Suplemento de la Revista Mexicana de Física **6** 011306 (2025) 1–7

On the numerical integration of two-particle functions for pair entropies of diatomic molecules

J. M. Solano-Altamirano

Facultad de Ciencias Químicas, Benemérita Universidad Autónoma de Puebla, 14 sur y Av. San Claudio, 72570, Puebla, Pue., México.

Received 2 March 2025; accepted 31 March 2025

In order to compute two-electron informational entropies of atoms or molecules, highly-accurate numerical integration methods are needed. In this contribution, we describe the details of a numerical algorithm specific for diatomic molecules, originally designed to numerically integrate 3D functions. The algorithm is adapted to integrate functions of two particles, *i.e.*, to integrate functions in domains of the form $\Omega \times \Omega$, where $\Omega \in \mathbb{R}^3$. The diatomic integration scheme is a cubature rule that combines Gauss-Legendre quadratures for the radial and angular parts, and the domain Ω is split into two semi-spheres, each with its own local center of coordinates. In addition, we compare the performance of the diatomic integration scheme *vs.* a Monte Carlo integrator, both for the 3D and 6D cases.

Keywords: Information theory; informational entropy; pair entropy; numerical integration; diatomic molecule; theoretical chemistry.

DOI: https://doi.org/10.31349/SuplRevMexFis.6.011306

1. Introduction

Since the seminal work of Shannon [1], the informational entropy has been used in many fields. For instance, in Quantum Chemistry, the Shannon Entropy in position space:

$$S_\rho = 1 \int \rho(\boldsymbol{r}) \ln \rho(\boldsymbol{r}) d^3r, \tag{1}$$

and in momentum space

$$S_{\tilde{\rho}} = 1 \int \tilde{\rho}(\boldsymbol{p}) \ln \tilde{\rho}(\boldsymbol{p}) d^3p, \tag{2}$$

has been used as a tool to measure the quality of the basis functions [2,3]. In Eqs. (1) and (2) $\rho(\boldsymbol{r})$ and $\tilde{\rho}(\boldsymbol{p})$ are the electron densities in position and momentum spaces, respectively. The entropy sum $S_T \equiv S_\rho + S_{\tilde{\rho}}$ has shown interesting properties such as reflecting the structure of the periodic table and it has also been used for measuring the quality of wave functions [4] and to study the effects of correlated methods upon the quality of $\rho(\boldsymbol{r})$ [2]. Furthermore, S_ρ has been used to measure the aromaticity in molecules [5]. These applications are only a very few selected examples and, for the interested reader, a recent review of the applications of the information theory in Chemistry can be found in Ref. [6].

More recently, the two-electron informational entropies are gaining attention from the scientific community, since these informational measures provide insight about the nature of two-particle correlations. However, so far the two-electron entropies, such as the pair entropy, have been studied in simple systems such as analytical quantum models [7] or atoms [8-10]. Again, this is only a very short list of two-electron entropy applications, and the interested reader may consult more extensive reviews, *e.g.*, Ref. [11].

The main obstacle for studying complex systems such as molecules has to do with the difficulty to compute the molecular two-electron density, $\rho_2(\boldsymbol{r}_1, \boldsymbol{r}_2)$, as well as the lack of ready-available methods to integrate functionals of $\rho_2(\boldsymbol{r}_1, \boldsymbol{r}_2)$, which are functions embedded in 6D spaces. In this context, the open-source suite DensToolKit [12] is capable of computing the $\rho_2(\boldsymbol{r}_1, \boldsymbol{r}_2)$ of molecules, regardless of its open- or closed-shell nature. In addition, the released version of DensToolKit includes numerical algorithms to integrate functionals of $\rho(\boldsymbol{r})$. In this contribution, we will describe our approach to specifically integrate functionals of $\rho(\boldsymbol{r})$ and the two-electron density function $\rho_2(\boldsymbol{r}_1, \boldsymbol{r}_2)$ for diatomic systems.

The organization of this contribution is as follows: In Sec. 2, we briefly describe the fundamental integrals that must be computed in the context of one- and two-electron informational entropies. In Sec. 3, we describe the details of the numerical integration scheme for diatomic molecules, which is useful to integrate functions in 3D domains, and compare the performance of the scheme against a Monte Carlo algorithm. In Sec. 4, we provide details of the extension of the scheme to 6D domains, so as to integrate two-electron densities of diatomic systems. Finally, we close the contribution with some conclusions.

2. The problem

Regardless of the one-electron informational measure we are interested in calculating, for an atom or a molecule, and for the purposes of this contribution, one must calculate integrals of the form:

$$I_1 = k \int_\Omega P_1(\boldsymbol{q}) d^3q. \tag{3}$$

Here, I_1 is a one-electron informational measure, k is a proportionality constant, Ω is a domain such that $\Omega \in \mathbb{R}^3$, and $P_1(\boldsymbol{q})$ is a one-electron density function that depends on $\boldsymbol{q}$. *E.g.*, for the Shannon entropy in momentum space of a

molecule $k=-1$, $\Omega=\mathbb{R}^3$, $\boldsymbol{q}=\boldsymbol{p}$ and $P(\boldsymbol{q})=\tilde{\rho}(\boldsymbol{p})\ln\tilde{\rho}(\boldsymbol{p})$. Other informational measures such as Fisher or Rényi entropies can be recovered from Eq. (3) through the appropriate choice of k, P_1, $\boldsymbol{q}$, and Ω.

