Chemical Physics Letters 479 (2009) 160–164

Contents lists available at ScienceDirect

Chemical Physics Letters

journal homepage: www.elsevier.com/locate/cplett

Evaluation of Slater-type geminal integrals using tailored Gaussian quadrature

Toru Shiozaki *

Quantum Theory Project, Department of Chemistry, University of Florida, Gainesville, FL 32611-8435, USA
Department of Applied Chemistry, Graduate School of Engineering, The University of Tokyo, Tokyo 113-8656, Japan

ARTICLE INFO

Article history:
Received 2 July 2009
In final form 31 July 2009
Available online 3 August 2009

ABSTRACT

A simple, yet effective algorithm is developed for evaluating two-electron Slater-type geminal and Yukawa potential integrals over Gaussian-type orbitals (GTOs), which arise in the so-called explicitly correlated methods, on the basis of the recent work of Ten-no [S. Ten-no, Chem. Phys. Lett. 398 (2004) 56; S. Ten-no, J. Chem. Phys. 126 (2007) 014108]. Gaussian quadrature is used in analogy with the Rys quadrature method for electron repulsion integrals. The quadrature grids are obtained by the two-dimensional Chebyshev interpolation. This algorithm is especially efficient for integrals over GTOs with high angular momenta, which are present owing to the use of the resolution-of-the-identity approximation.

1. Introduction

The explicitly correlated methods that have originated from the work of Kutzelnigg [1], referred to as the R12 (or F12) methods, have proven highly effective in providing near complete basis-set limits of electron correlation energies using a small basis set and a Slater-type geminal (STG) [2,3]. See recent publications [4–12] and references therein. In the so-called approximation C of Kedžuch et al. [13], the R12 (or F12) methods with a STG involve two types of two-electron integrals over Gaussian-type orbitals (GTOs) [5]:

$$(\phi_p\phi_q|e^{-\xi r_{12}}|\phi_r\phi_s), \tag{1}$$

$$\left(\phi_p\phi_q\left|\frac{e^{-\xi r_{12}}}{r_{12}}\right|\phi_r\phi_s\right), \tag{2}$$

where ξ is either the exponent of the STG or twice the exponent. The chemists' notation is used (i.e., ϕ_p and ϕ_q are GTOs of electron 1 and ϕ_r and ϕ_s electron 2, respectively). Since the R12 (or F12) methods use auxiliary basis functions with high angular momenta for the resolution-of-the-identity approximation (RI) [14,15], efficient implementation of Eqs. (1) and (2) for GTOs with high angular momenta is desirable, especially for low-level electron correlation treatments such as the explicitly correlated Møller–Plesset perturbation method (MP2-R12 or -F12) [16].

One way to evaluate these integrals is to approximate a STG by a linear combination of Gaussian-type geminals (GTGs) [17–19]. Although efficient algorithms based on this scheme exist [19,20], they can be inherently not optimal. Ten-no proposed an alternative that evaluates STG and related integrals without using the GTG expansion [2,3]. In his pioneering work, he formulated them in terms of the special function called $G_m(T,U)$, in analogy with the Boys function $F_m(T)$ in the ERI evaluation [21]. This corresponds to the generalization of a family of ERI evaluation algorithms such as McMurchie–Davidson [22], Pople–Hehre [23], Obara–Saika [24], Head-Gordon–Pople [25], PRISM of Gill et al. [26], and others [27–30] to the STG integrals.

In this Letter, we present an algorithm for STG and Yukawa potential (YP) integrals [Eqs. (1) and (2)] that uses Gaussian quadrature tailored specifically for these integrals on the basis of Ten-no's recent work [2,3]. Our approach is analogous to the so-called Rys quadrature method of Dupuis et al. [31–33] for ERI, later elaborated by Lindh et al. [34], which is known to be advantageous for integrals over GTOs with high angular momenta among other algorithms. Note that Ten-no has also studied a similar algorithm that uses twice the number of grid points as that of the Gaussian quadrature formula [35,36]. The floating-point operation counts and the timings are presented to show that the algorithm is efficient especially for integrals over GTOs with high angular momenta.

