mathbf{k}$) requires in principle the following intermediate transformations:

$$\mathbf{m^g} \xrightarrow{a} \mathbf{M^g} \xrightarrow{b} \mathbf{M^k} \xrightarrow{c} \mathbb{M}^\mathbf{k} \qquad (2)$$

corresponding to:

a) Generation of the full overlap and Fock matrices in real space by applying the set of symmetry operators $\mathbf{R}$ of the point group to the corresponding irreducible matrices:

$$\mathbf{M^{g'}} = \sum_\mathbf{R} \mathbf{R}^\dagger \mathbf{m^g} \mathbf{R} \qquad (3)$$

b) Fourier transform of $\mathbf{M^g}$ to reciprocal space, that is, from the AO to the BF basis (first transformation in Figure 1):

$$\mathbf{M^k} = \sum_\mathbf{g} e^{i\mathbf{k} \cdot \mathbf{g}} \mathbf{M^g} \qquad (4)$$

In the molecular case, obviously, no transformation to the reciprocal space is needed.

c) Transformation of $\mathbf{M^k}$ into the SACO block-diagonal form (second transformation in Figure 1):

$$\mathbb{M}^\mathbf{k} = \mathbf{W}^{\mathbf{k}\dagger} \mathbf{M^k} \mathbf{W^k} \qquad (5)$$

$\mathbf{W}$ matrices take a different form for the various $\mathbf{k}$ points, depending on their multiplicity. In the molecular case, the SAMO block-diagonal form $\mathbb{M}^\mathbf{k}$ is obtained directly from $\mathbf{M^g}$, with formally $\mathbf{k} = \mathbf{g} = \mathbf{0}$.

In Crystal09 the three steps in Eq. 2 implied the calculation of the four full matrices at every $\mathbf{k}$ point along subsequent steps. However, since steps a, b and c are mere linear transformations, by an appropriate mapping we can obtain every block of $\mathbb{M}^\mathbf{k}$ ($\mathbb{M}^{\mathbf{k},\mathrm{IR}}$) in a single step from $\mathbf{m^g}$, so reducing dramatically the amount of data to keep in memory with respect to storage of the full square matrices in the AO basis ($\mathbf{M^k}$). The required size is equal to the size of the largest matrix $\mathbb{M}^{\mathbf{k},\mathrm{IR}}$ (M_{IR} in Table I). This latter procedure is followed in Crystal14. As already mentioned, in the case of C60 this amounts to $M_{IR}^2 = 1444$ elements, instead of $N_{AO}^2 = 705'600$ in the AO basis, the ratio between the two allocations, $(\mathcal{R}_3)^2$, being as large as 489.

F. Every block of the overlap or Fock matrix in the SACO/SAMO basis ($\mathbb{M}^{\mathbf{k},\mathrm{IR}}$) is diagonalized and the eigenvectors $\mathbb{A}^{\mathbf{k},\mathrm{IR}}$ are obtained. As anticipated, the corresponding saving factor in computing time can be huge.

G. Formation of the irreducible density matrix in real space ($\mathbf{p^g}$) was achieved through the following steps in Crystal09:

$$\mathbb{A}^{\mathbf{k}_{\mathrm{IBZ}}} \xrightarrow{d} \mathbf{A}^{\mathbf{k}_{\mathrm{IBZ}}} \xrightarrow{e} \mathbf{A}^{\mathbf{k}_{\mathrm{BZ}}} \xrightarrow{f} \mathbf{P}^{\mathbf{k}_{\mathrm{BZ}}} \xrightarrow{g} \mathbf{p^g} \qquad (6)$$

where:

d) The eigenvectors of the Fock matrix in the BF basis for every $\mathbf{k}$ point in the asymmetric part of the first Brillouin zone (IBZ) were obtained from those in the SACO/SAMO basis. They formed large square matrices with the same size as $\mathbf{F^k}$.

e) The eigenvectors at every $\mathbf{k}$ point in the first Brillouin zone (BZ) were obtained from those in the IBZ by application of all symmetry operators $\mathbf{R}$, because back-Fourier transform to real space requires integration over the entire first Brillouin zone, *i.e.* a sum over all $\mathbf{k}$ points in the BZ.

f) Formation of the full square density matrix in reciprocal space.[26]

g) Fourier transform of $\mathbf{P}^{\mathbf{k}_{\mathrm{BZ}}}$ to real space.

