A Program for Single-center Expansion in Laguerre-type Orbitals for the Hydrogen Molecular Ion

Yasuyo HATANO,*,a Shigeyoshi YAMAMOTO b

a School of Information Sciences and Technology, Chukyo University, Toyota 470-0393, Japan
b School of International Liberal Studies, Chukyo University, Nagoya 466-8666, Japan

*e-mail: hatano@sist.chukyo-u.ac.jp

(Received: February 19, 2016; Accepted for publication: May 20, 2016; Online publication: September 3, 2016)

A Fortran program is given for calculating wave functions of the molecular hydrogen ion, expanded in terms of single-center Laguerre-type orbitals. Using this program, the radial quantum number has been extended to 203, and accuracy has been attained in energy of the order of 10^-6 a.u. The electron-nucleus Coulomb integrals are evaluated numerically by applying Gaussian quadratures.

Keywords: Associated Laguerre polynomials, Gaussian quadratures, Hydrogen molecular ion, Complete orthonormal basis set, Single-center expansion

1 INTRODUCTION

In modern quantum chemical calculations, Gaussian-type orbitals (GTOs) [1] are most commonly used as basis functions. To perform accurate calculations for atoms or small molecules, Slater-type orbitals (STOs) [1] are also utilized. GTOs provide an advantage in efficient multi-center 2-electron integral evaluation. As a result of overlap integrals among these basis functions, however, problems associated with linear dependence [2,3] occasionally prevent accurate computation. This may be a severe impediment when extending the basis toward completeness. In this paper we apply Laguerre-type orbitals (LTOs) [1] which do not suffer from linear dependence issues.

In Hartree-Fock calculations with basis-set expansions, the wave functions obtained often have artificial nodes [4–6], which should not appear in the exact wave function. Artificial nodes have been found for GTOs [5,6] and STOs [4,5]. It is of interest whether artificial nodes can occur even for orthogonal basis sets such as LTOs having no overlap integrals. This question motivated us to write the present program involving LTOs.

The Laguerre-type orbitals give rise to a complete orthonormal basis (CONS) [1,7,8]. It is well known that LTOs are complete sets for bound states of the hydrogen atom. By adding increasing numbers of LTOs, wave functions derived using this method approach the exact solution ever more closely. LTOs nevertheless have shortcomings. The biggest problem, common to all exponential-type orbitals (ETOs), is in evaluating multi-center Coulomb integrals. For this reason, LTOs are applied only to single-center expansion, and actual single-center expansion calculations have been limited to small systems such as the He atom [2,3], H2+ [9,10], or H2 [11].

The hydrogen molecular ion (H2+) is the simplest molecule and has been studied from the early days of quantum chemistry, beginning with the pioneering works of Burrau [12] and Hylleraas [13]. The exact solution was obtained. Peek’s paper [14] presently provides the most precise value of the exact electronic energy, −1.1026342144949 a.u., for the internuclear distance $R_{\text{band}} = 2.0$ a.u. using nonrelativistic theory and the Born-Oppenheimer approximation. This internuclear distance, which is quite close to the equilibrium distance of ($R_e = 1.9971933199$ a.u.) [15], is used throughout this paper. Bates et al. [16] gave details of the exact wave function including contour maps and the peak value (0.458) at the nucleus. Even today H2+ is an important molecular system used as a testing ground for novel numerical methods and approaches.

Single-center expansion is usually based on the molecular mid-point, but its convergence is slow. Because no basis functions are placed on the nuclear positions, it will be difficult to give satisfactory results for the nuclear cusp condition [17] if a finite number of basis functions is used. This explains the slow convergence.

Single-center expansions with LTOs have been made before for H2+: the first was by Howell and Shull [9], and the second by Kranz and Steinborn [10]. In the latter, the maximum number (Nmax) of the radial quantum number ($n$) in the basis is 25 and the resulting electronic energy (−1.1022 a.u.) is accurate to the order of 10^-4 a.u. This contrasts with the rapid convergence of the multi-center expansion method (linear combination of atomic orbitals, LCAO) using GTOs. Wells and Wilson [18] quote an energy accuracy of 2 x 10^-7 a.u. with a reasonably compact basis set of [23s 9p 4d 4f 2g] (232 primitive GTOs in total).

