Transformation of Harmonics for Molecular Calculations

Michael P. Barnett*

Department of Chemistry, Princeton University, Princeton, New Jersey 08544-1009

Received December 11, 2002

The evaluation of multicenter integrals over Gaussian and exponential-type orbitals requires the rotation and translation of spherical harmonics. We used computer algebra (CA) to construct a machine readable table of formulas for these transformations that support unrestricted precision numerical computations. Extensive checks are provided. The calculation is presented in an easy to follow top-down functional programming style that is very powerful and can be applied to many problems of computational chemistry.

INTRODUCTION

Spherical harmonics that are expressed in one coordinate system often have to be rotated and/or moved to another, for example, in the early stages of the reduction of multicenter molecular integrals containing STOs (Slater-type orbitals)[1] and GTOs (Gaussian-type orbitals).[2] The definitive treatment of harmonic transformations by classical methods was given by Steinborn and Ruedenberg 30 years ago.[3] This was extended in ref 4, and Guseinov discusses the topic under the rubric "multipole operator expansion" − see, for example, ref 5. Different authors use many different notations and conventions and often report their results in the language of theoretical physics. The problem can be handled, however, by very simple mathematical methods, given the resources of modern computer algebra (CA). This note describes how we

1. constructed some machine readable tables of formulas to translate and rotate solid spherical harmonics and
2. checked the results.

Checking is a matter of paramount importance in symbolic computations − see, for example, refs 6 and 7 − that has received little attention in the literature. The tabulation was facilitated by computer instructions that verbalize some basic components of conventional mathematical analysis, using the author's Mathscape package[8,9] that is coded in Mathematica.[10] This software is the key element of our ongoing work on two-center molecular integrals over STOs[7,11−13] and two-electron atomic energy levels.[14]

CONVENTIONS

The analysis and the calculations in this paper are based on the Laplacian surface spherical harmonics discussed, for example, by Hobson.[15] The general form, repeated in eq 103 of ref 3, is displayed here as eq 1. We use Hobson's definition of the associated Legendre function $P_l^m(\cos\theta)$.[15] The formulas in the standard reference source[16] and the LegendreP built-in function of Mathematica follow Hobson's definition. Some chemists prefer the functions that originate in the theory of angular momentum, particularly those that contain the Condon-Shortley phase factor $(-1)^m$ (see eq 106 of ref 3). Our analysis can be adapted to different conventions. Alternatively, our final results can be multiplied by appropriate factors. In ref 3, the notation $[P_l^m(\zeta)]_{\mathrm{H}}$ and $[P_l^m(\zeta)]_{\mathrm{CS}}$ for the Legendre functions used by Hobson and by Condon and Shortley makes the choice of convention very clear.

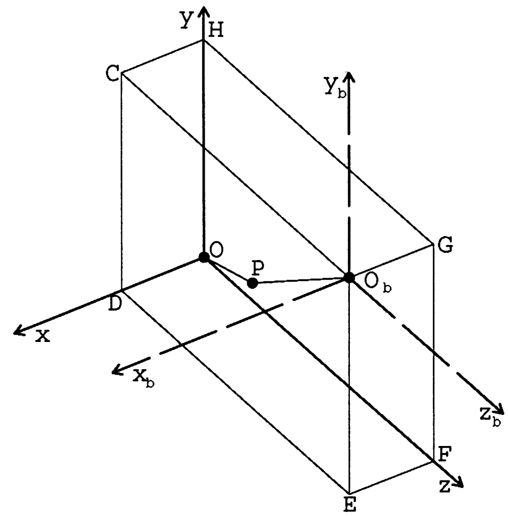

Figure 1. Coordinate systems.

We work with real harmonics in the present paper to enable simple geometrical arguments. Also, the use of real harmonics and the classical Legendre functions makes the analysis more accessible to computer scientists − it is based entirely on formulas in the familiar source[16] and it can be understood without knowledge of angular momentum theory. This accessibility helps collaborative efforts which enhance both the development and the application of computer algebra. In related work, however, we use complex harmonics, containing $\exp(\imath m\phi)$ for $-l \le m \le l$, because these simplify the major part of the reduction of the molecular integrals to closed formulas.[12] The conversion to real orbitals is made at the end, using de Moivre's theorem and Hobson's formula that is displayed here as eq 2. This formula appears as eq 107 in ref 3 and as eq 8.752-2 in ref 16. The use of real and complex orbitals in molecular integrals is reviewed in ref 17.