On the other hand, two-electron informational measures are integrals of the form:

$$I_2 = K\int_{\Omega\times\Omega} P_2(\boldsymbol{q}_1,\boldsymbol{q}_2)d^3q_1d^3q_2. \tag{4}$$

For instance, by setting $K=-1$, $\Omega=\mathbb{R}$, $\boldsymbol{q}_1=\boldsymbol{r}_1$, $\boldsymbol{q}_2=\boldsymbol{r}_2$, and $P_2=\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)\ln\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)$, one obtains the pair entropy. Other quantities such as the mutual information or the Kullback-Leibler divergence are also included in Eq. (4), through the respective expressions of K, P_2, $\boldsymbol{q}$, and Ω.

2.1. 3D molecular densities

The most fundamental quantity that is commonly obtained from molecular orbital calculations is the electron density, $\rho(\boldsymbol{r})$, which can be expressed as a linear combination of molecular orbitals, $\chi_m(\boldsymbol{r})$:

$$\rho(\boldsymbol{r}) = |\psi(\boldsymbol{r})|^2 = \sum_{m=1}^{M} C_m\chi_m^*(\boldsymbol{r})\chi_m(\boldsymbol{r}). \tag{5}$$

Here, M is the number of occupied molecular orbitals, C_m is known as the occupation number, and

$$\chi_m(\boldsymbol{r}) = \sum_{\dot{A}=\dot{1}}^{\dot{N}} D_{m\dot{A}}\phi_{\dot{A}}(\boldsymbol{r}-\boldsymbol{R}_{\dot{A}}). \tag{6}$$

Equation (6) describes the most popular definition of a molecular orbital (MO), which is constructed by a linear combination of the so-called primitive functions $\phi_{\dot{A}}$. In Eq. (6), the coefficients that relate the m-th MO with the respective $\dot{N}$ primitive functions are denoted by $D_{m\dot{A}}$, and $\dot{A}=\dot{1},\ \dot{2},\ \ldots\ ,\ \dot{N}$. In the rest of this contribution, we will use dotted indices to identify sums over primitive functions. The most popular form of the primitive functions, and the form we will describe here, is that of a Gaussian function, centered at some nucleus (each primitive is associated with a nucleus), which can be written as:

$$\begin{aligned}\phi_{\dot{A}}(\boldsymbol{r}) =& (x^1-R^1_{\dot{A}})^{a^1_{\dot{A}}}(x^2-R^2_{\dot{A}})^{a^2_{\dot{A}}}(x^3-R^3_{\dot{A}})^{a^3_{\dot{A}}}\\ &\times\exp\left(-\alpha_{\dot{A}}(\boldsymbol{r}-\boldsymbol{R}_{\dot{A}})^2\right).\end{aligned} \tag{7}$$

Here, the integers $a^i_{\dot{A}}$ (i=1, 2, 3) fix the atomic orbital type (1 for s, 2 for p, and so on), and the number $\alpha_{\dot{A}}$ is the exponent of the primitive.

The molecular wave function, *i.e.*, the sets $\{C_m\}$, $\{D_{m\dot{A}}\}$, $\{\boldsymbol{R}_{\dot{A}}\}$, and $\{\alpha_{\dot{A}}\}$ can be obtained, for a given molecule, from many quantum chemistry programs such as Gaussian 09 [13], ORCA [14], Nwchem [15], Gamess [16], etc.

Finally, the electron density can be calculated as (see Refs. [17,18] and DensToolKit [12] manual for additional details):

$$\rho(\boldsymbol{r}) = \sum_{m=1}^{M} C_m\sum_{\dot{A}=\dot{1}}^{\dot{P}}\sum_{\dot{B}=\dot{1}}^{\dot{P}} D_{m\dot{A}}D_{m\dot{B}}\phi_{\dot{A}}(\boldsymbol{r})\phi_{\dot{B}}(\boldsymbol{r}), \tag{8}$$

where $\dot{P}$ is the total number of primitives used to describe the molecular wave function. Eq. (8) can be further reduced by contracting the coefficients C_m, $D_{m\dot{A}}$, and $D_{m\dot{B}}$, which yields:

$$\rho(\boldsymbol{r}) = \phi_{\dot{A}}(\boldsymbol{r})c_{\dot{A}\dot{B}}\phi_{\dot{B}}(\boldsymbol{r}). \tag{9}$$

In Eq. (9), and hereafter, unless otherwise specified, we will use the Einstein summation convention, and $c_{\dot{A}\dot{B}}$ is a matrix known as the density matrix.