In spite of the above shortcomings of LTOs, they are still valuable for their capability to construct a complete orthogonal basis. In testing newly developed methods on the hydrogen molecular ion, it is desirable to lift the limit for the radial quantum number ($n$) and attain accuracy in energy of the order of 10^-6 a.u.
In calculations with LTOs, the most time-consuming step is the evaluation of electron-nucleus Coulomb integrals. Although analytical integration is possible, there is serious loss of accuracy, as will be described in section 3.1. To overcome this problem a multiple-precision arithmetic package such as MPfun90 [19] or FM [20] is required.

To evaluate integrals efficiently and precisely we wrote a program that implements numerical integration by Gaussian quadratures [21–23]. We examined the accuracy of the integrals and found that, for $n = 203$, at least 20 decimal digits are maintained as significant figures. Using this program we have successfully evaluated electron-nucleus Coulomb integrals. Although one method is required.

The program is written in Fortran 2003. It is self-contained, so that no external library programs are needed. The computation is performed with quadruple precision arithmetic based on the IEEE Standard for Floating-Point Arithmetic [24], which is supported by most modern Fortran compilers. As a result, the program has good portability.

Studies of $H_2^+$ continue and recent theoretical work can be found in the literature [25–27].

2 FORMULATION

The wave function for $H_2^+$ is expanded in terms of basis functions $\varphi_{nlm}$ in polar coordinates, as

$$\Psi(r, \theta, \phi) = \sum_{nlm} \sum_{\nu} C_{nlm} \varphi_{nlm}(r, \theta, \phi),$$

(1)

$$\varphi_{nlm}(r, \theta, \phi) = R_n(r) Y_{nm}(\theta, \phi)$$

(2)

Here, $Y_{nm}$ is a spherical harmonic function. The expansion coefficients $C_{nlm}$ are obtained by the variational principle,

$$\delta \int \Psi^* H \Psi \, dv = 0,$$

(3)

$$H = -\frac{1}{2} \nabla^2 - \frac{Z}{|r - R_A|} - \frac{Z}{|r - R_B|} - \frac{Z^2}{|R_A - R_B|}$$

(4)

Here $H$ is the Hamiltonian, and $R_A$ and $R_B$ are the position vectors of nuclei A and B. $Z$ is the nuclear charge. For $H_2^+$, $Z = 1$. The fourth term, the nuclear repulsion energy, is constant in the Born-Oppenheimer approximation and will be omitted henceforth. As a result we consider the electronic energy (EE) rather than the total energy (TE). Atomic units are used throughout.

We adopt a single-center expansion based on the molecular midpoint for the wave function. Laguerre-type orbitals (LTOs) [1] are used for the radial functions,

$$R_n(r) = \left(2z_0^{1/2}\sqrt{\pi} \right)^{-1/2} \frac{z_0^{21/2}}{\Gamma(n + l + 1)} (2z_0r)^l L_n^{21/2}(2z_0r)e^{-z_0r}$$

(5)

Here $L_n^{21/2}(2z_0r)$ is an associated Laguerre polynomial; $z_0$ is called the exponent. The polynomial is written as

$$L_n^m(x) = \sum_{k=0}^{n} \frac{(n + m)!}{k!(n - k)!} \frac{(-x)^k}{(m + k)!}$$

(6)

These polynomials satisfy the following recurrence relation:

$$nL_n^m(x) = (2n + m - 1 - x)L_{n-1}^m(x) - (n + m - 1)L_{n-1}^{m-1}(x),$$

(7)

$$L_0^0(x) = 1,$$  

$$L_1^0(x) = x + 1.$$  

(8)

The Hamiltonian element consists of three terms,

$$\int \varphi_{nlm}^* H \varphi_{nl'm'} \, dv = T(nlm, n'l'm') + V_s(nlm, n'l'm') + V_s(nlm, n'l'm')$$

(9)

The first term, representing the kinetic energy, is evaluated according to the following formula, given in an Appendix (Eq.7) of the paper by Hagstrom and Shull [11],

$$T(nlm, n'l'm') = (nlm) - \frac{1}{2} \nabla^2 |n'l'm'|$$

$$= z_0^2 \frac{\delta_{nm} \delta_{nm}}{2l + 3} \left\{ \frac{(n + l + 1)(n' + l + 1)!}{2(l + 3)!} \right\}^2 \left\{ (n - l + 1)(n' + l + 1)! \right\}$$