For the problem at hand, we use elementary trigonometry and the polynomial forms of surface harmonics without recourse to specialized auxiliary functions or awkward notations. Let **r** be the vector from the origin of a reference Cartesian system *OXYZ* to an arbitrary point *P* with Cartesian coordinates (x,y,z) and the corresponding polar coordinates (r,θ,ϕ), that is, $\mathbf{r} = \mathbf{i}x + \mathbf{j}y + \mathbf{k}z$. Introduce a displaced

* Corresponding author e-mail: michaelb@princeton.edu.

10.1021/ci0256539 CCC: $25.00
Published on Web 06/13/2003

Cartesian coordinate system $O_bX_bY_bZ_b$ with origin at $\mathbf{b} = \mathbf{i}b_x + \mathbf{j}b_y + \mathbf{k}b_z$ in the $OXYZ$ system. Denote the Cartesian coordinates in the $O_bX_bY_bZ_b$ system by (x_b,y_b,z_b) and the corresponding polar coordinates by (r_b,θ_b,ϕ_b), that is, $\mathbf{r_b} = \mathbf{i}x_b + \mathbf{j}y_b + \mathbf{k}z_b$. Figure 1 displays this notation on a pair of left-handed systems. C, E, G, D, H, and F are the projections of O_B onto the OXY, OXZ, and OYZ planes and OX, OY, and OZ axes, respectively; b_x, b_y, and b_z are the lengths OD, OH, and OF; and $\overrightarrow{OP}$ and $\overrightarrow{O_bP}$ are $\mathbf{r}$ and $\mathbf{r_b}$.

$$S_l(\theta,\phi) = \sum_{m=0}^{l}\left[A_m\cos m\phi + B_m\sin m\phi)P_l^m(\cos\theta)\right] \tag{1}$$

$$P_l^{-m}(\cos\theta) = (-1)^m\frac{(l-m)!}{(l+m)!}P_l^m(\cos\theta) \tag{2}$$

$$r^l P_l^m(\cos\theta)(\cos|\sin)m\phi \tag{3}$$

$$(r^{l-m-\varpi})\times(r^{\varpi}\cos^{\varpi}\theta)\times(r^{\varsigma}\sin^{\varsigma}\theta\sin^{\varsigma}\phi))\times(r^{m-\varsigma}\sin^{m-\varsigma}\theta\cos^{m-\varsigma}\phi) \tag{4}$$

SOME PROPERTIES OF HARMONICS

The main problem to be considered consists of expressing the solid spherical harmonic $r_b^{l'}P_{l'}^{m'}(\cos\,\theta_b)(\cos|\,\sin)m'\phi_b$ centered on B as a finite sum of solid spherical harmonics of the form 3 centered on the reference origin, when O_bX_b, O_bY_b, and O_bZ_b are parallel to OX, OY, and OZ, respectively. The indexes l' and l' are non-negative integers, and $0 \le m' \le l'$, $0 \le m \le l$. These restrictions on (l',l,m',m) follow from the genesis of the surface harmonics as solutions of a differential equation connected to the angular momentum of a particle in a quantum theoretical context. Equivalent mathematical conditions impose the same restrictions in other applications of potential theory.

The problem is shown to be valid by reference to three results in ref 16. Formula 8.812 shows that $P_l^m(\cos\,\theta)$ can be written as a finite sum of terms $\sin^m\theta\cos^{\varpi}\theta$ where $\varpi = l - m$, $l - m - 2$, ..., $l - (m \bmod 2)$. Formulas 1.331 and 1.333 show that $\sin m\phi$ and $\cos m\phi$ can be written as finite sums of terms $\cos^{m-\varsigma}\phi\sin^{\varsigma}\phi$ where $\varsigma = 1, 3, ..., m - (m+1 \bmod 2)$ and $\varsigma = 0, 2, ..., m - (m \bmod 2)$, respectively. Consequently (3) can be written as a finite sum of terms of the form 4, where ϖ has the parity of $l - m$ and $0 \le \varsigma \le m$. Hence, (3) can be written as a finite sum of terms $r^{\nu_r}x^{\nu_x}y^{\nu_y}z^{\nu_z}$ where ν_r, ν_x, ν_y, and ν_z are non-negative integers, $\nu_r + \nu_x + \nu_y + \nu_z = l$, $\nu_x + \nu_y = m$, and ν_r takes values 0, 2,..., $2\lfloor(l - m)/2\rfloor$. Consequently, the harmonics 3 for $l = 0$ to some $\mathbf{l}$, $0 \le m \le l$ are linear in the $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ for $\nu_x + \nu_y + \nu_z \le \mathbf{l}$. Hence, $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ for any triple of non-negative integers (ν_x,ν_y,ν_z) is linear in the harmonics 3 with $l \le \nu_x + \nu_y + \nu_z$. Also, the l and m in these harmonics have the same parities as $\nu_x + \nu_y + \nu_z$ and $\nu_x + \nu_y$, respectively.