2. Slater-type geminal and Yukawa potential integrals

The Cartesian GTOs are defined as

$$\phi_p = (x_1 - A_x)^{a_x}(y_1 - A_y)^{a_y}(z_1 - A_z)^{a_z}e^{-\zeta_a(\mathbf{r}_1-\mathbf{A})^2}, \tag{3}$$

$$\phi_q = (x_1 - B_x)^{b_x}(y_1 - B_y)^{b_y}(z_1 - B_z)^{b_z}e^{-\zeta_b(\mathbf{r}_1-\mathbf{B})^2}. \tag{4}$$

Similarly, ϕ_r and ϕ_s are written with **c**, **d**, ζ_c, ζ_d, **C**, and **D**. Using the integral representation of Ten-no [2] (see also Appendix A),

$$\frac{e^{-\xi r_{12}}}{r_{12}} = \frac{2}{\sqrt{\pi}}\int_0^\infty e^{-r_{12}^2u^2-\frac{\xi^2}{4u^2}}du, \tag{5}$$

the YP integral over Cartesian GTOs has the form:

* Address: Quantum Theory Project, Department of Chemistry, University of Florida, Gainesville, FL 32611-8435, USA. Fax: +1 (352)392 8722.
E-mail address: shiozaki@qtp.ufl.edu

0009-2614/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.cplett.2009.07.108

$$\left(\phi_p\phi_q\left|\frac{e^{-\xi r_{12}}}{r_{12}}\right|\phi_r\phi_s\right) = \frac{2E_{AB}E_{CD}}{\sqrt{\pi}}\int_0^\infty I_x^*(a_x,b_x,c_x,d_x,u) \times I_y^*(a_y,b_y,c_y,d_y,u)I_z^*(a_z,b_z,c_z,d_z,u)e^{-\frac{\xi^2}{4u^2}}du, \tag{6}$$

where $E_{AB} = e^{-\zeta_a\zeta_b|\mathbf{A}-\mathbf{B}|^2/p}$, $E_{CD} = e^{-\zeta_c\zeta_d|\mathbf{C}-\mathbf{D}|^2/q}$, and

$$I_x^*(a_x,b_x,c_x,d_x,u) = \iint (x_1-A_x)^{a_x}(x_1-B_x)^{b_x}(x_2-C_x)^{c_x}(x_2-D_x)^{d_x} \times e^{-p(x_1-P_x)^2-q(x_2-Q_x)^2-u^2(x_1-x_2)^2}dx_1dx_2. \tag{7}$$

The following quantities are used: $p = \zeta_a+\zeta_b$, $q = \zeta_c+\zeta_d$, $\mathbf{P} = (\zeta_a\mathbf{A}+\zeta_b\mathbf{B})/p$, and $\mathbf{Q} = (\zeta_c\mathbf{C}+\zeta_d\mathbf{D})/q$. Note that the definition of $I^*(a,b,c,d,u)$ is identical to that in ERI. Changing the integration variable in Eq. (6) with

$$t = \frac{u^2}{u^2+\rho}, \tag{8}$$

we arrive at a simpler expression for the YP integral,

$$\left(\phi_p\phi_q\left|\frac{e^{-\xi r_{12}}}{r_{12}}\right|\phi_r\phi_s\right) = \frac{2\pi^2\sqrt{\pi}}{pq\sqrt{p+q}}E_{AB}E_{CD}\int_0^1 tI_x(a_x,b_x,c_x,d_x,t) \times I_y(a_y,b_y,c_y,d_y,t)I_z(a_z,b_z,c_z,d_z,t)w(t)dt, \tag{9}$$

in which $T = \rho|\mathbf{P}-\mathbf{Q}|^2$ and $U = \xi^2/4\rho$ with $\rho = pq/(p+q)$, and

$$w(t) = \frac{e^{-Tt+U(1-t^{-1})}}{2t\sqrt{t}}. \tag{10}$$

The scaled two-dimensional integrals [such as $I_x(a_x,b_x,c_x,d_x,t)$] in Eq. (9) are defined as

$$I_x(a_x,b_x,c_x,d_x,t) = I_x^*(a_x,b_x,c_x,d_x,t)/I_x^*(0,0,0,0,t). \tag{11}$$

The corresponding expression for the STG integral is derived by applying the relation,

$$e^{-\xi r_{12}} = -\frac{\partial}{\partial\xi}\frac{e^{-\xi r_{12}}}{r_{12}}, \tag{12}$$

to the YP integral [2]:

$$(\phi_p\phi_q|e^{-\xi r_{12}}|\phi_r\phi_s) = \frac{4U\pi^2\sqrt{\pi}}{\xi pq\sqrt{p+q}}E_{AB}E_{CD}\int_0^1(1-t)I_x(a_x,b_x,c_x,d_x,t) \times I_y(a_y,b_y,c_y,d_y,t)I_z(a_z,b_z,c_z,d_z,t)w(t)dt. \tag{13}$$