(10)

The second term in the Hamiltonian,

$$V_s(nlm, n'l'm') = \int \varphi_{nlm}^* \left( \frac{1}{|r - R_A|} \right) \varphi_{nl'm'} \, dv,$$

(11)

represents the electron-nucleus Coulomb energy. The Coulomb term is expanded in terms of Legendre polynomials as

$$\frac{1}{|r - R_A|} = \sum_{l=0}^\infty \frac{r_{l}}{r_{l+1}} P_l(\cos \theta_A),$$

(12)

where $r_{l} = \min (r, R_A)$, $r_{l} = \max (r, R_A)$. The second and third terms in the Hamiltonian are the most difficult to calculate.

The Hamiltonian is nonzero only for $m = 0$. For $H_2^+$ ($1s$ $\sigma$), both $l$ and $l'$ are even. By substituting $R$ and $V$ for $R_A$ and $V_s$, $V$ can be written as follows:
### 3 NUMERICAL INTEGRATION

#### 3.1 Term-by-term analytical integration

By inserting Eq.6 into Eq.16 and Eq.17, the integrals $F$ and $G$ can be written as the sum of the individual terms:

\[
F(n,l,n',l';R) = \int_{0}^{\infty} \frac{e^{i\int f_{0}^{R} w(x) f(x) dx}}{R^{l'+2}} dr
\]

\[
G(n,l,n',l';s,R) = \left( \frac{F(n,l,n',l';s,R)}{R^{l'+2}} + R^{l'+2} G(n,l,n',l';s,R) \right) \times \frac{(n-l-1)!}{(n+l+1)!} \frac{(n-l'+1)!}{(n+l'+1)!}
\]

#### 3.2 Gaussian quadratures

For numerical integration we adopt the Gaussian quadrature formula [21–23].

\[
\int_{a}^{b} h(x) dx = \sum_{k=1}^{M} W_{k} f(x_{k}) + \text{error}
\]

Here $w(x)$ is a weighting function, $x_{k}$ are abscissas, $W_{k}$ are weights, and $M$ is the number of abscissas.

The integrands of Eq.16 and Eq.17 are plotted in Figure 1 and Figure 2, respectively. These figures reveal that the integrands oscillate frequently for large $n$ or $n'$, and that as many abscissas as the frequency should be taken in order to provide sufficient accuracy for evaluation of the integrals.

Since the error in Gaussian quadrature can be made zero for $G$, but not for $F$, we describe the numerical integration of $G$ below, first to simplify the discussion. Later we shall consider $F$.

Equation 17 can be changed to the following form by transformation of variable:

\[
\int_{a}^{b} g(x) e^{-x} dx = \int_{a}^{b} \sum_{k=1}^{M} W_{k} f(x_{k}) + g(x) e^{-x} dx
\]
to which we apply the Gauss-Laguerre quadrature rule \([21]\) on \((a,b) = (0,\infty)\) with the weighting function \(w(x) = e^{-x}\).

\[
G(n,l,n',l',s,R) = \sum_{k=1}^{N} W_k g(x_k) + \text{error}
\]

\[
g(x) = \frac{1}{z_i + z_j} e^{-(x+z_i-z_j)} \left( \frac{x}{z_i + z_j} + R \right)^{2z_i+1} \left( 2z_i \left( \frac{x}{z_i + z_j} + R \right) \right) \times L_{2z_i}^{2z_i+1} \left( 2z_j \left( \frac{x}{z_i + z_j} + R \right) \right)
\]

\[
J = 2s + n - l + n' - l' - 1.
\]

We next checked the accuracy of the integrals. The results of the test are listed below to 33 decimal places, taking the case of \(n=n'=151, l=l'=0, s=0, z_i=z_j = 37.65\) as an example. For this check we wrote a test program by modifying subroutines in the mathematical library NUMPAC \([29,30]\). The numerical integration was performed with quadruple precision arithmetic. The result from the analytical integration with 158 significant digits specified for MPfun90 is quoted below as scheme (A). In the scheme (B)/(C), 166/173 digits are specified.