SUMMARY OF THE CALCULATION

The calculation consists of four steps.

1. Single-term multinomials $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ are expressed as sums of solid spherical harmonics 3 for $l = 0$ to some $\mathbf{l}$, $m = 0$ to l — see eq 5.

2. A subset of these equations, that constitutes a left-triangular system, is solved for the surface harmonics $P_l^m(\cos\,\theta)(\cos|\,\sin)m\phi$, leading to the list of equations 6 for the solid harmonics.

3. In this second set of equations, we

(a) replace (x,y,z,ϕ,θ) by $(x_b,y_b,z_b,\theta_b,\phi_b)$,

(b) replace (x_b,y_b,z_b) by $(x-b_x,y-b_y,z-b_z)$ and expand, and

(c) replace the multinomial terms that contain (x,y,z) by harmonics that contain (r,θ,ϕ), using the equation list 5.

This gives a set of equations 7 that relate harmonics in the (r_b,θ_b,ϕ_b) system to harmonics in (r,θ,ϕ).

4. The final results are checked by

(a) reversing the translation,

(b) evaluating the formulas for spot values of the coordinates, and

(c) comparing with well-known formulas when the displaced origin is on an axis of the reference system.

To rotate the coordinates, we simply let $(x_b,y_b,z_b,r_b,\theta_b,\phi_b)$ stand for the coordinates in a system based on the same origin as the (x,y,z,r,θ,ϕ) system, oriented by the Euler angles (α,β,γ). Then step 3(b) is changed to use the equations for the (x_b,y_b,z_b), as just defined, in terms of $(x,y,z,\alpha,\beta,\gamma)$. The rest of the process is unaltered. Any linear transformation can be handled by the corresponding change in step 3(b).

These processes are summarized in the sections that follow. Further details that include complete program listings are in the Supporting Information.

$$\left\{1 = P_0(\cos\theta), z = rP_1(\cos\theta), \ldots, x^2y = -\frac{r^3}{5}P_1^1(\cos\theta)\sin\phi + \frac{r^3}{30}P_3^1(\cos\theta)\sin\phi - \frac{r^3}{60}P_3^3(\cos\theta)\sin 3\phi, \ldots\right\} \tag{5}$$

$$\left\{P_0(\cos\theta) = 1, rP_1(\cos\theta) = z, \ldots, r^2P_2^2(\cos\theta)\cos 2\phi = -3r^2 + 3(2x^2+z^2), \ldots\right\} \tag{6}$$

$$\begin{aligned}\Big\{&P_0(\cos\theta_b) = P_0(\cos\theta),\\ &r_bP_1(\cos\theta_b) = rP_1(\cos\theta) - b_z, \ldots,\\ &r_b^3P_3^2(\cos\theta_b)\sin 2\phi_b = r^3P_3^2(\cos\theta)\sin 2\phi\\ &\quad +5\Big(6b_xb_yrP_1(\cos\theta) + 2r^2P_2^1(\cos\theta)(b_x\sin\phi + b_y\cos\phi)\\ &\quad -b_z\Big(r^2P_2^2(\cos\theta)\sin 2\phi + 6\big(b_xb_y + rP_1^1(\cos\theta)(b_x\sin\phi + b_y\cos\phi)\big)\Big)\Big),\\ &\ldots\Big\}\end{aligned} \tag{7}$$

EXPANDING THE MULTINOMIALS

Consider the multinomial in eq 8 where the C coefficients are numeric. From the analysis leading to eq 4, f has a finite harmonic expansion. Using the orthogonality properties of Legendre and trigonometric functions, we write this as eq 9. The equation list 5 is produced by substituting single-term multinomials $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ in place of f and evaluating the integrals and sums.