Other density fields are obtained by first computing $\rho(\boldsymbol{r})$; hence, the computational cost of any 3D-density function is determined by the calculation of ρ or its derivatives $\nabla\rho$. As we commented, in our research group, we have developed the suite DensToolKit, which is an open source set of subprograms that can compute many one-electron density functions in 1D, 2D, and 3D grids [12]. Naturally, there are other great programs to compute one-electron densities, such as Multiwfn [19], ORBKIT [20], Critic2 [21], and so on.

2.2. 3D numerical integration schemes

In general, the main challenge when integrating one-electron densities includes, on the one hand, the large values of the respective density around the nuclei (see Fig. 1), and on the other hand, the fact that small values can sum up to 1-2 % of the integral. For instance, domain regions where $\rho\sim10^{-12}$ a.u. do contribute to the integral.

Furthermore, save for ρ (and only when the primitives are Gaussian functions), the integrals must be determined through numerical integration schemes, and most frequently, the integration is carried out using cubature (*i.e.*, 3D quadrature) rules.

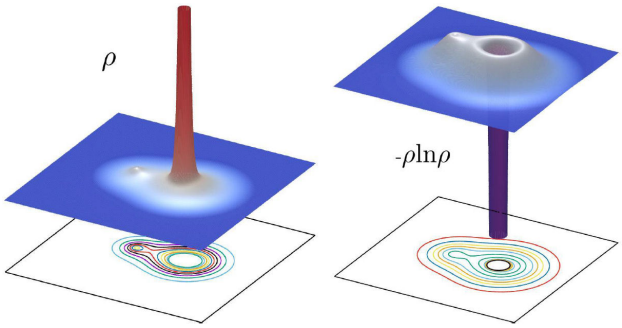

FIGURE 1. Electron density, $\rho(\boldsymbol{r})$, and Shannon entropy density, $-\rho(\boldsymbol{r})\ln\rho(\boldsymbol{r})$, of the HN molecule. Here, the fields are computed on the plane that intersects both nuclei.

3. Numerical integration 3D

In very rough terms, a cubature rule is a set of weights, $\{w_i\}$, and abscissas, $\{\boldsymbol{q}_i\}$, so that [22]:

$$I_1 = k \int_\Omega P(\boldsymbol{q}) d^3q \approx k \sum_i w_i P(\boldsymbol{q}_i). \tag{10}$$

Under this approximation, every improvement to the numerical scheme reduces to selecting the best change of coordinates together with the best standard quadrature rule(s) for each independent coordinate or coupled coordinates.[i]

For the purposes of having the possibly most-accurate integration scheme, in this contribution, we will describe the cubature schemes that we have implemented in DensToolKit [12], in particular, the integration of diatomic system densities.

3.1. Monoatomic systems

A most direct approach to design cubature rules consists of integrating each coordinate through independent quadrature rules. Anecdotally, for one-electron densities of single atoms, we have observed that decomposing the cubature into one radial quadrature and spherical-t designs [23] for the solid angles render better results with the same number of abscissas. Furthermore, the integration of one-electron densities in momentum space can always be carried out using this scheme, *i.e.*, as if the system was monoatomic, regardless of the number of atoms in the system. As a general procedure, it is best to reduce the number of points related to the angles, and to increase the number of abscissas associated with the radial part. Typical quadrature rules, such as the Gauss-Legendre quadratures, suffice to integrate the radial part.

Here, it is important to remark that, formally, $\Omega = \mathbb{R}^3$ for any molecule. However, for monoatomic systems, we have found that the semi-infinite domain in r can be replaced by a finite domain $0 \leq r \leq a$, such that $\rho(\boldsymbol{a}) \sim 10^{-14}$ a.u., and similarly by a domain wherein $0 \leq p \leq b$, where $\tilde{\rho}(\boldsymbol{b}) \sim 10^{-14}$ a.u. Here, $\tilde{\rho}(\boldsymbol{p})$ is the electron density in momentum space. In addition, if the system is azimuthally symmetric, then the quadrature rule related to the azimuthal angle can be omitted; this reduces the number of abscissas considerably.

3.2. Polyatomic systems

A very clever approach to integrate 3D densities of polyatomic molecules has been around in the literature for some time. Becke [24] divided the space using Voronoi cells (one per atom), and integrated each cell using quadrature rules for the radial part and a trapezoidal algorithm to integrate over the angles. To account for the polyhedron nature of the Voronoi cells, Becke used a smoothed step-functions, which removed abscissas outside the cell. This approach was further refined by Pérez-Jordá [25], so as to automatically choose an error tolerance, based on increasing (and reusing) the number of points over the radial part of the cubatures. This scheme has allowed to integrate many 3D functions of polyatomic molecules and it is implemented in several programs such as MultiWFN.

One small caveat of the Becke integration scheme is that it requires around 5 000 abscissas per atom to obtain a numerical precision of 10^{-4} in energy [25]. In contrast, a nice feature of the scheme is that, once implemented, it works for any polyatomic molecule.