\[
G(n,n',l,l';s,R) = \sum_{k=1}^{N} W_k f(x_k) + \text{error}
\]

\[
f(x) = x^{(n+l'+s-1)} \left( \frac{2}{z_i} \right)^{2z_i+2} L_{2z_i}^{2z_i+2} \left( 2z_j x e^{-z_i} \right)
\]

\[
J = l + l' - 2s + n + n'
\]

We turn now to the evaluation of \(F\) (Eq. 16). The Gauss-Legendre quadrature rule \([23]\) is used, with the weighting function \(w(x) = 1\).

\[
J = \left( \frac{2s + n - l + n' - l' - 1}{2} \right) / 2n
\]

We tested also the double of this value,
\[
M = l + l' - 2s + n + n' + 1 \tag{32}
\]

We list below one of the results of the trial calculations.

For \( n = n' = 125, l = l' = 20, s = 5, z_l = z_{l'} = 114.98528. \)

(a') Gauss 141 points
\[0.2878459649515099317492932366032297E-16\]

(b') Gauss 281 points
\[0.2878459649515099317492932366032297E-16\]

(A') MPfun90 158 digits
\[0.2878459649515099317492932366032297E-16\]

(B') MPfun90 166 digits
\[0.2878459649515099317492932366032297E-16\]

(C') MPfun90 173 digits
\[0.2878459649515099317492932366032297E-16\]

The scheme (a') gives only 17 significant digits, whereas the scheme (b') gives 28 digits. To obtain the desired accuracy, scheme (b') is needed. We therefore adopt scheme (b'), i.e., we set \( M = J + 1. \) Since \( J \) reaches its maximum when \( l = l' = L_{\text{max}} \) and \( s = 0, \) the maximum \( M \) is given by \( 2 \times N_{\text{max}} + 2 \times L_{\text{max}} + 1. \) For \( N_{\text{max}} = 203 \) and \( L_{\text{max}} = 202, M \) takes the value 811. Here \( L_{\text{max}} \) is the maximum of \( l \) or \( l' \) in the basis.

The abscissas (zeros of the Legendre polynomials \( P_m(x) \)) are calculated using Newton’s iterative method, starting from approximate roots given by

\[
x_k = \sin\left(\frac{M + 1 - 2k}{2M + 1}\pi\right) \quad (k = 1, 2, \ldots, M). \tag{33}
\]

The weights \([23], W_k,\) are calculated from the following formula:

\[
W_k = \frac{2(1 - x_k^2)}{[M_{p(x)}]_k^2} \tag{34}
\]

To conclude this subsection, we compare the Coulomb integral \( V \) obtained from the Gaussian quadrature scheme (a and b') with that evaluated from Eq.13 using MPfun90 [19].

\( V \) for \( n = n' = 151, l = l' = 0, z_l = z_{l'} = 37.65. \)

Gaussian quadratures
\[
\begin{align*}
0.43933850711936265787621563083154 & \\
\text{MPfun90 158 digits} & \\
0.43933850711936265787621569152698 & \\
\text{MPfun90 166 digits} & \\
0.43933850711936265787621563540349 & \\
\text{MPfun90 173 digits} & \\
0.43933850711936265787621563540349 & 
\end{align*}
\]

\( V \) for \( n = n' = 151, l = l' = 150, z_l = z_{l'} = 37.65. \)

Gaussian quadratures
\[
\begin{align*}
0.253398692737192741982923026971287 & \\
\text{MPfun90 158 digits} & \\
0.253398692737192741982923026971291 & 
\end{align*}
\]

Further tests confirm at least 20-figure accuracy for \( N_{\text{max}} \leq 203. \) The cost of the numerical integration is \( O(N_{\text{max}}). \)

4 PROGRAM STRUCTURE

4.1 Calling sequence

The program system provided is self-contained, so that other library routines are not necessary. The calling sequence of the major routines is shown in Figure 3. For Gauss-Legendre quadrature, subroutine TGLEGQ of NUMPAC library [29,30] is incorporated. For diagonalizing the Hamiltonian matrix to obtain EE, subroutine HOBSVQ of the NUMPAC library [29,30] is used.

