On the topic of Lebedev quadrature

Ruocheng Han

July 2022

Lebedev quadrature has been employed in the quantum chemistry codes for the numerical integration.

The integration of a given function, e.g. electron density $\rho(r, \theta, \phi)$, on a unit sphere is

$$I[\rho](r) = \int_0^{2\pi} d\phi, \int_0^{\pi} \rho(r, \theta, \phi) sin\theta d\theta \tag{1}$$

where (r, θ, ϕ) denotes the spherical coordinates. Note that we omit the integration on r, which is usually estimated by the Gauss–Legendre quadrature.

Numerical approximation to the above spherical integral can be written as

$$I[\rho](r) \approx 4\pi \sum_i w_i \rho(r, \theta_i, \phi_i), \tag{2}$$

where w_i is the weight of the grid point i on the sphere.

Lebedev grid points are constructed based on O_h point group (octaheral symmetry). An example of O_h symmetry is cubic, with its face, edge, and vertex are invariant under the octaheral rotation. These three are also the simplest examples of Lebedev grid points (in Cartesian coordinates)

$$\begin{aligned}
&a^1: \ (\pm 1, 0, 0),\ (0, \pm 1, 0),\ (0, 0, \pm 1) \\
&a^2: \ \frac{1}{\sqrt{2}}(\pm 1, \pm 1, 0),\ \frac{1}{\sqrt{2}}(0, \pm 1, \pm 1),\ \frac{1}{\sqrt{2}}(\pm 1, 0, \pm 1) \\
&a^3: \ \frac{1}{\sqrt{3}}(\pm 1, \pm 1, \pm 1),\ \frac{1}{\sqrt{3}}(\pm 1, \pm 1, \pm 1),\ \frac{1}{\sqrt{3}}(\pm 1, \pm 1, \pm 1)
\end{aligned} \tag{3}$$

on the unit sphere. a^1, a^2, and a^3 are the basic sets of grid points following the notation in the original paper [1].

Besides, there are another three sets of points b^k, c^k, and d^k

$$\begin{aligned}
&b^k: \ (\pm m_k, \pm l_k, \pm l_k),\ (\pm l_k, \pm m_k, \pm l_k),\ (\pm l_k, \pm l_k, \pm m_k) \\
&c^k: \ (\pm p_k, \pm q_k, 0),\ (\pm q_k, \pm p_k, 0),\ (\pm p_k, 0, \pm q_k), \\
&(\pm q_k, 0, \pm p_k),\ (0, \pm q_k, \pm p_k),\ (0, \pm p_k, \pm q_k) \\
&d^k: (\pm r_k, \pm u_k, \pm w_k),\ (\pm r_k, \pm w_k, \pm u_k),\ (\pm u_k, \pm w_k, \pm r_k), \\
&(\pm u_k, \pm r_k, \pm w_k),\ (\pm w_k, \pm r_k, \pm u_k),\ (\pm w_k, \pm u_k, \pm r_k),
\end{aligned} \tag{4}$$

with the normalization criteria

$$\begin{aligned}
&2l_k^2 + m_k^2 = 1, \\
&p_k^2 + q_k^2 = 1, \\
&r_k^2 + u_k^2 + w_k^2 = 1.
\end{aligned} \tag{5}$$

The combination of six sets also follow the O_h symmetry, and the integration can be approximated by N grid points using

$$\begin{aligned} I_N[\rho](r) =& A_1 \sum_i^{6} \rho(a_i^1) + A_2 \sum_i^{12} \rho(a_i^2) + A_3 \sum_i^{8} \rho(a_i^3) \\ &+ \sum_k^{N_B} B_k \sum_i^{24} \rho(b_i^k) + \sum_k^{N_C} C_k \sum_i^{24} \rho(c_i^k) + \sum_k^{N_D} D_k \sum_i^{48} \rho(d_i^k), \end{aligned} \tag{6}$$

where total number of grid points are the sum

$$N = 26 + 24(N_B + N_C) + 48N_D. \tag{7}$$

The number of Lebedev grid points utilized in my code [2, 3] is 350, 6 in a^1, 8 in a^1, 6×24 in b^k, 2×24 in c^k, and 3×48 in d^k. The example code of generating grid points a^k-d_k on a unit sphere can be found in the attached python script *Lebedev_quad_example.py*.

The weights of grid points from a single set is $1/N$ since they are evenly distributed on the sphere. However, obtaining the weights of the combined grid points from different sets are not trivial, and a set of nonlinear equations needs to be solved.

This is different from the Gaussian quadrature, where the analytical expression in polynomials is given (e.g Legendre polynomials for the Gauss–Legendre quadrature). Nevertheless, Lebedev quadrature has its own advantage that it follows a high level symmetry O_h and thus is more efficient for the sampling of the sphere. One can have a intuitive view from Fig. 1 in Ref. [4], which shows the comparison of Gauss–Legendre grids (product of one dimension) and Gaussian grids.

The available Python package for generating Lebedev grids and weights is quadpy [5] and a C implementation is also available [6]. Both weights are hard-coded.

References

[1] V.I. Lebedev. Quadratures on a sphere. *USSR Computational Mathematics and Mathematical Physics*, 16(2):10–24, January 1976.

[2] Møller–plesset correlation energy density calculator. *https://gitlab.uzh.ch/lubergroup/energydensity*.

[3] Ruocheng Han, Mauricio Rodríguez-Mayorga, and Sandra Luber. A machine learning approach for MP2 correlation energies and its application to organic compounds. *Journal of Chemical Theory and Computation*, 17(2):777–790, January 2021.

[4] Casper HL Beentjes. Quadrature on a spherical surface. *http://people.maths.ox.ac.uk/beentjes/Essays*, 2015.

[5] Your one-stop shop for numerical integration in python. *https://pypi.org/project/quadpy/*.

[6] A c code which computes a lebedev quadrature rule over the surface of the unit sphere in 3d. *https://people.sc.fsu.edu/j̃burkardt/c_src/sphere_lebedev_rule/sphere_lebedev_rule.html*.