Equation 9 is input to our Mathscape package in a Mathematica session as the script (M-1).We use cos[x], sin[x], sqrt[x], P[l,m,x], sum[i,j,k][s], and integral[x,a,b][s], instead of the built-in Mathematica names Cos, Sin, ..., to exercize tight control over the calculations. This avoids the premature evaluations that make some of the results indeterminate, erroneously, when the Mathematica names are used. It also permits the use of "radical expressions" containing explicit square roots. For example, cos $\pi/3$ is kept as $\sqrt{3}/2$ in the construction of eq 17 below. The selective avoidance of built-in names enables many other steps in the calculation, too. The Mathscape package includes modules to deal systematically with numerous objects that are denoted by "compound heads" such as sum[...] and integral[...].

$$f = \sum_{(\nu_x,\nu_y,\nu_z)\in S} C^{(l)}_{\nu_x,\nu_y,\nu_z} x^{\nu_x} y^{\nu_y} z^{\nu_z},$$
$$S = \{(\nu_x,\nu_y,\nu_z) | \nu_x \ge 0, \nu_y \ge 0, \nu_z \ge 0, \nu_x+\nu_y+\nu_z \le l, 0 \le l \le 1\} \quad (8)$$

$$\begin{aligned} f = \sum_{l=0}^{1} \Big[&\frac{2l+1}{4\pi} P_l(\cos\theta) \\ &\times \int_0^{\pi}\int_0^{2\pi} P_l(\cos\theta) f \sin\theta \mathrm{d}\theta \mathrm{d}\phi \\ &+ \sum_{m=1}^{l} \frac{2l+1}{2\pi}\frac{(l-m)!}{(l+m)!} P_l^m(\cos\theta) \\ &\times \int_0^{\pi} P_l^m(\cos\theta)\Big(\cos m\phi \int_0^{2\pi} f\cos m\phi \mathrm{d}\phi \\ &\quad + \sin m\phi \int_0^{2\pi} f \sin m\phi \mathrm{d}\phi\Big) \sin\theta \mathrm{d}\theta \Big] \end{aligned} \quad (9)$$

$$\begin{aligned} Y_{l,0,-1} &= P_l(\cos\theta), \\ Y_{l,m,-1} &= P_l^m(\cos\theta)\cos m\phi, \\ Y_{l,m,1} &= P_l^m(\cos\theta)\sin m\phi \end{aligned} \quad (10)$$

The inner summand is coded in a separate statement, using the same notation as the expression that contains it. The Mathematica implementation of most of the Mathscape elements that are used here is quite simple.

The equation list 5 was constructed by the Mathematica Table statement in the script (M-2). This applies an operator that we wrote for the current problem, under the name convert$xyzToPlm, to successive instances of eq 9. The iterator follows a Cantor enumeration of (ν_x,ν_y,ν_z). Syntactically, the operator is a Mathematica "function". We use the terms operator and function interchangeably, when discussing the roles of such objects.

```
eqn[harmonicExpansion] =
 f ==
  sum[l, 0, lmax][
   (2 l + 1)/(4 pi) *
   P[l, 0, cos[theta]] *
   integral[theta, 0, pi][
    integral[phi, 0, 2pi][
     f * P[l, 0, cos[theta]] sin[theta]] +
     sum[m, 1, l][innerSummand]]
```
(M-1)

```
xyzToPlmEquations =
 Table[
  eqn[harmonicExpansion]  //
   convert$xyzToPlm,
 {ll, 0, lbar}, {nuz, ll, 0, -1},
 {nux, ll-nuz, 0, -1}] // Flatten
```
(M-2)

```
convert$xyzToPlm =
 pipe[
  instantiateTheEquation,
  toTheRhs[
    expandTheNestedSum,
    toIntegrals[innermost][
     convert$xyzToPolar,
     useBuiltinIntegrate],
    toEachIntegral[
     convertPlmToTrig,
     ExpandAll,
     Distribute,
     useBuiltinIntegrate]]]
```
(M-3)

```
harmonicTranslations =
 PlmTo$xyzEquations //
  toEachElement[
   toTheRhs[convert$rTo$xyz],
   toBothSides[labelTheCoordinates],
   toTheRhs[
    convertRemote$xyzToReference$xyz,
    Expand,
    convertReference$xyzToPlm,
    Simplify]]
```
(M-4)

The general action of the Mathematica Table statement was mentioned in a recent paper in this journal.[18] The postfix notation $x//f$ is equivalent to $f[x]$. We write Mathematica statements this way to parallel the narrative style that lists actions from left to right on the printed page in verbal explanations — very often the postfix operator (function) is a composition of subsidiary operators. Reference 10 explains all the Mathematica functions and options that we use, indexed under the keywords and the symbolic abbreviations. The Mathscape names begin in lower case, except where noted.