4.2 Array size

The memory size required to run the program is determined by one large array named HMATRIX, where the Hamiltonian elements are kept. The size of HMATRIX is \( NN_{\text{QH}} \times NN_{\text{QH}}, \) while \( NN_{\text{QH}} = N_{\text{max}} \times (N_{\text{max}}+1)/4 + N_{\text{max}}. \) With \( N_{\text{max}} = 203, \) the size of HMATRIX exceeds 2 GB assuming quadruple precision of 16 B.

5 PROGRAM MANUAL

5.1 Installation

First, expand the downloaded file. Here we take its file name as scelto_gauss.tar.gz [31].

\texttt{tar -zxvf scelto_gauss.tar.gz}

Compile the source programs in the directory src using the make command.

\texttt{cd src}
\texttt{make}
\texttt{cd ..}

An executable program named \texttt{sscelto_gauss.exe} will be generated. Move it into the directory bin.

\texttt{mv src/sscelto_gauss.exe bin/}

Run the program.

\texttt{cd wshop.N19}
\texttt{../bin/sscelto_gauss.exe}

The Intel Fortran compiler is the default. For the Fujitsu compiler, \texttt{Makefile.Fujitsu} is available. To use other compilers, the command name and compile options must be changed. In particular, some compile options may have to be changed; the large array (HMATRIX) should therefore be placed in a memory region beyond 2 GB.
The value of the parameter constant NQMAX, which defines the upper limit for Nmax, is set to 211. By compiling the source programs with smaller NQMAX, the program size can be reduced.

Test input data c5nqg.inp is stored in the directory wshop.N19. This data is for the case Nmax = 19. It gives EE of $-1.10227888655218872575$ a.u.

5.2 Input data
We describe the input data parameters, taking the input file in the directory wshop.N19 as an example.

Line:Variable or array / Meaning
1: comment
2: Nmax / maximum of $n$.
3: comment
4: Lmax / maximum of $l$.
5: comment
6: R / nuclear position (a.u.) from the midpoint.
7: comment
8: ZNUC / nuclear charge. (1.0 for H$_2^+$)
9: comment
10–19: ZL / exponents for even $l$ numbers.

5.3 Output files
The following files will be generated after execution of the program.

c5nqg.outlist
c5nqg.log
c5nqg.out9.txt
c5nqg.out.txt

For each value of $l$, the Hamiltonian matrix is diagonalized and the exponents, eigenvalues, and eigenvectors are written to the file c5nqg.out.txt. Immediately after the diagonalization for a particular $l$ is finished, the number $l$ is output to the file c5nqg.
7

Table 1. Electronic energy of hydrogen molecular ion at \(R_{\text{bond}}=2.0\) a.u.

| Method   | Nmax | Lmax | Ndim | Exponent \((z_l)\) | EE (a.u.) | CPU-time (sec)* |
|----------|------|------|------|-------------------|----------|-----------------|
| Scaling  | 51   | 50   | 676  | 14.59721          | −1.10258505 | 271             |
| Scaling  | 71   | 70   | 1296 | 19.26503          | −1.10261435 | 2235            |
| Scaling  | 101  | 100  | 2601 | 25.97300          | −1.10262669 | 22493           |
| Scaling  | 131  | 130  | 4356 | 32.84360          | −1.10263056 | 126906          |
| Scaling  | 203  | 202  | 10404| 51.18760          | −1.10263287 | 2521290         |
| Full opt.† | 131 | 130  | 4356 | 1.10263323        | 1.10263395 | 2521290         |
| Emp. opt.‡ | 203 | 202  | 10404| 1.10263395        | 1.10263421 |                 |
| Exact (by Peek [14]) | 131 | 130  | 4356 | 1.10263323        | 1.10263395 | 2521290         |

* CPU-time on an Intel Core i7-3970X (3.5 GHz) in serial mode.
† The exponents are fully optimized.
‡ The exponents are optimized empirically. See text.