3.3. Our approach to treat diatomic systems

While seeking for highly-accurate numerical integration methods, in 2015, we developed a cubature rule that reduces considerably the number of points for functions with azimuthal symmetry and that decay rapidly relative to the coordinates origin [26]. The method was originally implemented for integration domains with the shape of a sphere with two hollows, and basically, it consists of splitting the circular domain into two halves (the azimuthal angle is trivially integrated out to be 2π). Thereafter, the integration is performed upon the upper half, using the change of variables and over the shadowed region depicted in Fig. 2. Here, we denote the shadowed domain as Σ_2^u, ϕ_1 is the angle that is formed by the z axis and the vector $\boldsymbol{P}_1$, d_u is the distance between the primed coordinate origin and the center of the cavity, c is the distance between the non-primed coordinate origin and the center of the big circle, and a and b are the radii of the cavity and the big circle, respectively. The angle ϕ_1 divides the shadowed region into two parts. In other words, the integral over the spherical domain with two cavities (Σ_2) is reduced to compute the following integrals:

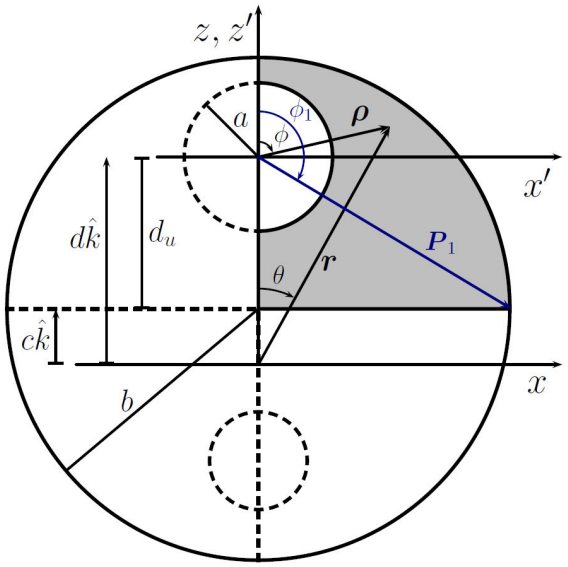

FIGURE 2. Diatomic upper domain decomposition (valence). The shadowed region is denoted as Σ_{2v}^u.

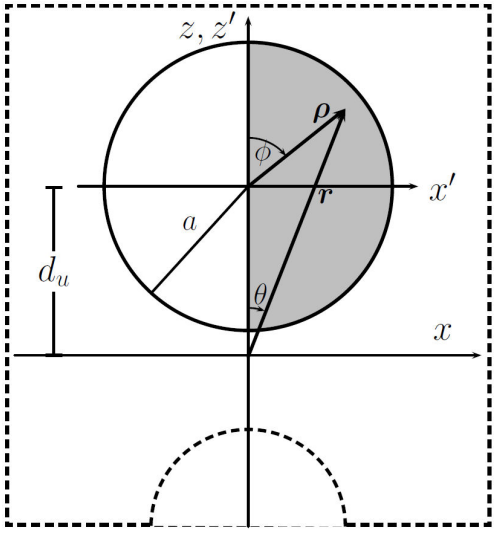

FIGURE 3. Diatomic upper domain decomposition (core), denoted in the text as Σ^u_{2c}.

$$\int_{\Sigma_2} f(r,\theta)dV = 2\pi \int_{\Sigma_2^u} f(r,\theta)\rho^2 \sin\phi d\phi d\rho$$
$$= 2\pi \int_0^{\phi_1} \sin\phi d\phi \int_a^{B(\phi)} \rho^2 f(\rho,\phi)d\rho$$
$$+ 2\pi \int_{\phi_1}^{\pi} \sin\phi d\phi \int_a^{C(\phi)} \rho^2 f(\rho,\phi)d\rho, \quad (11)$$

where

$$B(\phi) \equiv \sqrt{b^2 - d_u^2 \sin^2\phi} - d_u \cos\phi, \quad (12)$$

$$C(\phi) \equiv -\, d_u/\cos\phi. \quad (13)$$

For diatomic systems, we use the above numerical scheme to integrate one-electron densities over the "valence" regions, denoted as Σ^u_{2v} and Σ^l_{2v}, where u and l are for upper and lower, together with cubature rules for integrating the "core" regions, *i.e.*, the spherical domains centered at the upper and lower nuclei, denoted as Σ^u_{2c} and Σ^l_{2c}. In Fig. 3, the shadowed region is Σ_{2c}, and the scheme is valid when the function is azimuthally symmetric. Hence, for diatomic molecules, $\Sigma_2 = \Sigma^u_{2v} \cup \Sigma^u_{2c} \cup \Sigma^l_{2v} \cup \Sigma^l_{2c}$. In DensToolKit, the core valence is set to be 0.75 times the van der Waals radius of the respective atom and we choose b so that $\rho(\pm b\hat{z}) \leq 10^{-14}$ a.u. Finally, if the function is not azimuthally symmetric, *e.g.*, for open-shell systems, then a third quadrature rule is added to integrate over the azimuthal angle. After applying the changes of coordinates suggested in Eq. (11), we use standard Gauss-Legendre quadrature rules for each coordinate.