Here $\mathbf{p^g}$ is in compact form because it takes advantage of both symmetry and interaction screening. In the second release of Crystal14 (to be distributed in 2014), this matrix is generated straightforwardly from the eigenvectors in the SACO/SAMO basis, $\mathbb{A}^\mathbf{k}$, by means of the same mapping that was used for transforming the overlap and Fock matrices from real to reciprocal space (point E. above). Such an improvement has a strong impact both on computational time and memory usage, because full square matrices, such as $\mathbf{A^k}$ and $\mathbf{P^k}$, are no longer to be allocated (their size would be 840×840 in the case of C60 in Table I).

The new scheme implies the following steps:

$$\mathbb{A}^{\mathbf{k}_{\mathrm{IBZ}}} \xrightarrow{h} \mathbb{P}^{\mathbf{k}_{\mathrm{IBZ}}} \xrightarrow{i} \mathbb{P}^{\mathbf{k}_{\mathrm{BZ}}} \xrightarrow{l} \mathbf{p^g} \qquad (7)$$

namely:

h) Construction of the density matrix in the SACO/SAMO basis at reciprocal space points in the IBZ. Unlike step f above, eigenvectors have the size of the IR they belong to, instead of the full number of the AOs in the unit cell.

i) Generation of the density matrix at all reciprocal space points in the BZ from those in the IBZ by application of all symmetry operators (again, arrays are much shorter than in the AO basis adopted in step e above).

l) Fourier transform and basis change from SACOs

(or SAMOs for molecules) to AOs. Since such a transformation implies matrix double products, it scales in principle with the third power of the number of basis functions. Nevertheless, matrices in the SACO basis are in block form, whose maximum dimension in C60 is $M_{IR} \times M_{IR} = 38 \times 38$ to be compared with the larger $N_{AO} \times N_{AO} = 840 \times 840$. Actually, both Fourier and SACO to AO transformations now take full advantage of this property.

III. COMPUTATIONAL CONDITIONS

Simulations were performed with the B3LYP hybrid functional.[29–31] Carbon atoms were described using a 6-31G(d)[32] Gaussian-type basis set, consisting of 14 atomic orbitals (AOs) per atom. The level of accuracy in evaluating the two-electron Coulomb and Hartree-Fock exchange series is controlled by five parameters T_i ($i = 1, ...5$).[16] T_1 and T_2 refer to the Coulomb integrals, T_3, T_4 and T_5 to exchange. In this study accurate values such as 8, 8, 8, 8, 20 were chosen. The DFT exchange-correlation contribution is evaluated by numerical integration over the unit cell volume. In CRYSTAL, radial and angular points of the grid are generated through Gauss-Legendre radial quadrature and Lebedev two-dimensional angular point distributions. A (75,974)p grid was used, corresponding to a pruned grid with 75 radial and 974 angular points (XLGRID keyword in the CRYSTAL14 manual).[16] For nanotubes, the reciprocal space was sampled along the unique lattice vector according to a sublattice with shrinking factor[16] set to 10, corresponding to 6 independent $\vec{k}$ vectors in the irreducible part of the Brillouin zone. All calculations were run on a cluster with 8 Intel Xeon E5620 cores working at 2.40 GHz, and 2 GBytes RAM per core.

IV. RESULTS AND DISCUSSION

A. Fullerenes

Fullerenes, as highly symmetric molecular structures, are an excellent first example of how symmetry can make *ab initio* calculations for large molecules affordable with a low request for computational resources. We consider here (n, n) fullerenes of increasing size, from C60 ($n = 1$) to C6000 ($n = 10$; see Figure 2 a); the corresponding number of atoms grows quadratically with n, $N_{at}^{tot} = 60n^2$, so that it ranges from 60 to 6000 (see Table I). Correspondingly, the basis set size (N_{AO}) increases from 840 to 84'000 AOs. The number of symmetry operators is constant and equal to 120 along the whole series.

Let us begin with discussing the effect of symmetry on the size of matrices involved in the SCF process. All 60 atoms in C60 are related by symmetry so that there is only one C atom in the asymmetric unit (N_{at}^{asym} in Table I). Because there are 120 symmetry operators in
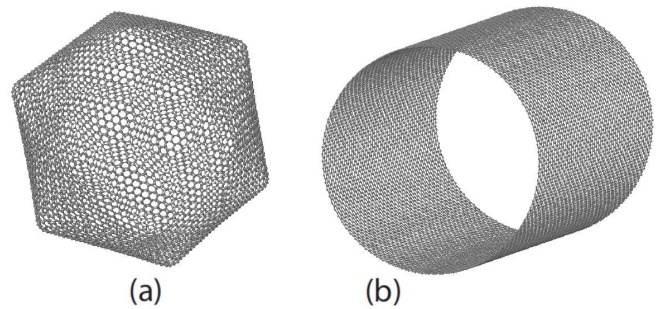