6 PERFORMANCE

Table 1 shows the computational results. The fourth column, named Ndim, gives the dimensions of the Hamiltonian matrices. For the method titled ‘Scaling’, identical exponents were used for all the \(l\) numbers. For Nmax = 203, the scaling method yields the energy as −1.10263287 a.u, which differs from the exact energy by 1.34 \(\times 10^{-6}\) a.u. The exponents were determined by extrapolation from the preceding runs of \(n=1–70\).

\[
z_l = 1.6193375636179528 + 0.2782894519200024n - 0.0005804140793906168n^2 + 0.000002103469132713343n^3
\] (35)

By optimizing the exponents for \(l\) individually, EE has been decreased further to −1.10263395 a.u. The difference between the final energy for Nmax = 203 and the exact value is less than 2.6 \(\times 10^{-7}\) a.u.

For Nmax = 203, the exponents were optimized in an empirical manner. First, the exponent for \(l = 0\) (denoted as \(z_0\)) is optimized, with Lmax set to 0. Next, the exponent for \(l = 2\) (\(z_2\)) is optimized from an initial value given by multiplying \(z_0\) by 1.1, with Lmax set to 2. By repeating this procedure the exponents are determined, until \(l = 50\). The exponent for \(l = 202\) (\(z_{202}\)) is fixed to 203.5 by \(n+0.5\). The exponents for \(l = 52–200\) are calculated from a line connecting \(z_{50}\) and \(z_{202}\) (The optimization routines are not provided.)

The electronic energies are plotted versus Nmax in Figure 4. The blue curve marked with squares shows the energy obtained by the scaling method. The red curve marked with circles shows the energy obtained with optimized exponents. Solid circles indicate that the exponents are fully optimized. The open circles for Nmax = 161–203 signify that the exponents are optimized empirically. The calculated energies approach the exact value.

NOTE ADDED IN PROOF

To observe the \(l\)-dependency of the eigenvalue, the Hamiltonian was diagonalized for each \(l\)-block \((l=0, 1, ..., L_{\text{max}})\). The CPU-time in Table 1 includes all the diagonalization steps for \(l\). If the Hamiltonian is diagonalized only once for the maximum \(l(L_{\text{max}})\), the CPU time is drastically reduced. Moreover, we have found that the Hamiltonian can be diagonalized in the double-precision without loss of accuracy. By considering these 2 factors, we have reduced the CPU-time for Nmax = 203 to 681190 sec (about 1/4 of the original). Finally we succeeded to extend \(n\) to 241 and obtained EE = −1.10263406 a.u.

For the case of Nmax = 39 and Lmax = 38, EE was calculated to be −1.10253135 a.u. We have performed additional calculations by locating the expansion center at one nucleus and have obtained EE = −1.10244106 a.u. We call this computational scheme one-side expansion against midpoint expansion described in text. The midpoint expansion gives slightly better EE. In the one-side expansion, we have also calculated the nuclear cusp condition to
be $-2.34$. Since in the midpoint expansion, the peak position differs from the nuclear position, the calculated cusp condition loses its exact physical meaning. However, we estimated its value to be $-0.33$ by numerical differentiation along the $Z$ axis. Along the $y$ axis (perpendicular to the molecular axis), it was $-1.0 \times 10^{-5}$. In all these cases, the calculated values deviate considerably from the true value ($-1$). Although the LTOs themselves are considered to be superior to the GTOs for computing the cusp conditions, their single-center expansion gives unsatisfactory results.

**ACKNOWLEDGEMENTS**

We are grateful to Professor Emeritus Hiroshi Tatewaki of Nagoya City University for valuable suggestions. Y.H. thanks Professor Emeritus Takemitsu Hasegawa at Fukui University for their helpful suggestions regarding Gaussian quadratures. The main computation was performed using a cluster of Intel Xeon-based computers located at the Institute for Advanced Studies in Artificial Intelligence (IASAI) of Chukyo University. The Fujitsu PRIMEHPC FX100 supercomputer at the Information Technology Center of Nagoya University was used for portability testing.