Since $I(a,b,c,d,t)$ is a polynomial of t (see, for instance, Ref. [34]), the one-dimensional integrals in Eqs. (9) and (13) can be evaluated by the Gaussian quadrature formula,

$$\int_0^1 P(t)w(t)dt = \sum_{g=1}^{\lceil(n+1)/2\rceil} P(t_g)W_g, \tag{14}$$

where t_g and W_g are the position and weight of the grid point, respectively, and n is the degree of a polynomial $P(t)$.

3. Integral evaluation using tailored Gaussian quadrature

To use the Gaussian quadrature formula, we must calculate the positions and weights of the grid points. In the ERI evaluation (i.e., in the Rys quadrature method [31–34]), the weight function has only one parameter associated with the exponents of the GTOs; therefore, they are typically precomputed and stored for representative values of the parameter in the implementation step and interpolated for other values at run-time [31,37–39]. Recently, an elegant approach that evaluates them on the fly by the so-called modified Chebyshev algorithm [40] has been proposed by Flocke et al. [41,42]. The weight function for STG and YP integrals, by contrast, has two parameters (T and U), which makes the calculation of quadrature grids less straightforward. In the current work, we calculate the quadrature grids by interpolation based on two-dimensional tables.

3.1. Preparing two-dimensional tables for interpolation

The two-dimensional tables for interpolating the positions and weights of the grid points are generated at the time of implementation as follows. Given the weight function $[w(t)]$ and the integral range $[0,1]$, there exists a set of monic orthogonal polynomials which satisfy

$$\int_0^1 P_m(t)P_n(t)w(t)dt = 0 \quad \text{if } m \neq n. \tag{15}$$

These polynomials have a three-term recurrence relation because of their mutual orthogonality,

$$P_m(t) = (t-\alpha_{m-1})P_{m-1}(t) - \beta_{m-1}P_{m-2}(t), \tag{16}$$

where

$$\alpha_m = \frac{\int_0^1 tP_m(t)^2w(t)dt}{\int_0^1 P_m(t)^2w(t)dt}, \tag{17}$$

$$\beta_m = \frac{\int_0^1 P_m(t)^2w(t)dt}{\int_0^1 P_{m-1}(t)^2w(t)dt}. \tag{18}$$

The α's and β's are related to the quadrature grid. The Gaussian quadrature of rank n_{quad}, which is capable of integrating polynomials of degrees up to $2n_{\text{quad}}-1$ exactly under the given $w(t)$ and integral range, exploits the roots of $P_{n_{\text{quad}}}(t)$ as the positions of the grid points. Since the integrands of Eqs. (9) and (13) divided by $w(t)$ are polynomials of degree $L_{\text{total}}+1$ with $L_{\text{total}} = \sum_{w=x,y,z}(a_w+b_w+c_w+d_w)$ [note the factors of t and $(1-t)$ in Eqs. (9) and (13)], we choose

$$n_{\text{quad}} = \lceil L_{\text{total}}/2\rceil + 1 \tag{19}$$

as the rank of the quadrature. For integrals over four f-type GTOs, for instance, $n_{\text{quad}} = 7$ is chosen.

While it is formally possible to use the Chebyshev algorithm [43] for calculating α's and β's from the zeroth- through $(2n_{\text{quad}}-1)$th-order moments introduced by Ten-no [2],

$$G_m(T,U) = \int_0^1 t^{m+1}w(t)dt, \quad -1 \leqslant m \leqslant 2n_{\text{quad}}-2, \tag{20}$$

the mapping from moments to α's and β's is known to be ill-conditioned and hence numerically unstable [40]. In order to circumvent this problem, we resort to multiple precision arithmetic [44]. The moments are calculated by the recurrence formula of Ten-no [2],

$$G_m = \frac{1}{2T}\left[(2m-1)G_{m-1} + 2UG_{m-2} - e^{-T}\right], \tag{21}$$

$$G_{-1}(T,U) = \frac{e^{-T}}{4}\sqrt{\frac{\pi}{U}}\left[e^{\kappa^2}\operatorname{erfc}(\kappa) + e^{\lambda^2}\operatorname{erfc}(\lambda)\right], \tag{22}$$