In each cycle of the tabulation, the equation for the harmonic expansion is transformed by a few successive steps that are taken for granted in "pencil-and-paper" work but requires explicit attention when present mechanization tools are employed. These steps

1. set the exponents of (x,y,z) and the limit of the outer sum to values that correspond to the current values of the iterators,

2. rewrite the $\sum$ expression as an explicit "plus",

3. convert the $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ factor in each integrand to polar coordinates,

4. convert the surface harmonic to a sum of products $\cos^{\nu_c}\theta\sin^{\nu_s}\theta(\cos|\sin)m\phi$ within each integrand, and

5. integrate over ϕ and then over θ.

The process is performed mechanically by the operator (M-3). In this operator, pipe connotes left-to-right composition. It is a Mathscape keyword. So are the self-explanatory expandTheNestedSum and useBuiltinIntegrate and the "targeting functions" toTheRhs, toIntegrals[innermost], and toEachIntegral.

The Mathscape package supports a large suite of targeting functions that is open-ended. These direct action to subexpressions which are specified in many different ways, that can be extended as occasion demands. Targeting is used in convert$xyzToPlm$

1. to prevent the conversion of (x,y,z) to (r,θ,ϕ) on the left-hand side of the equation,

2. to preserve the $P_l^m(\cos\theta)(\cos|\sin)m\phi$ outside the integration from conversion, and

3. to break the overall integration into pieces that the Mathematica Integrate function can handle.

HARMONICS TO MULTINOMIALS

The equation list 5 for $\nu_x + \nu_y + \nu_z <= \mathbf{l}$ is a left triangular system that can be solved by iteration for the harmonics with $l <= \mathbf{l}$. To simplify the mechanics, we write the "unknowns" as the Y's defined in eq 10 and apply a powerful CA technique that works for any left triangular system. We use it extensively in other contexts to tabulate special functions symbolically by iterating n-term recurrence formulas.

In the first cycle of the present iteration, the first equation $(1 = Y_{0,0,-1})$ is solved for the first unknown $(Y_{0,0,-1})$. The result, $Y_{0,0,-1} = 1$, is converted to the replacement rule that changes the left-hand side into the right-hand side — in Mathematica syntax this is Y[0,0,−1] → 1. This rule is made the first element of a list that is extended in each successive cycle. In the second cycle, the list of rules (containing, albeit, just a single element so far) is applied to the second equation, to replace any occurrences of the first unknown. Then the result is solved for the second unknown. The corresponding replacement rule is constructed and appended to the growing list. In general, the ith cycle solves the equation that introduces the ith unknown, using the rules for the first $i - 1$ unknowns that were formed in earlier cycles.

The equations 5 overdetermine the $Y_{l,m,\sigma}$. These unknowns appear, for the first time, at positions 1−9, 11−18, 21−28, 31, 32, 36−40, ... in the list of equations. The cursor list consisting of these ordinals is constructed by a simple Mathematica statement. Then, in the ith cycle of the iteration, the ith item in the cursor list identifies the equation that is solved for the ith item in the list of names. The equation for each $Y_{l,m,\sigma}$ is multiplied by r^l, and eq 10 restores the P_l^m notation. The result is appended to the list PlmTo$xyzEquations.

We cannot solve directly for the solid harmonics containing r^l because, for example, the first equation that contains P_3 also contains $3r^3/5P_1(\cos\theta)$ which the pattern matching does not recognize as a product containing $rP_1(\cos\theta)$.

SHIFTING THE ORIGIN

The equation list 7 was constructed from the solution of the triangular system by the statement (M-4). Within this statement,

1. toEachElement focuses, one by one, on the successive elements of a list.

2. convertrToxyz changes r to $\sqrt{x^2+y^2+z^2}$ — only even powers of r occur at this stage, so the result stays rational.

3. labelTheCoordinates puts the subscript b on (y,z,r,θ,ϕ).

4. convertRemote$xyzToReference$xyz replaces (x_b,y_b,z_b) by $(x-b_x,y-b_y,z-b_z)$.