3.4. 3D benchmark

In this section, we compare our numerical integration rules against the Monte Carlo Miser algorithm (see, *e.g.*, Ref. [22]

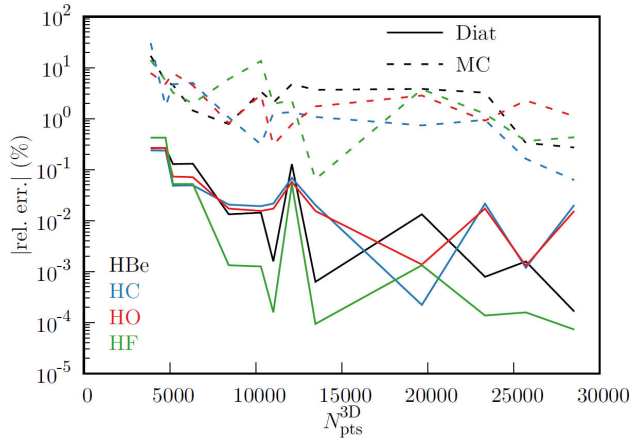

FIGURE 4. Absolute relative % errors of the integral $\int \rho dV = N_e$ *vs.* the number of integration points N^{3D}_{pts}, for several diatomic systems. Solid (dashed) lines correspond to the errors obtained with our integration scheme (Monte Carlo Miser algorithm), and color denotes the molecule.

for a specific implementation). We use as a benchmark the integral:

$$\int \rho(\boldsymbol{r})dV = N_e, \quad (14)$$

where ρ is the electron density of HBe, HC, HO, and HF, respectively, and N_e is the corresponding number of electrons, which is determined by using analytical overlap integrals $\left(\int \phi_{\dot{A}}\phi_{\dot{B}} d^3r\right)$. In Fig. 4, we depict the absolute % errors of the integral Eq. (14), for wave functions obtained from Gaussian09, using experimental separation distances between atoms, and at the HF/6-31G level of theory. Overall, with the same number of points we obtain improvements of 1-3 orders of magnitude, relative to Monte Carlo absolute errors, and the precision tends to increase as the number of integration points (N_{pts}) increases. Small variations of the trends (*e.g.*, large jumps or increasing —rel. err.— with increasing N_{pts} are due to changes between the number of points used in the radial or angular parts. For the results shown in Fig. 4, we used $N_{\text{pts},\rho}$ =12, 16, 26, or 34 abscissas for ρ; $N_{\text{pts},\phi}$ =18 or 22 for the polar angle (ϕ); and $N_{\text{pts},\varphi}$ =6 for the azimuthal angle. This renders (as it is implemented in DensToolKit) a total $N^{3D}_{\text{pts}} = 3N_{\text{pts},\rho} \times N_{\text{pts},\phi} \times N_{\text{pts},\varphi}$ (see Table I for specific combinations).

The diatomic integration scheme is already implemented in the program `dtkintegrate` of the suite DensToolKit 2.0, which can be obtained *via* github: `https://github.com/jmsolano/denstoolkit`.

4. Numerical integration 6D

Unfortunately, to integrate functions over a 6D space $\Omega \times \Omega$, there are no obvious reductions to the number of abscissas, since even systems with azimuthal symmetry in 3D lose this advantage once the two-electron densities are integrated.

In addition, the computational cost of computing the two-electron density is 2-5 times the time taken to compute ρ, de-

TABLE I. Number of points per coordinate ($N^{3\mathrm{D}}_{\mathrm{pts},\rho}$, $N^{3\mathrm{D}}_{\mathrm{pts},\phi}$, and $N^{3\mathrm{D}}_{\mathrm{pts},\varphi}$), as well as the total number of points ($N^{3\mathrm{D}}_{\mathrm{pts}}$ or $N^{6\mathrm{D}}_{\mathrm{pts}}$) and the absolute percentage relative errors shown in Figs. 4 and 5, for the systems HBe, HC, HO, and HF.