$$G_0(T,U) = \frac{e^{-T}}{4}\sqrt{\frac{\pi}{T}}\left[e^{\kappa^2}\operatorname{erfc}(\kappa) - e^{\lambda^2}\operatorname{erfc}(\lambda)\right], \tag{23}$$

with $\kappa = -\sqrt{T}+\sqrt{U}$ and $\lambda = \sqrt{T}+\sqrt{U}$. Subsequently, α's and β's are computed by the Chebyshev algorithm [43,40]:

$$\sigma_{k,l} = \sigma_{k-1,l+1} - \alpha_{k-1}\sigma_{k-1,l} - \beta_{k-1}\sigma_{k-2,l}, \quad l = k,k+1,\ldots,2n_{\text{quad}}-k-1, \tag{24}$$

$$\alpha_k = (\sigma_{k,k+1}/\sigma_{k,k}) - (\sigma_{k-1,k}/\sigma_{k-1,k-1}), \tag{25}$$

$$\beta_k = \sigma_{k,k}/\sigma_{k-1,k-1}, \tag{26}$$

with

$$\alpha_0 = G_0(T,U)/G_{-1}(T,U), \tag{27}$$
$$\beta_0 = G_{-1}(T,U), \tag{28}$$
$$\begin{aligned}&\sigma_{-1,l} = 0, \quad l = 1,2,\ldots,2n_{\text{quad}}-2,\\ &\sigma_{0,l} = G_{l-1}(T,U), \quad l = 0,1,\ldots,2n_{\text{quad}}-1.\end{aligned} \tag{29}$$

The position of the grid point is obtained as the eigenvalue of the tridiagonal matrix [40],

$$\begin{pmatrix} \alpha_0 & \sqrt{\beta_1} & & & \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & \\ & \ddots & \ddots & \ddots & \\ & & \sqrt{\beta_{n_{\text{quad}}-2}} & \alpha_{n_{\text{quad}}-2} & \sqrt{\beta_{n_{\text{quad}}-1}} \\ & & & \sqrt{\beta_{n_{\text{quad}}-1}} & \alpha_{n_{\text{quad}}-1} \end{pmatrix}, \tag{30}$$

and the weight is calculated using the first element of the associated eigenvector ($x_{j,0}$) as

$$W_j = x_{j,0}^2 G_{-1}(T,U). \tag{31}$$

The positions of the grid points as functions of T and U are depicted in Fig. 1 for the case of $n_{\text{quad}} = 2$. They move along with the pole of $w(t)$ located around $\sqrt{U/T}$.

Finally, we store the Chebyshev expansion coefficients [43] of the positions and weights of the grid points using 100 boxes in the ranges of $T \in [0, 2^{10}]$ and $U \in [10^{-7}, 10^3]$—$U \in [10^k, 10^{k+1}]$ $(k = -7, -6, \ldots, 2)$ and $T \in [0,2]$ and $[2^k, 2^{k+1}]$ $(k = 1,2,\ldots,9)$—inside which T and U are mapped onto $\bar{T}$, $\bar{U} \in [-1,1]$. Note that the weights are scaled by $\sqrt{U}$ to avoid singular behavior at $U \to 0$. The asymptotic forms are used for $T > 2^{10}$.

3.2. Two-dimensional interpolation

At run-time, the positions and weights of the grid points of given T_{target} and U_{target} are obtained by the successive application of the Chebyshev interpolation (which should not be confused with the Chebyshev algorithm described above). The computational procedure starts with one-dimensional interpolation of rank 14 to obtain $\{f(\bar{T}_k, \bar{U}_{\text{target}})\}$ $(0 \leqslant k \leqslant 13)$, where f is a function of $\bar{T}$ and $\bar{U}$ for either the position or the (scaled) weight of the grid point. Clenshaw's recurrence formula [43] is used:

$$d_{k,15} = d_{k,14} = 0, \tag{32}$$
$$\begin{aligned}&d_{k,j} = 2\bar{U}_{\text{target}} d_{k,j+1} - d_{k,j+2} + c_{k,j},\\ &\quad (j = 13, 12, \ldots, 1),\end{aligned} \tag{33}$$
$$f(\bar{T}_k, \bar{U}_{\text{target}}) = \bar{U}_{\text{target}} d_{k,1} - d_{k,2} + \frac{1}{2} c_{k,0}, \tag{34}$$