FIG. 2. Graphical representation of the structure of (a) the (10,10) carbon fullerene of I_h point group with an average radius of 35.0 Å and (b) the (100,0) carbon nanotube with a radius of 39.2 Å.

the I_h point group of fullerenes, the ratio $N_{at}^{tot}/N_{at}^{asym}$ becomes more favourable as the molecule size increases (it raises to 109 for C6000). Thus, the impact of the use of symmetry on the size of matrices and on general performance is expectedly huge. In spite of the applied screening to reduce the number of the interactions to be considered, the size of the Fock matrix in real space ($S_{\mathbf{F}}$ in Table I) varies from 0.17 to 17.5 millions of elements along the series considered here when symmetry is not taken into account; note the linear trend as a function of the number of atoms. The size of the same matrix ($S_{\mathbf{f}}$ in Table I) drops to 1759÷151'071 elements when reduced by symmetry, corresponding to a saving factor as large as 97 for C60 and 116 for C6000 ($\mathcal{R}_1$ in Table I), that is, very close to N_{op} (120). Beside the great reduction in memory storage, $\mathcal{R}_1$ is also an important index of performance as being also the saving factor in the calculation of one- and two-electron integrals as a consequence of the application of symmetry. The overall scaling of matrix sizes with system size, taking into account both screening and symmetry, is linear as revealed by index $\mathcal{R}_2 = S_{\mathbf{f}}/N_{at}^{tot}$ (average allocation per atom): its value is 29 for C60 and 25 for C6000. In this particular case, the global trend is even better than linear due to the averagely higher multiplicity of atoms in larger fullerenes. If no screening strategy were adopted and symmetry neglected, scaling with size would be quadratic as matrices would contain N_{AO}^2 elements. In this respect, the ratio $N_{AO}^2/S_{\mathbf{f}}$ measures the saving factor (in terms of matrix elements) of a calculation with symmetry and interaction screening over a "brute force" calculation: it would be close to 50'000 in the case of C6000.

However, there is a step where square shape is, indeed, an unavoidable requirement, *i.e.* when matrices are to be diagonalized. In such a case, symmetry can have even more dramatic effects. When transformed to the SAMO basis, the full square matrices for (n, n) fullerenes decompose to ten blocks, each of them corresponding to an IR. As an example, block sizes range from 4 to 38 for C60, and from 670 to 3530 for C6000, to be compared with the number of AOs, N_{AO}, 840 and 84'000,

respectively. There are, thus, several relevant advantages in such a basis transformation; here the key quantity is $\mathcal{R}_3 = N_{AO}/M_{IR}$ (see Table I):

i) Memory requirement is dramatically reduced since M_{IR}^2 is the maximum number of matrix elements to be kept in memory, instead of N_{AO}^2, if all matrix diagonal blocks are stored to disc and retrieved one by one on need. Saving in memory usage is in the order of $(\mathcal{R}_3)^2$.

ii) Diagonalization of many small matrices is to be preferred to that of a larger matrix as the employed algorithms scale with the third power of the matrix size. Here the saving factor is in the order of $(\mathcal{R}_3)^3/N_{IR}$.

iii) Only one block of each IR needs to be diagonalized because of degeneracy; eigenvectors for all other rows of an IR are obtained by symmetry.

The largest block in the SAMO basis, M_{IR}, for these fullerenes corresponds to IR H_g in all cases, whose size is between $1/22$ (C60) and $1/24$ (C6000) of N_{AO} basis functions (Table I). Consequently, the requirement of memory for C60 is less than 12 KB per matrix, instead of more than 5 MB (double-precision real numbers) and, more relevantly, nearly 100 MB instead of 56 GB for C6000. Under the hypothesis of a third-power scaling algorithm for diagonalization (point ii above), gain in efficiency is also huge, as a speed-up of at least 1000 times is expected from a rough estimate in terms of $(\mathcal{R}_3)^3/N_{IR}$.

We now proceed in analyzing time performance and memory requirements. Beside reporting on the efficiency and scalability of single steps, we chose to consider also the time required for an SCF energy-plus-gradient calculation, in order to give a realistic representation of the power of the implemented algorithms. We refer to absolute CPU running time instead of purely relative data about scalability. Though widely arbitrary and strongly depending on the employed hardware, such a choice is meant to convey an idea of feasibility of such large size calculations at a good level of accuracy, even with low-level hardware. Calculations were run both on a single core (serial running) and on 8 cores (parallel running) so as to show the advantage of a change in the parallelization strategy of CRYSTAL14 with respect to CRYSTAL09: from the IBZ **k** points (with no effect for the present molecular case) to a finer task farming over the IRs (ten IRs, then ten cores at a time can be busy at the diagonalization step). In all the considered cases, wall-clock and CPU running times differ from each other by less than 1%.