$N^{3\mathrm{D}}_{\mathrm{pts},\rho}$	$N^{3\mathrm{D}}_{\mathrm{pts},\phi}$	$N^{3\mathrm{D}}_{\mathrm{pts},\varphi}$	$N^{3\mathrm{D}}_{\mathrm{pts}}$	|% rel. err.| $\left(\int \rho(\boldsymbol{r})d^3r\right)$				$N^{6\mathrm{D}}_{\mathrm{pts}}$	|% rel. err.| $\left(\int \rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)d^3r_1d^3r_2\right)$			
				HBe	HC	HO	HF		HBe	HC	HO	HF
12	18	6	3888	0.2657	0.2379	0.2615	0.4220	15116544	0.3208	0.3110	0.1322	0.1701
12	22	6	4752	0.2669	0.2350	0.2640	0.4230	22581504	0.0888	1.3133	0.8658	0.4775
16	18	6	5184	0.1281	0.0482	0.0729	0.0521	26873856	0.1996	0.1911	0.6441	0.3684
16	22	6	6336	0.1308	0.0493	0.0710	0.0520	40144896	0.0336	0.1515	0.4636	0.0453
26	18	6	8424	0.0133	0.0205	0.0172	0.0013	70963776	0.2033	0.1312	0.0517	0.1487
26	22	6	10296	0.0143	0.0192	0.0154	0.0013	106007616	0.0527	0.0843	0.0990	0.0286
34	18	6	11016	0.0016	0.0215	0.0171	0.0002	121352256	0.0781	0.1107	0.0175	0.1086
16	18	14	12096	0.1281	0.0686	0.0571	0.0521	146313216	0.0438	0.0389	0.0504	0.1000
34	22	6	13464	0.0006	0.0202	0.0154	0.0001	181279296	0.0028	0.0050	0.0378	0.0567
26	18	14	19656	0.0133	0.0002	0.0014	0.0013	386358336	0.0311	0.0328	0.0363	0.0428
72	18	6	23328	0.0008	0.0214	0.0171	0.0001	544195584	0.0279	0.0087	0.0193	0.0148
34	18	14	25704	0.0016	0.0012	0.0013	0.0002	660695616	0.0313	0.0354	0.0551	0.0085
72	22	6	28512	0.0002	0.0201	0.0154	0.0001	812934144	0.0187	0.0408	0.0385	0.0079

pending on whether the system is open- or closed-shell [12]. This is so, because the two-electron pair density for closed-shell systems is given by:

$$\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2) = \frac{1}{2}\rho(\boldsymbol{r}_1)\rho(\boldsymbol{r}_2) - \frac{1}{4}\left[\Gamma_1(\boldsymbol{r}_1,\boldsymbol{r}_2)\right]^2, \tag{15}$$

where Γ_1 is the density matrix of order 1, which in terms of the primitives is:

$$\Gamma_1(\boldsymbol{r}_1,\boldsymbol{r}_2) = \phi_{\dot{A}}(\boldsymbol{r}_1)c_{\dot{A}\dot{B}}\phi_{\dot{B}}(\boldsymbol{r}_2). \tag{16}$$

The expression for computing $\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)$ is more complex for open shell systems (see Ref. [12] for details), as one needs to compute single-spin one-electron ρ's and Γ_1's (to take into account α and β contributions, separately).

Regardless of the issues discussed above, the integration over $\Omega\times\Omega$ for diatomic systems can be carried out using the diatomic decomposition described in Sec. 3.3. To this end, we use one cubature rule (built as described in Sec. 3.3) per each subset Ω, *i.e.*, we approximate 6D integrals as follows:

$$\begin{aligned}\int_{\Omega\times\Omega} P(\boldsymbol{r}_1,\boldsymbol{r}_2)d^3r_1d^3r_2 = &\int_0^{2\pi} d\phi_1 \int_{\Sigma_1} r_1^2\sin\theta_1 dr_1 d\theta_1 \times \\ &\int_0^{2\pi} d\phi_2 \int_{\Sigma_2} r_2^2\sin\theta_2 dr_2 d\theta_2 \times \\ &P(\boldsymbol{r}_1,\boldsymbol{r}_2) \\ \approx &\sum_{i,j} w_i w_j P(\boldsymbol{r}_i,\boldsymbol{r}_j).\end{aligned} \tag{17}$$

Unfortunately, we have found it mandatory to numerically integrate over the angles ϕ_1 and ϕ_2, even if the 3D molecule has azimuthal symmetry. This precludes decreasing the number of abscissas; however, the diatomic scheme can still render good numerical precisions, relative to Monte Carlo methods. Furthermore, our integration scheme is reliable, in the sense that it will always render the same integrated value if the calculation is repeated with the same $N_{\mathrm{pts},\rho}$, $N_{\mathrm{pts},\phi}$, and $N_{\mathrm{pts},\varphi}$, which amount to a total $N^{6\mathrm{D}}_{\mathrm{pts}} = \left(N^{3\mathrm{D}}_{\mathrm{pts}}\right)^2 = 9\left(N_{\mathrm{pts},\rho}\right)^2 \times \left(N_{\mathrm{pts},\phi}\right)^2 \times \left(N_{\mathrm{pts},\varphi}\right)^2$, for integrating two-electron densities over the domain $\Omega\times\Omega$. In this context and with the current DensToolKit version, the algorithm is not automatically adaptive. However, one may perform tests for computing the number of electron pairs, *i.e.*, $\int \rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)dV_1dV_2 = N_e(N_e+1)/2$, in order to select the best combination of $N_{\mathrm{pts},\rho}$, $N_{\mathrm{pts},\phi}$, and $N_{\mathrm{pts},\varphi}$; once selected, the same combination can be used for integrating other two-electron densities. As a general rule, if the molecule is open-shell, then a larger $N_{\mathrm{pts},\varphi}$ is required (see Table I for some specific cases and combinations).