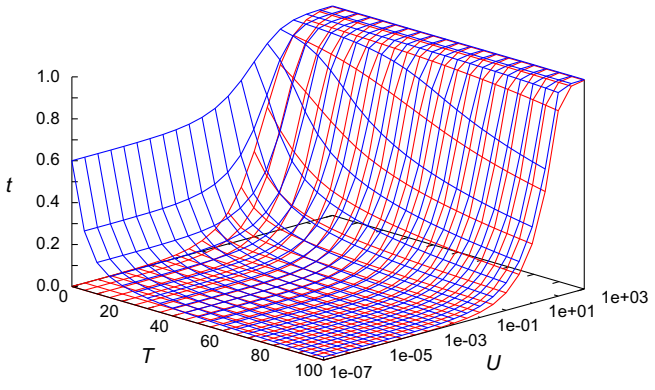

Fig. 1. The positions of the grid points for $n_{\text{quad}} = 2$ as functions of T and U. The weight function of the quadrature is $e^{-Tt+U(1-t^{-1})}/2t\sqrt{t}$.

in which all the Chebyshev expansion coefficients ($c_{k,j}$) are known *a priori*. Successively, we compute the Chebyshev expansion coefficients of these 14 points ($\bar{c}_j$) as

$$\bar{c}_j = \frac{1}{7}\sum_{k=0}^{13} f(\bar{T}_k, \bar{U}_{\text{target}}) \cos\left(\frac{\pi j(2k+1)}{28}\right), \quad (j = 0,1,\ldots,13), \tag{35}$$

from which $f(\bar{T}_{\text{target}}, \bar{U}_{\text{target}})$ is calculated using Clenshaw's formula. Note that cosines are to be precomputed and stored in computer codes at the time of implementation.

The computational cost of the interpolation procedure is dominated by the contributions from Eqs. (33)–(35). The number of total floating-point operations in the interpolation procedure is, therefore, roughly $10 \times 2n_{\text{quad}} \times 14^2 \times n_{\text{prim}}^4 \sim 4000 n_{\text{quad}} n_{\text{prim}}^4$. Though the prefactor is large, it scales linearly with respect to n_{quad} or, equivalently, to L_{total}, which is in contrast to the quadratic scaling of the operation counts of the Rys quadrature method. For a batch of integrals over four f-type GTOs ($n_{\text{quad}} = 7$), the operation count for evaluating the quadrature grid is about $28\,000 n_{\text{prim}}^4$, which is 3–10 times smaller than the $77\,000 n_{\text{prim}}^4 + 135\,000 n_{\text{cont}}^4$ operation counts [34] of the Rys quadrature scheme (n_{prim} and n_{cont} are the numbers of primitive and contracted GTO quartets, respectively). In addition, the interpolation procedure does not require any loops of variable lengths, and therefore, can be effectively implemented. When L_{total} is small, the interpolation procedure can nonetheless be a large overhead.

The accuracy of the interpolation procedure is shown in Table 1 for $n_{\text{quad}} = 7$ by comparing the computed moments [i.e., $G_m(T,U)$ $(-1 \leqslant m \leqslant 12)$] and their exact values. The largest error is found to be below 10^{-11}, which is sufficient since STG and YP integrals account for the small basis-set truncation errors of correlation energies in the R12 (or F12) methods. One might instead pursue higher accuracy by increasing the order of Chebyshev expansions or the number of boxes.

3.3. Recurrence relations

The partial differentiation of Eq. (7) with respect to x_1 and x_2 and the rearrangement of terms lead to the so-called vertical and horizontal recurrence relations (denoted as VRR and HRR) for $I(a_x, b_x, c_x, d_x, t)$ [34]. Since the definition of $I(a_x, b_x, c_x, d_x, t)$ is identical to that in the ERI evaluation as mentioned above, both VRR

Table 1
The maximum deviations from the reference values of $G_m(T,U)$ $(-1 \leqslant m \leqslant 12)$ computed by Gaussian quadrature ($n_{\text{quad}} = 7$). The reference values are calculated by multiple precision arithmetic. The maximum deviation is usually seen at $m = -1$.