Running times with a single core are represented graphically in Figure 3, and reported in Table S1 in the Supplementary Information. The various steps of a SCF+gradient calculation were illustrated in Section II. In particular, there are two steps that are preliminary to the SCF procedure, namely *initialize*, which includes
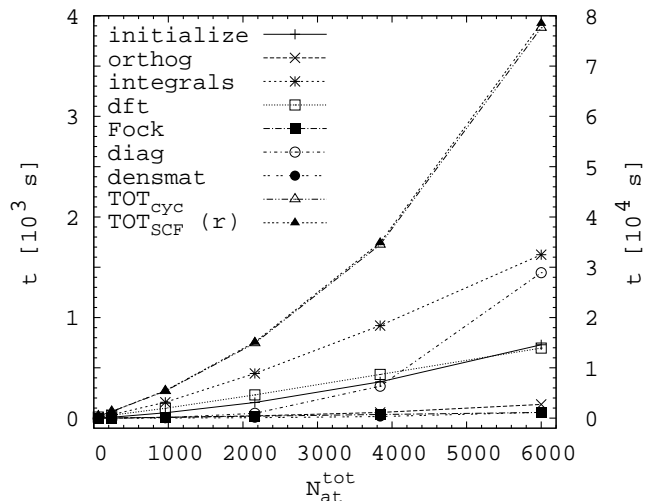

FIG. 3. Running time (seconds) for the various steps of total energy calculation of (n, n) fullerenes with one single core. Data refer to the following steps: 1) initialization of the simulation, including the construction of the symmetry group and transformation matrices (*initialize*); 2) orthogonalization of the basis set (*orthog*); 3) calculation of one- and two-electron integrals (*integrals*); 4) numerical integration of the exchange-correlation density functional (*dft*); 5) transformation of $\mathbf{f}$ into $\mathbb{F}$ (*Fock*); 6) Fock matrix block-diagonalization (*diag*); 7) construction of the density matrix (*densmat*); 8) a single SCF cycle (TOT_{cyc}, corresponding to a step 3-7 sequence); 9) the entire SCF procedure (TOT_{SCF}, 20 SCF cycles in this case). All calculations were performed on a cluster with 8 Intel Xeon E5620 cores working at 2.40 GHz and 2 GB of RAM per core. Note that data for TOT_{SCF} refer to the right (r) time axis.

the construction of tables for screening and symmetry analysis, and *orthog*, which refers to the orthogonalization of the AO basis set. In a direct-SCF strategy every SCF cycle is made up of the evaluation of the integrals (*integrals*), numerical integration of the exchange-correlation density functional (*dft*), transformation from $\mathbf{f}^{\mathbf{g}}$ to $\mathbb{F}^{\vec{k}}$ in block-diagonal form (*Fock*), diagonalization of every $\mathbb{F}^{\mathbf{k},\text{IR}}$ block (*diag*), construction of the density matrix and transformation to $\mathbf{p}^{\mathbf{g}}$ (*densmat*). TOT_{cyc} refers to a complete SCF cycle. All (n, n) fullerenes considered in the present study converged to an accuracy of 10^{-8} Ha in the total energy in about 20 SCF cycles; the corresponding total time is labeled as TOT_{SCF} in the tables. Calculation of the total energy gradient with respect to nuclear positions is also taken into account (*gradient*) as being an important step in structure optimizations and several other properties such as phonons, elastic tensors etc; note that the corresponding data are reported only in the Tables.

Perhaps the most relevant information from Figure 3 is that a complete SCF calculation for C6000 (with 84'000 AOs as a basis set) can be run with a single core within one day. As to be expected, the calculation of

one- and two-electron integrals and the integration of the exchange-correlation functional are the most time consuming steps in a single SCF cycle and time for diagonalization becomes comparable from (8,8) on, showing a clear trend to become dominant for larger size molecules (so that one can easily imagine how prohibitive diagonalization time would soon be if it were not performed in the SAMO basis). Linear transformations in step *Fock* that were introduced with the new algorithms implemented in CRYSTAL14 only account for about 2% of the time for a whole SCF cycle.

Also basis set orthogonalization largely benefits from the use of symmetry. Instead, time spent in preliminary analyses and classifications (*initialize*) tends to grow significantly with the system size as this step becomes dominated by the decomposition of the full reducible representation into irreducible representations and the formation of transformation matrices. This is, of course, a price to pay for the incommensurable advantage of using symmetry; it must be noticed that a single performance is required along one entire SCF calculation, which accounts for no more than 1% of the total time for the SCF calculation (when used for *normal* symmetries, with a maximum of 48 operators, the cost of this step is negligible[23]).