5. Expand and Simplify are elementary built-in functions of Mathematica.

6. convertReference$xyzToPlm replaces the multinomial terms in (x,y,z) by harmonics, using the equation list 5.

THE TABULATION

To get a sense of the calculation and to work out the checking procedures, we set $\mathbf{l} = 10$ and ran the three stages in succession. An E4500 8 × 400 MHz UltraSparcII with 8 GB RAM running Mathematica 4 under Unix took 1700 s to produce formulas for 286 multinomial terms $x^{\nu_x}y^{\nu_y}z^{\nu_z}$, 20 s to solve the equations for the 121 harmonics that these contain, and 1700 s to produce the 121 translation formulas. The output of the successive stages occupy 127KB, 14KB, and 780KB on disk. The computation times on an SGI Origin 200 4 × 270 MHz R12K with 2 GB RAM were very close to those on the UltraSparcII.

To streamline the tabulation, we

1. used closed expressions and recurrence formulas for the integrals over ϕ and θ,

2. checked the equation for each $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ as soon as it was formed, to see if it introduced a new harmonic, and solved for this if it did,

3. precomputed a table of formulas for the individual Legendre functions, instead of reconstructing these repeatedly,

4. used table lookup to streamline the back substitution,

5. organized the overall calculation in a Do loop that worked through the three stages for each l before proceeding to the next, and

6. collected terms and changed notation to make the results more compact.

This reduced the computation time for $l <= 10$ by a factor of 3. Step 3 still becomes excessively time-consuming for high l. Steps 1 and 2 were continued to $l = 40$ and can be extended by expressing the calculation in lower level Mathematica functions. Assigning numerical values to (b_x,b_y,b_z) makes step 3 run much faster, allowing its continuation to high l. The use of rationals, radicals (as in $\sqrt{3}/2$), and unrestricted precision real numbers provides accuracy that is determinate and as high as is needed.

ANALYTIC INTEGRATION

Although the formulas to integrate ϕ and θ over the ranges $(0,\pi)$ and $(0,2\pi)$ are not explicit in ref 16, they follow directly from the eqs 3.621-5 and 8.384-1 therein for the classical Beta function (see, e.g., ref 19). Because only odd powers of $\sin\theta$ occur in the outer integration, the integrals over θ are covered by eq 13.

The streamlined integration over ϕ uses a simple recurrence scheme. Let $\mathrm{cors}_0 = \cos$ and $\mathrm{cors}_1 = \sin$ and define the g integrals by eq 14. Then $g_{q,k,l,m} = 0$ if $k + q$ is odd, or

$$\int_0^{\pi/2} \sin^{\mu-1} x \cos^{\nu-1} x \mathrm{d}x = \frac{1}{2} B\left(\frac{\mu}{2}, \frac{\nu}{2}\right) \quad (11)$$

$$B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} \quad (12)$$

$$\int_0^{\pi} \sin^j \theta \cos^k \theta \mathrm{d}\theta = \frac{2^{\frac{i+1}{2}} \left(\frac{j-1}{2}\right)!(k-1)!!}{(j+k)!!}, \quad \text{if } j \text{ is odd and } k \text{ is even} \quad (13)$$

$$g_{q,k,l,m} = \int_0^{2\pi} \sin^k \phi \cos^l \phi \operatorname{cors}_q m\phi \, \mathrm{d}\phi \quad (14)$$

$$g_{0,l,k,0} = \frac{(k-1)!!(l-1)!!\pi}{2^{\frac{k+l}{2}-1} \left(\frac{k+l}{2}\right)!} \quad (15)$$

$$g_{q,k,l,m} = (-1)^{q+1} g_{1-q,k+1,l,m-1} + g_{q,k,l+1,m-1} \quad (16)$$

$l + m + q$ is odd, or $q = 1$ and $m = 0$, or $k + l < m$. Nonzero values are computed from the starting values eq 15 and the recurrence formula eq 16.