T	U	$G_{-1}(T,U)$	$\max\{\Delta G_m(T,U)\}$
0.125	0.002	1.9E+01	4.7E−12
0.125	0.2	1.2E+00	4.5E−12
0.125	20	2.2E−02	1.0E−12
0.5	0.002	1.8E+01	9.0E−12
0.5	0.2	1.1E+00	8.6E−12
0.5	20	1.5E−02	1.9E−12
2.5	0.002	1.7E+01	7.6E−15
2.5	0.2	5.8E−01	3.8E−13
2.5	20	2.3E−03	3.8E−15
10	0.002	1.5E+01	1.5E−14
10	0.2	1.4E−01	2.3E−12
10	20	1.9E−06	2.7E−15
40	0.002	1.1E+01	3.6E−13
40	0.2	8.5E−03	3.6E−12
40	20	2.6E−17	1.3E−18
160	0.002	6.4E+00	3.7E−12
160	0.2	3.0E−05	5.7E−12
160	20	–	–[a]

[a] All the moments are negligibly small.

and HRR formulas remain the same as those in the ERI evaluation. The explicit formulas have been given elsewhere [34,41].

4. Implementation and timings

The computer code that implements the aforementioned integral evaluation method has been a part of an in-house electronic structure program for molecules and solids [45]. Most of the codes for STG and/or YP integrals (as well as those for ERI) have been generated by simple code generators, while the subroutines accounting for integrals with small L_{total} have been optimized by hand. The quadrature grid evaluation and the VRR are common for STG and YP integrals when they are computed simultaneously, whereas the assembly step [i.e., numerical integration of Eqs. (9) and (13) using the Gaussian quadrature formula], basis-set contractions, Cartesian-to-spherical transformations, and the HRR have been performed separately.

The YP integrals have been verified by comparison with the values computed with the finite-difference method developed by us [46] and modified to solve the screened Poisson equation [see also Eq. (39)],

$$\left(\phi_p\phi_q\left|\frac{e^{-\xi r_{12}}}{r_{12}}\right|\phi_r\phi_s\right) = \int V_Y(\mathbf{r})\phi_p(\mathbf{r})\phi_q(\mathbf{r})d\mathbf{r}, \tag{36}$$

$$\left(\nabla^2 - \xi^2\right)V_Y(\mathbf{r}) = -4\pi\phi_r(\mathbf{r})\phi_s(\mathbf{r}). \tag{37}$$

This is in analogy with the computation of ERI using the Poisson equation [47]. The STG integrals have been checked by numerical derivatives of the YP integrals with respect to ξ.

The timings in CPU seconds for evaluating the symmetry distinct [48] ERI, STG, and YP integrals of D_{2h} C_2H_4 without screening are listed in Table 2. The subsets of the aug-cc-pVQZ basis set [49] are used. All the timings are measured on a single core of Core 2 Duo 2.0 GHz CPU (i.e., the other core is idle). The difference between the computational cost for STG integrals and that for ERI stems from two reasons: (i) the quadrature grid evaluation is more expensive for STG integrals than for ERI; and (ii) the rank of Gaussian quadrature for STG integrals is higher by one than that for ERI when L_{total} is odd, owing to the factor of t and $(1-t)$ in Eqs. (9) and (13).

For the subset consisting of d, f, and g shells, the algorithm presented here works remarkably well, resulting in a similar CPU timing to (only 1.08 times larger than) that of ERI as expected from the floating-point operation counts. When integrals with small L_{total} are included, the ratio deteriorates (1.61 for the entire aug-cc-PVQZ basis set), since the two-dimensional interpolation procedure is more expensive than the other operations for small L_{total}. The simultaneous calculation of STG and YP integrals halves this overhead, taking advantage of the fact that they can be computed with the same quadrature grid. The computational timing (divided by 2) is 1.28 times as great as that for ERI using the aug-cc-pVQZ basis set when STG and YP integrals are computed simultaneously.

Table 2
Timings (in CPU seconds) for evaluating STG and/or YP integrals as well as ERI. The subsets of the aug-cc-pVQZ basis set [49] are used for symmetry distinct integrals of D_{2h} C_2H_4.

Shells	STG	STG + YP	ERI[a]	STG/ERI	STG + YP/ERI
fg	4.12	8.34	3.98	1.04	1.05
dfg	21.6	44.5	20.1	1.08	1.11
pdfg	59.3	112.3	48.8	1.22	1.15
spdfg	133.3	213.1	82.7	1.61	1.28

[a] Evaluated by the Rys quadrature method. The quadrature grid is obtained by the one-dimensional interpolation.