It is possible to estimate the scaling with system size by comparing the series including (1,1), (2,2), (4,4) and (8,8), along which the number of atoms increases by a constant factor of 4 (Figure 3 and Table S1 in the Supplementary Information). Up to the case of (4,4) with 960 atoms, scaling of the whole total energy calculation is essentially linear, going from 392 for (1,1), to 1312 for (2,2), to 5443 seconds for (4,4). For larger fullerenes scaling is worse than linear; for example from (4,4) to (8,8) the SCF time increases by a factor of 6.4 instead of 4. Steps that are mainly responsible for this behavior are *diag*, with its cubic scaling, and *integrals*, that shows a scaling $O(N_{at}^{tot\,1.27})$ from (4,4) to (8,8) due to a component with quadratic scaling.

When run in parallel with 8 cores (see Table S2 in the Supplementary Information), a complete calculation of the total energy plus the energy gradient for C6000 is accomplished within less than 5 hours, with an efficiency around 60%. Despite an excellent scaling (about 100%) of the *integrals* and *dft* steps with the number of cores, running time is here mainly determined by the *diag* step, whose efficiency is only 39%. The latter is affected by some unbalancing of tasks due to both the fact that $N_{IR}=10$ is not a multiple of the number of cores and, above all, the very different sizes of the various $\mathbb{F}^{\mathbf{k},IR}$, so that diagonalization of matrices with size M_{IR} are dominant. Steps *orthog*, *Fock* and *densmat* scale in a similar way (40÷60%) for the same reasons. A massive parallel approach would improve core balancing dramatically, but it has not been implemented in this case, yet. For (6,6) and smaller fullerenes, the diagonalization step is less relevant; in these cases scaling efficiency over the entire SCF procedure is higher than 80%.

Peaks in memory usage are reported in Table II.

The largest allocation for the largest fullerene occurs in *initialize* (1.5 GB), which is actually comparable to the amount of memory requested in the *Fock+diag* steps. Even if it does not correspond to the maximum memory reduction attainable in principle, such memory request still permits to run a calculation of a C6000 fullerene with 84'000 AOs using less than 2 GB per core.

TABLE II. Memory peaks (MBytes) in SCF+gradient calculations for (n, n) fullerenes. Labels as in Figure 3.

	(1,1)	(2,2)	(4,4)	(6,6)	(8,8)	(10,10)
N_{at}^{tot}	60	240	960	2'160	3'840	6'000
initialize	60	90	245	539	957	1500
orthog	17	69	201	422	757	1236
integrals	21	78	216	398	654	986
dft	19	69	185	368	624	956
Fock + diag	17	70	207	449	840	1432
densmat	17	69	204	435	798	1333
TOT_{SCF}	60	90	245	539	957	1500
gradient	63	188	314	481	725	1236

B. Carbon Nanotubes

Carbon nanotubes (CNTs) are an extraordinary example of the power of symmetry in the case of periodic systems. Here we consider a series of CNTs of the zigzag $(n, 0)$ family, with n ranging from 10 to 100 (see Table III). A graphical representation of the structure of the (100,0) nanotube is given in Figure 2 b. There are $4n$ atoms in the unit cell (40÷400 in the considered series) and $8n$ symmetry operators (80÷800), namely roto-translations and both longitudinal and transversal mirror planes. The number of atomic orbitals ranges from 560 to 5600 with the adopted basis set.

Unlike fullerenes, the number of symmetry operators increases with the nanotube size, so that N_{at}^{asym} is constant and equal to 1. This feature has a strong impact on efficiency, if used properly as in the case of our implementation. Indeed, the size of $\mathbf{m^g}$ matrices (see Table III) remains almost unaltered with increasing number of atoms. Still, it slightly decreases upon reaching the large radius regime, because interactions with the opposite side of the tube become negligible. For instance, the size $S_{\mathbf{fg}}$ of the symmetry irreducible Fock matrix, which is 2361 for (10,0), decreases to 2133 for (40,0) and to 2117 for (100,0). As $S_{\mathbf{Fg}}$ increases linearly with system size, the symmetry-related saving factor $\mathcal{R}_1$ grows linearly along the series, going from 54 for (10,0) to 545 for (100,0). As shown by index $\mathcal{R}_2$ the overall scaling with the system size is even better than linear, its value decreasing from 59 to 5 when going from (10,0) to (100,0), this being a consequence, again, of the nearly fixed size of $\mathbf{m^g}$ matrices.