**REFERENCES**

[1] T. Helgaker, P. Jorgensen, J. Olsen, *Molecular Electronic-Structure Theory*, John Wiley & Sons, Chichester, 2000.
[2] M. W. J. Bromley, J. Mitroy, *Int. J. Quantum Chem.*, **107**, 1150 (2007). [CrossRef]
[3] L. G. Jiao, Y. K. Ho, *Int. J. Quantum Chem.*, **115**, 434 (2015). [CrossRef]
[4] M. T. Marron, N. C. Handy, R. G. Parr, H. J. Silverstone, *Int. J. Quantum Chem.*, **4**, 245 (1970). [CrossRef]
[5] J. Hachmann, P. T. A. Galek, Y. Yanai, G. K.-L. Chan, N. C. Handy, *Chem. Phys. Lett.*, **392**, 55 (2004). [CrossRef]
[6] N. Takeda, Y. Hatano, S. Yamamoto, H. Tatewaki, *Comput. Theor. Chem.*, **1045**, 99 (2014). [CrossRef]
[7] R. Courant, D. Hilbert, *Methods of Mathematical Physics*, Vol. I, Interscience, New York, 1953.
[8] G. Szegö, *Orthogonal Polynomials*, American Mathematical Society, Providence, Rhode Island, 1939.
[9] K. M. Howell, H. Shull, *J. Chem. Phys.*, **30**, 627 (1959). [CrossRef]
[10] H. H. Kranz, E. O. Steinborn, *Phys. Rev. A*, **25**, 66 (1982). [CrossRef]
[11] S. Hagstrom, H. Shull, *J. Chem. Phys.*, **30**, 1314 (1959). [CrossRef]
[12] Ø. Burrau, *Det. Kgl. Danske Vid. Selskab.*, **7**, 1 (1927).
[13] E. A. Hylleraas, *Z. Phys.*, **71**, 739 (1931). [CrossRef]
[14] J. M. Peck, *J. Chem. Phys.*, **43**, 3004 (1965). [CrossRef]
[15] R. M. Fernández, *J. Chem. Phys.*, **103**, 6581 (1995). [CrossRef]
[16] D. R. Bates, K. Ledsham, A. L. Stewart, *Phil. Trans. Roy. Soc. London A.*, **246**, 215 (1953). [CrossRef]
[17] T. Kato, *Commun. Pure Appl. Math.*, **10**, 151 (1957). [CrossRef]
[18] B. H. Wells, S. Wilson, *J. Phys. At. Mol. Opt. Phys.*, **22**, 1285 (1989). [CrossRef]
[19] D. H. Bailey, *ACM Trans. Math. Softw.*, **21**, 379 (1995). [CrossRef]
[20] D. M. Smith, *J. Comput. Sci. Eng.*, **392**, 245 (1970). [CrossRef]
[21] F. B. Hildebrand, *Introduction to Numerical Analysis*, 2nd ed., Dover, New York, 1987.
[22] G. H. Golub, J. H. Welsch, *Math. Comput.*, **23**, 221 (1969). [CrossRef]
[23] P. Davis, P. Rabinowitz, *J. Res. Natl. Bur. Stand.*, **56**, 35 (1956). [CrossRef]
[24] IEEE Computer Society, (August 29, 2008), IEEE Standard for Floating-Point Arithmetic, IEEE, doi: 10.1109/IEEESTD.2008.4610935.
[25] J. A. Kempe, S. P. Goldman, *J. Chem. Phys.*, **108**, 7679 (1998). [CrossRef]
[26] A. Ishikawa, H. Nakashima, H. Nakatsuji, *J. Chem. Phys.*, **128**, 124103 (2008). [CrossRef]
[27] Y. Ning, Z.-C. Yan, *Phys. Rev. A*, **90**, 032516 (2014). [CrossRef]
[28] K. Matsubara, Y. Horino, T. Murai, *Bull. Nara Univ. Educ.*, **17**, 13 (1969).
[29] I. Ninomiya, Y. Hatano, “Performance of the Vector Version of NUMPAC”, in: *High Performance Computing: Research and Practice in Japan*, R. Mendez (Ed.), John Wiley & Sons, Chichester, 1992. The source programs and the manual can be downloaded from http://www2.itc.nagoya-u.ac.jp/center/ja/numpac/index.html.
[30] T. Hasegawa, T. Sandoh, Y. Sato, Y. Hatano, I. Ninomiya, *Memoirs of the Faculty of Engineering, Fukui Univ.*, **48**, 193 (2000).
[31] scelto_gauss.tar.gz