5. Conclusions

We have presented an efficient way of computing STG and YP integrals using tailored Gaussian quadrature. The positions and weights of the grid points have been evaluated by the two-dimensional interpolation. Both the theoretical operation counts and the actual timings have proven that, for integrals with large L_{total}, the computational cost of the current scheme is similar to that of the Rys quadrature method for ERI. Since one needs to include basis functions with high angular momenta in the R12 (or F12) methods owing to the introduction of auxiliary basis functions for the RI, this algorithm is useful for the efficient implementation of the R12 (or F12) methods. It is worth noting that this algorithm can be used for integrals over GTOs with arbitrarily high angular momenta.

The application of this algorithm to integrals with small L_{total} has been, however, less effective, since two-dimensional interpolation procedures constitute a large overhead, though the overhead can be halved by simultaneous calculations of STG and YP integrals. We expect that, to attain the best performance, the other algorithms such as the one proposed by Ten-no [2,3] must be combined for integrals with small L_{total} (say, for $L_{\text{total}} \leqslant 5$). Together with the present scheme, these STG and YP integral evaluation methods serve to increase the performance of the R12 (or F12) methods. The on-the-fly evaluation of quadrature grids, proposed for ERI by Flocke et al. [41,42] and for STG and YP integrals by Ten-no [36], can further increase the efficiency of the current algorithm even for small L_{total}.

Acknowledgements

Professor So Hirata is greatly thanked for generous support. The author acknowledges Professor Seiichiro Ten-no for helpful discussions and comments on the manuscript. He is also grateful to Dr. Norbert Flocke and Professors James D. Talman and Edward F. Valeev for useful discussions. The author is supported by the Japan Society for the Promotion of Science Research Fellowship for Young Scientist and by US Department of Energy (Grant No. DE-FG02-04ER15621).

Appendix A. Integral representation of the Yukawa potential

The Fourier transform of the YP is

$$\frac{e^{-\xi r_{12}}}{r_{12}} = \frac{1}{2\pi^2}\int \frac{e^{i\mathbf{k}\cdot\mathbf{r}_{12}}}{k^2+\xi^2}d\mathbf{k}, \tag{38}$$

which implies that it is Green's function of the screened Poisson equation, i.e.,

$$\left(\nabla_1^2 - \xi^2\right)\frac{e^{-\xi r_{12}}}{r_{12}} = -4\pi\delta(\mathbf{r}_{12}), \tag{39}$$

in analogy with the Coulomb potential which is of the usual Poisson equation. Using the following Laplace transformation,

$$\frac{1}{s} = \int_0^\infty e^{-su'}\,du', \tag{40}$$

Eq. (38) becomes

$$\frac{1}{2\pi^2}\int_0^\infty e^{-\xi^2 u'}\int e^{-k^2u'+i\mathbf{k}\cdot\mathbf{r}_{12}}\,d\mathbf{k}\,du' = \frac{1}{2\sqrt{\pi}}\int_0^\infty \frac{1}{u'\sqrt{u'}}e^{-\xi^2u'-\frac{r_{12}^2}{4u'}}\,du'. \tag{41}$$

We arrive at Eq. (5) by changing the integration variable to $u = 1/(2\sqrt{u'})$.