Regarding the integrals in momentum space, as we advanced above, the integrals can be always computed over $V_1\times V_2$, where V_i is the volume of a sphere of radius P (in momentum space), such that $\max\left(\tilde{\rho}(\hat{e}^iP)\right) \approx 10^{-14}$ a.u., and $\hat{e}^1 = \hat{\imath}$, $\hat{e}^2 = \hat{\jmath}$, and $\hat{e}^3 = \hat{k}$. Empirically, we have found that integrals in momentum space require using $N_{\mathrm{pts}\pi} > 100$, in order to render numerical precisions of $\sim 10^{-3}$.

4.1. 6D benchmark

In Fig. 5, we show the absolute % errors of the integral $\int \rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)dV_1dV_2 (= N_e(N_e+1)/2)$ *vs.* $N_{\mathrm{pts}} (= N_{\mathrm{pts},\rho}\times N_{\mathrm{pts},\phi}\times N_{\mathrm{pts},\varphi})$.

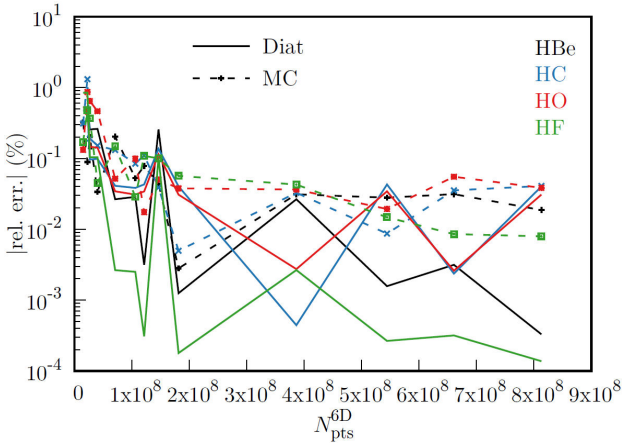

FIGURE 5. Absolute relative % errors of the integral $\int \rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)dV_1dV_2 = N_e(N_e+1)/2$ *vs.* the number of integration points $N_{\text{pts}}^{6\text{D}} = \left(N_{\text{pts}}^{3\text{D}}\right)^2 = 9\left(N_{\text{pts},\rho}\right)^2 \times \left(N_{\text{pts},\phi}\right)^2 \times \left(N_{\text{pts},\varphi}\right)^2$, for several diatomic systems. Solid (dashed) lines correspond to the errors obtained with our integration scheme (Monte Carlo Miser algorithm), and color denotes the molecule.

In contrast with the 3D case, Monte Carlo methods are an acceptable choice for integrating 6D functions. However, as we commented previously, our scheme still offers a reliability advantage, in the following sense. After selecting an appropriate set of $N_{\text{pts},\rho}$, $N_{\text{pts},\phi}$, and $N_{\text{pts},\varphi}$, other properties can be integrated expecting that the same accuracy is obtained. (Let us recall that Monte Carlo methods are random in nature; thus, there is no guarantee that repeating the calculation with the same number of points will always render the same accuracy.) In addition, our integration scheme still might render smaller —rel. err.— by up to 2 orders of magnitude, relative to the Monte Carlo Miser variant.

Regarding the computational cost, as we commented previously, the most expensive calculation is to evaluate $\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)$, thus the computational cost of integrating any function of the form shown in Fig. 4 is directly proportional to the number of integration points. The cost of computing the abscissas and weights is negligible, compared with the computational cost of computing $\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)$ at the set of abscissas. For further time profiling details, see Ref. [12].

The reliability that our scheme may achieve makes it a convenient numerical method to study two-electron density functions, in particular, any two-electron informational entropy in the near future.

5. Conclusions

In this contribution, we have described a numerical scheme, specialized to integrate 3D and, in particular, 6D functions for diatomic molecules. The scheme was designed specifically to integrate molecular one-electron (3D) or two-electron (6D) density fields. These fields are, in essence, functionals of the one-electron density $\rho(\boldsymbol{r})$ or the two-electron pair density function $\rho_2(\boldsymbol{r}_1,\boldsymbol{r}_2)$. We have presented benchmarks of our scheme and compared them against integrations performed with the Monte Carlo Miser method. In general, one may decrease errors by 1-2 orders of magnitude, for a given number of integration points.

We are currently working on implementing algorithms to integrate 3D and 6D fields for polyatomic molecules, which will be released in future versions of DensToolKit.

The scheme presented here, together with the current capabilities of the program DensToolKit [12], constitutes a numerical tool that will be useful in studying two-electron informational entropies of diatomic molecules in the near future.

Acknowledgements

JMSA acknowledges the organizing board of the symposium *Applications of Information Theory in Natural Sciences*.

---

i. Notice that Monte Carlo integration schemes are also described by Eq. (10).