TABLE III. General information about a family of $(n,0)$ carbon nanotubes. The number of irreducible atoms N_{at}^{asym} is 1 in all cases; N_{op} denotes the number of symmetry operators. The number of irreducible representations, N_{IR}, and the ratio $\mathcal{R}_3$ have distinct values for the Γ point, the zone boundary Z point and the other reciprocal space points (labeled with $\mathbf{k}$). All other symbols as in Table I. All nanotubes have the same value for M_{IR} (required to compute $\mathcal{R}_3$), that is 14 for Γ and 28 for Z and the other reciprocal space points.

$(n,0)$	N_{op}	N_{IR}			N_{at}^{tot}	N_{AO}	$S_{\mathbf{fg}}$	$S_{\mathbf{Fg}}$	$\mathcal{R}_1$	$\mathcal{R}_2$	$\mathcal{R}_3$		
		Γ	Z	$\mathbf{k}$							Γ	Z	$\mathbf{k}$
$(10,0)$	80	26	8	13	40	560	2361	127'900	54	59	40	20	20
$(20,0)$	160	46	13	23	80	1120	2181	235'960	108	27	80	40	40
$(40,0)$	320	86	23	43	160	2240	2133	464'240	218	13	160	80	80
$(60,0)$	480	126	33	63	240	3360	2133	696'360	326	9	240	120	120
$(80,0)$	640	166	43	83	320	4480	2133	928'480	435	7	320	160	160
$(100,0)$	800	206	53	103	400	5600	2117	1'154'200	545	5	400	200	200

Another very important consequence of increasing symmetry with the number of atoms is that the size of the blocks in $\mathbb{M}^{\mathbf{k}}$ is constant along the series. The largest blocks at the Γ point, the most symmetric case, contains 14×14 elements, but even at other $\mathbf{k}$ points the largest size is as small as 28×28. As a consequence, $\mathcal{R}_3$ increases linearly with system size, implying huge saving factors for both computational time and memory storage at the diagonalization step. Being the size of blocks constant, their number N_{IR} increases along the series. For instance, it varies from 26 to 206 in Γ and the trend is linear also at other $\mathbf{k}$ points (Table III).

Figure 4 and Table S3 in the Supplementary Information show timings for the CNT family on one core. The most time consuming steps for small tubes, namely the calculation of integrals and the integration of the exchange-correlation functional, become irrelevant for large tubes because of their excellent scaling with the number of atoms: in particular, the time required for the calculation of the integrals remains nearly constant as a consequence of the constant number of irreducible atoms N_{at}^{asym}, discussed above, so that the most time consuming steps for larger tubes are the linear transformation from AOs to SACOs basis ($Fock$) and the linear back transformation from SACOs to AOs basis during the reconstruction of the density matrix ($densmat$). Both these steps show a nearly quadratic behavior with the system size. The most important aspect here is the scaling of the diagonalization step: it is linear rather than cubic (as it would be in a calculation with no symmetry), this being a consequence of the size of the SACO blocks remaining constant, and of the number of diagonalizations to be performed increasing linearly. This is indeed the greatest advantage of using symmetry in nanotube calculations. Overall, a single SCF cycle shows linear scaling with system size up to (60,0), i.e. 240 atoms, so that for a (100,0) CNT only 120 seconds are required.

Preliminary steps of the calculation are more critical as basis set orthogonalization shows a quadratic behav-ior with the nanotube size, requiring about 25 seconds for (100,0). From (40,0) on, the most time consuming step of all the total energy calculation, accounting for more than 15% of the total time, is the initialization step, which includes decomposition of representations and construction of transformation matrices; the percentage raises to more than 50% for (100,0). This step is here inevitably more consuming than for fullerenes, due to the increasing number of symmetry operators along the series.

In the case of 8-core runs (Table S4 in the Supplementary Information), the total time for total energy calculation shows a scaling efficiency ranging between 68% and 82%. From (40,0) on, scaling of the total time is always lower than that for single SCF cycles, e.g. 70% vs 85% for (100,0), due to the low level of parallelism in the subroutines for symmetry classification ($initialize$
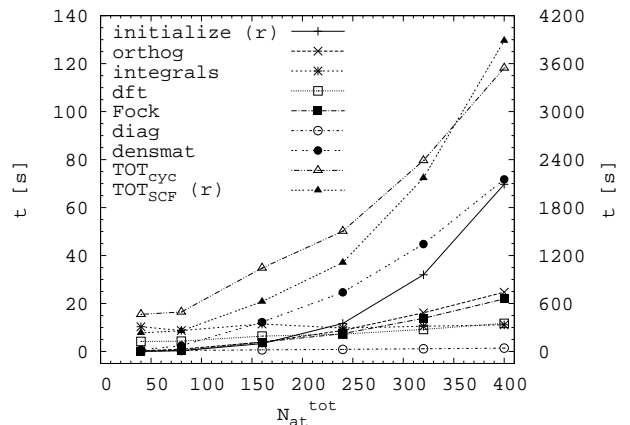