References

[1] W. Kutzelnigg, Theor. Chim. Acta 68 (1985) 445.
[2] S. Ten-no, Chem. Phys. Lett. 398 (2004) 56.
[3] S. Ten-no, J. Chem. Phys. 126 (2007) 014108.
[4] D.P. Tew, W. Klopper, C. Hättig, Chem. Phys. Lett. 452 (2008) 326.
[5] J. Noga, S. Kedžuch, J. Šimunek, S. Ten-no, J. Chem. Phys. 128 (2008) 174103.
[6] E.F. Valeev, T.D. Crawford, J. Chem. Phys. 128 (2008) 244113.
[7] T. Shiozaki, M. Kamiya, S. Hirata, E.F. Valeev, J. Chem. Phys. 129 (2008) 071101.
[8] D. Bokhan, S. Bernadotte, S. Ten-no, Chem. Phys. Lett. 469 (2009) 214.
[9] T. Shiozaki, M. Kamiya, S. Hirata, E.F. Valeev, J. Chem. Phys. 130 (2009) 054101.
[10] G. Knizia, T.B. Adler, H.-J. Werner, J. Chem. Phys. 130 (2009) 054104.
[11] J. Yang, C. Hättig, J. Chem. Phys. 130 (2009) 124101.
[12] A. Köhn, J. Chem. Phys. 130 (2009) 131101.
[13] S. Kedžuch, M. Milko, J. Noga, Int. J. Quantum Chem. 105 (2005) 929.
[14] W. Klopper, C.C.M. Samson, J. Chem. Phys. 116 (2002) 6397.
[15] E.F. Valeev, Chem. Phys. Lett. 395 (2004) 190.
[16] W. Klopper, W. Kutzelnigg, Chem. Phys. Lett. 134 (1987) 17.
[17] A.J. May, E.F. Valeev, R. Polly, F.R. Manby, Phys. Chem. Chem. Phys. 7 (2005) 2710.
[18] D.P. Tew, W. Klopper, J. Chem. Phys. 123 (2005) 074101.
[19] E.F. Valeev, Libint, machine-generated library for efficient evaluation of molecular integrals over Gaussian, 2007.
[20] A.J. May, F.R. Manby, J. Chem. Phys. 121 (2004) 4479.
[21] S.F. Boys, Proc. Roy. Soc. London, Ser. A 200 (1950) 542.
[22] L.E. McMurchie, E.R. Davidson, J. Comput. Phys. 26 (1978) 218.
[23] J.A. Pople, W.J. Hehre, J. Comput. Phys. 27 (1978) 161.
[24] S. Obara, A. Saika, J. Chem. Phys. 84 (1986) 3963.
[25] M. Head-Gordon, J.A. Pople, J. Chem. Phys. 89 (1988) 5777.
[26] P.M.W. Gill, J.A. Pople, Int. J. Quantum Chem. 40 (1991) 745.
[27] T.P. Hamilton, H.F. Schaefer III, Chem. Phys. 150 (1991) 163.
[28] S. Ten-no, Chem. Phys. Lett. 211 (1993) 259.
[29] T. Yanai, K. Ishida, H. Nakano, K. Hirao, Int. J. Quantum Chem. 76 (2000) 396.
[30] H. Nakai, M. Kobayashi, Chem. Phys. Lett. 388 (2004) 50.
[31] H.F. King, M. Dupuis, J. Comput. Phys. 21 (1976) 144.
[32] M. Dupuis, J. Rys, H.F. King, J. Chem. Phys. 65 (1976) 111.
[33] J. Rys, M. Dupuis, H.F. King, J. Comput. Chem. 4 (1983) 154.
[34] R. Lindh, U. Ryu, B. Liu, J. Chem. Phys. 95 (1991) 5889–5897.
[35] S. Ten-no, O. Hino, Int. J. Mol. Sci. 3 (2002) 459.
[36] S. Ten-no, private communication.
[37] K. Ishida, J. Chem. Phys. 98 (1993) 2176.
[38] R. Lindh, Electron repulsion integrals, in: H.F. Schaefer III (Ed.), Encyclopedia of Computational Chemistry, vol. 4, Wiley, Chichester, 1998.
[39] K. Yasuda, J. Comput. Chem. 29 (2008) 334.
[40] W. Gautschi, Orthogonal Polynomials, Computation and Application, Oxford University Press, Oxford, 2004.
[41] N. Flocke, V. Lotrich, J. Comput. Chem. 29 (2008) 2722–2736.
[42] N. Flocke, J. Chem. Phys. (2009), in press.
[43] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C++, Cambridge University Press, New York, 2002.
[44] The Gmp and Mpfr libraries are used with a wrapper called Mpfr C++ written by P. Holoborodko.
[45] T. Shiozaki, An electronic structure program for molecules and solids, Gainesville, FL, 2009, in collaboration with S. Hirata.
[46] T. Shiozaki, S. Hirata, Phys. Rev. A 76 (2007) 040503(R).
[47] A.D. Becke, R.M. Dickson, J. Chem. Phys. 89 (1988) 2993.
[48] M. Dupuis, H.F. King, Int. J. Quantum Chem. 11 (1977) 613.
[49] R.A. Kendall, T.H. Dunning Jr., R.J. Harrison, J. Chem. Phys. 96 (1992) 6796.