1. C. E. Shannon, A Mathematical Theory of Communication, Bell Syst. Tech. J. **27** (1948) 379, https://doi.org/10.1002/j.1538-7305.1948.tb01338.x
2. M. Hô *et al.*, A numerical study of molecular information entropies, *Chem. Phys. Lett.* **219** (1994) 15, https://doi.org/10.1016/0009-2614(94)00029-8.
3. H. G. Laguna, S. J. C. Salazar, and R. P. Sagar, Information theoretical statistical discrimination measures for electronic densities, *J. Math. Chem.* **60** (2022) 1422, https://doi.org/10.10.1007/s10910-022-01363-6.
4. S. R. Gadre *et al.*, Some novel characteristics of atomic information entropies, *Phys. Rev. A* **32** (1985) 2602, https://doi.org/10.1103/PhysRevA.32.2602
5. S. Noorizadeh and E. Shakerzadeh, Shannon entropy as a new measure of aromaticity, Shannon aromaticity, *Phys. Chem. Chem. Phys.* **12** (2010) 4742, https://doi.org/10.1039/B916509F
6. D. S. Sabirov and I. S. Shepelevich, Information Entropy in Chemistry: An Overview, *Entropy* **23** (2021) 1240, https://doi.org/10.3390/e23101240
7. S. J. C. Salazar-Samaniego, Correlaciones estadísticas de órdenes altos en sistemas cuánticos de tres partículas, Phd thesis, Universidad Autónoma Metropolitana, CDMX (2022), Available at. https://doi.org/10.24275/uami.sf268539b
8. N. L. Guevara, R. P. Sagar, and R. O. Esquivel, Information uncertainty-type inequalities in atomic systems, *J. Chem. Phys.* **119** (2003) 7030, https://doi.org/10.1063/1.1605932
9. R. P. Sagar, H. G. Laguna, and N. L. Guevara, Electron pair density information measures in atomic systems, *Int. J.*

Quantum Chem. **119** (2011) 3497, https://doi.org/10.1002/qua.22792

10. S. López-Rosa *et al*., Electron-pair entropic and complexity measures in atomic systems, *Int. J. Quantum Chem*. **119** (2019) e25861, https://doi.org/10.1002/qua.25861
11. F. J. Torres *et al*., A review on the information content of the pair density as a tool for the description of the electronic properties in molecular systems, *Int. J. Quantum Chem*. **119** (2019) e25763, https://doi.org/10.1002/qua.25763
12. J. M. Solano-Altamirano, *et al*., DensToolKit2: A comprehensive open-source package for analyzing the electron density and its derivative scalar and vector fields, *J. Chem. Phys*. **161** (2024) 232501, https://doi.org/10.1063/5.0239835
13. M. J. Frisch *et al*., Gaussian 09 Revision D.01, Gaussian Inc. Wallingford CT 2009.
14. F. Neese, Software update: the ORCA program system, version 4.0, *WIREs Comput. Mol. Sci.* **8** (2018) e1327, https://doi.org/10.1002/wcms.1327
15. M. Valiev *et al*., NWChem: A comprehensive and scalable open-source solution for large scale molecular simulations, *Comput. Phys. Commun*. **181** (2010) 1477 , https://doi.org/10.1016/j.cpc.2010.04.018
16. M. W. Schmidt *et al*., General atomic and molecular electronic structure system, *J. Comput. Chem.* **14** (1993) 1347, https://doi.org/10.1002/jcc.540141112.
17. R. G. Parr and W. Yang, Density-functional theory of atoms and molecules (Oxford University Press, 1989).
18. J. Andrés and J. Beltrán (Eds.), Química teórica y computacional (Publicacions de la Universitat Jaume I, 2000), Ch. 2.
19. T. Lu and F. Chen, Multiwfn: A multifunctional wavefunction analyzer, *J. Comput. Chem.* **33** (2012) 580, https://doi.org/10.1002/jcc.22885
20. G. Hermann *et al*., ORBKIT: A modular python toolbox for cross-platform postprocessing of quantum chemical wavefunction data, *J. Comput. Chem*. **37** (2016) 1511, https://doi.org/10.1002/jcc.24358.
21. A. O. de-la Roza, E. R. Johnson, and V. L. na, Critic2: A program for real-space analysis of quantum chemical interactions in solids, Comput. *Phys. Commun*. **185** (2014) 1007, https://doi.org/10.1016/j.cpc.2013.10.026
22. W. Press *et al.*, Numerical Recipes, The Art of Scientific Computing, 1st ed. (Cambridge University Press, 1989)
23. C. An and Y. Xiao, Numerical construction of spherical t-designs by Barzilai-Borwein method, *Appl. Numer. Math*. **150** (2020) 295, https://doi.org/10.1016/j.apnum.2019.10.008
24. A. D. Becke, A multicenter numerical integration scheme for polyatomic molecules, *J. Chem. Phys*. **88** (1988) 2547, https://doi.org/10.1063/1.454033
25. J. M. Pérez-Jordá, A. D. Becke, and E. San-Fabián, Automatic numerical integration techniques for polyatomic molecules, *J. Chem. Phys.* **100** (1994) 6520, https://doi.org/10.1063/1.467061
26. J. Solano-Altamirano and S. Goldman, Thermodynamic stability in elastic systems: Hard spheres embedded in a finite spherical elastic solid, *Eur. Phys. J. E* **38** (2015) 133, https://doi.org/10.1140/epje/i2015-15133-1
27. W. Zou and C. Gao, Molden2AIM, https://github.com/zorkzou/Molden2AIM (2015).