FIG. 4. Running time (seconds) for the various steps of total energy calculation of $(n,0)$ nanotubes with one single core. The SCF procedure consists of 15 cycles in this case. Note that data for $initialize$ and TOT_{SCF} refer to the right (r) time axis. Labels as in Figure 3.

TABLE IV. Memory peaks (MBytes) in SCF+gradient calculations for $(n,0)$ nanotubes. Labels as in Figure 3.

	(10,0)	(20,0)	(40,0)	(60,0)	(80,0)	(100,0)
N_{at}^{tot}	40	80	160	240	320	400
initialize	15	54	208	462	816	1270
orthog	11	13	20	29	37	46
integrals	15	19	29	41	52	64
dft	13	15	23	32	40	50
Fock + diag	11	13	20	29	37	47
densmat	11	13	20	29	37	46
TOT_{SCF}	15	54	208	462	816	1270
gradient	54	60	79	159	270	412

step, efficiency in the range 52÷61%). Within a single SCF cycle, the best scaling steps for large nanotubes are *Fock*, *densmat* and *diag*, with efficiencies of 89, 94 and 100% in the case of (100,0); their algorithms take full advantage of the increasing number of IR blocks.

A look at memory usage in Table IV reveals that in all cases the most memory consuming step is the inizialiation part, by requiring as much as 1.3 GB for (100,0). All other SCF steps require much smaller amounts of memory: the second most consuming is *integrals*, requiring 15÷64 MB along the series. The other steps require 11÷50 MB.

As a final note, in the present implementation the construction of irreducible representations and transformation matrices between AO and SACO bases remains a critical step for the simulation of nanotubes: its cost in terms of both memory and time becomes very large when the number of operators gets larger than 800-1000. It is possible to avoid exceeding this threshold by considering a sub-group of the actual symmetry group, e.g. the one formed by roto-translations only: in this way a full energy-plus-gradient calculation for a (320,0) nanotube (1280 atoms, 640 operators) requires about 2.5 GB RAM and 5 hours on a single core (see Ref. 17, Section "Symmetry and efficiency"). A new algorithm is currently under study, that scales less than with the third power of the number of operators.

V. CONCLUSIONS

We have presented a scheme for Self-Consistent-Field *ab initio* calculations, featuring full exploitation of translational and point group symmetries, and enabling high efficiency in many crucial steps: integrals calculation, integration of exchange-correlation functional, sampling of the reciprocal space, diagonalization of the Fock matrix, reconstruction of the density matrix. The formulation is general and suitable for both molecular and periodic calculations. Key features of the novel implementation include avoidance of full square matrices, direct transformation between AO and SACO bases and reconstruction

of the density matrix in the SACO basis.

Performance of the scheme as implemented in the CRYSTAL code is documented in terms of both computational time and memory requirements, in the case of two families of large, highly symmetric compounds, namely carbon fullerenes (up to 6000 atoms; point group with 120 symmetry operators) and nanotubes (up to 400 atoms and 800 operators). Calculation of integrals and integration of exchange-correlation functional are among the most time consuming steps for both families; perfect scaling with number of cores of these steps permits to take full advantage of parallel computing. Diagonalization step becomes a relevant step for large fullerenes; its poor scaling with number of cores makes it the bottleneck for this kind of systems. In nanotubes diagonalization is a minor step, thanks to exploitation of the increasing number of symmetry operators which keeps constant the size of matrices to diagonalize. As a drawback, basis set transformations take up a significant time for large nanotubes; again, parallel computing permits to solve the issue, thanks to the excellent scaling of this step with number of cores. Time scaling of an SCF cycle with system size is linear for fullerenes up to about 1000 atoms and for nanotubes up to about 250 atoms.

As a synthetic figure, a full energy-plus-gradient calculation for a nanotube with 400 atoms runs on a single core requiring 1.3 GB RAM and 70 minutes of computational time; more impressively, the same calculation for a fullerene with as many as 6000 atoms requires only 1.5 GB RAM and 24 hours of computational time.

[1] *TURBOMOLE V6.5 2013, a development of University of Karlsruhe and Forschungszentrum Karlsruhe GmbH, 1989-2007, TURBOMOLE GmbH, since 2007; available from* http://www.turbomole.com.

[2] M. Strain, G. Scuseria, and M. Frisch, Science **271**, 51 (1996).

[3] R. Stratmann, G. Scuseria, and M. Frisch, Chem. Phys. Lett. **257**, 213 (1996).

[4] J. Burant, G. Scuseria, and M. Frisch, J. Chem. Phys. **105**, 8969 (1996).

[5] J. Millam and G. Scuseria, J. Chem. Phys. **106**, 5569 (1997).

[6] G. Scuseria, J. Phys. Chem. A **103**, 4782 (1999).

[7] K. Kudin and G. Scuseria, Phys. Rev. B **61**, 16440 (2000).

[8] E. Schwegler and M. Challacombe, J. Chem. Phys. **105**, 2726 (1996).

[9] M. Challacombe and E. Schwegler, J. Chem. Phys. **106**, 5526 (1997).

[10] E. Schwegler, M. Challacombe, and M. Head-Gordon, J. Chem. Phys. **106**, 9708 (1997).

[11] M. Challacombe, J. Chem. Phys. **110**, 2332 (1999).

[12] E. Artacho, D. Sánchez-Portal, P. Ordejón, A. García, and J. M. Soler, Phys. Stat. Sol. (b) **215**, 809 (1999).

[13] P. Ordejón, Phys. Stat. Sol. (b) **217**, 335 (2000).

[14] A. Rusakov, M. Frisch, and G. Scuseria, J. Chem. Phys. **139**, 114110 (2013).

[15] R. Dovesi, Int. J. Quantum Chem. **29**, 1755 (1986).

[16] R. Dovesi, V. R. Saunders, C. Roetti, R. Orlando, C. M. Zicovich-Wilson, F. Pascale, B. Civalleri, K. Doll, N. M. Harrison, I. J. Bush, et al., *CRYSTAL14 User's Manual*, Università di Torino, Torino (2014).

[17] R. Dovesi, R. Orlando, A. Erba, C. Zicovich-Wilson, B. Civalleri, S. Casassa, L. Maschio, M. Ferrabone, M. De La Pierre, P. D'Arco, et al., Int. J. Quantum Chem. **114**, 1287 (2014).

[18] C. Zicovich-Wilson and R. Dovesi, Int. J. Quantum Chem. **67**, 299 (1998).

[19] C. Zicovich-Wilson and R. Dovesi, Int. J. Quantum Chem. **67**, 311 (1998).

[20] Y. Noël, M. De La Pierre, C. M. Zicovich-Wilson, R. Orlando, and R. Dovesi, Phys. Chem. Chem. Phys. **16**, 13390 (2014).

[21] Y. Nöel, P. D'Arco, R. Demichelis, C. M. Zicovich-Wilson, and R. Dovesi, J. Comput. Chem. **31**, 855 (2010).

[22] R. Demichelis, Y. Nöel, P. D'Arco, M. Rérat, C. M. Zicovich-Wilson, and R. Dovesi, J. Phys. Chem. C **115**, 8876 (2011).

[23] M. De La Pierre, R. Orlando, M. Ferrabone, C. M. Zicovich-Wilson, and R. Dovesi, Sci. China Chem. p. DOI: 10.1007/s11426-014-5191-y (2014).

[24] Supplementary material available on-line.

[25] C. M. Zicovich-Wilson, Y. Noël, A. M. Ferrari, R. Orlando, M. De La Pierre, and R. Dovesi, AIP Conf. Proc. **1456**, 248 (2012).

[26] C. Pisani, R. Dovesi, and C. Roetti, *Hartree-Fock ab-initio treatment of crystalline systems*, vol. 48 of *Lect. N. Chem.* (Spinger Verlag, Heidelberg, Germany, 1988).

[27] I. Bush, S. Tomic, B. Searle, G. Mallia, C. Bailey, B. Montanari, L. Bernasconi, J. Carr, and N. Harrison, Proc R. Soc. A **467**, 2112 (2011).

[28] R. Orlando, M. Delle Piane, I. Bush, P. Ugliengo, M. Ferrabone, and R. Dovesi, J. Comput. Chem. **33**, 2276 (2012).

[29] A. D. Becke, J. Chem. Phys. **98**, 5648 (1993).

[30] C. Lee, W. Yang, and R. G. Parr, Phys. Rev. B **37**, 785 (1988).

[31] P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch, J. Phys. Chem. **98**, 11623 (1994).

[32] L. Ge, B. Montanari, J. Jefferson, D. Pettifor, N. Harrison, and G. Briggs, Phys. Rev. B **77**, 235416 (2008).