ROTATING THE AXES

The equations in step 4 can be replaced by any linear transformation of the coordinates. For example, let (x_b,y_b,z_b) denote a system that is brought into coincidence with the (x,y,z) system by rotation through the Eulerian angles (α,β,γ). Then the formulas for harmonics of (θ_b,ϕ_b) are converted to harmonics of (θ,ϕ) by scripts (M-5) and (M-6).

```
harmonicRotations =
 PlmTo$xyzEquations //
  toEachElement[
   toTheRhs[convert$rTo$xyz],
   toBothSides[labelTheCoordinates],
   toTheRhs[
    convertRotated$xyzToReference$xyz,
    Expand,
    convertReference$xyzToPlm,
    cosmeticize]]                              (M-5)
eqn[rotation] =
 eqn[rotation] =
 Thread[{x[B], y[B], z[B]} ==
{{cos[alpha] cos[beta] cos[gamma] -
    sin[alpha] sin[gamma],
   sin[alpha] cos[beta] cos[gamma] +
    cos[alpha] sin[gamma],
   -sin[beta] cos[gamma]},
  {-cos[alpha] cos[beta] sin[gamma] -
    sin[alpha] cos[gamma],
    -sin[alpha] cos[beta] sin[gamma] +
    cos[alpha] cos[gamma],
    sin[beta] sin[gamma]},
   {cos[alpha] sin[beta],
     sin[alpha] sin[beta],
     cos[beta]}}. {x, y, z}]                   (M-6)
```

The formulas for the individual harmonics in terms of symbolic (α,β,γ) are unwieldy, but precise formulas can be constructed for simple molecular and crystal geometries by the use of radical arithmetic. For example, the Web site[20] animates the oscillation of a set of axes about the orientation $(\alpha = \pi/6, \beta = \pi/3, \gamma = \pi/4)$. Assigning the sines and cosines of these to the exact expressions containing $\sqrt{2}$ and $\sqrt{3}$ and then invoking the harmonicRotation procedure gives a table of results that are typified by eq 17.

$$\begin{aligned} r_b^3 P_3^2(\cos\theta_b)\sin 2\phi_b = \\ &-\frac{45r^3}{8}P_3(\cos\theta) \\ &+\frac{5r^3}{32}P_3^1(\cos\theta)(3\cos\phi + \sqrt{3}\sin\phi) \\ &+\frac{r^3}{32}P_3^2(\cos\theta)(\cos 2\phi + \sqrt{3}\sin 2\phi) \\ &+\frac{5\sqrt{3}r^3}{32}P_3^3(\cos\theta)\sin 3\phi \end{aligned} \quad (17)$$

$$\int_0^{2\pi} \operatorname{cors}_p^k \phi \operatorname{cors}_q m\phi \, \mathrm{d}\phi = \frac{(-1)^{p\times\lfloor(m-q)/2\rfloor}\pi}{2^{k-1}} \binom{k}{(k-m)/2} \quad (18)$$

The Web site[21] displays a recent summary of the different conventions that are used to specify 3D rotations. Any of these can be accommodated by changing the statement eqn[rotation].

CHECKING

The equation list 5 is checked by an operator check$xyzToPlm that treats each constituent equation as follows.

1. The (x,y,z) in the left-hand side are converted to polar coordinates (r,θ,ϕ).
2. Each $P_l^m(\cos\theta)$ on the right-hand side is converted to a polynomial in $\cos\theta$ and $\sin\theta$.
3. Each $\cos m\phi$ and $\sin m\phi$ is converted to a polynomial in $\cos\phi$ and $\sin\phi$.
4. All odd powers of $\sin\theta$ greater than 1 are divided and multiplied by $\sin\theta$ to give an even power of $\sin\theta$ times an isolated $\sin\theta$ that cannot coalesce (because it is wrapped in the Mathematica "head" HoldForm). Then the rule to replace $\sin\theta$ by $(1 - \cos^2\theta)^{1/2}$ is applied to each even power of $\sin\theta$.
5. Both sides are expanded (multiplied out). The equation collapses to True because the sides are identical.

To check the equation list 6, we just transpose the actions that are applied to the left- and right-hand sides of the target equations when checking the equation list 5.

To check the translation formulas, the roles of (x,y,z,r,θ,ϕ) and $(x_b,y_b,z_b, r_b,\theta_b,\phi_b)$ in the equation list 7 are interchanged, and the signs of (b_x,b_y,b_z) are reversed. Then the replacement rules that correspond to these reversed translations are applied to the right-hand side of each equation. Expanding the results collapses them to the value True. The formulas for rotations are checked correspondingly, using $(\gamma,-\beta,\alpha)$ to reverse the rotation.

The $g_{q,k,0,m}$ and $g_{q,0,l,m}$ of eq 14 were spot-checked by eq 18 when $k + m$ is even and $k \geq m$ and $(p,q) \neq (0,1)$ is true, and $(p,q) = (1,0)$ and k is odd is false, and $(p,q) = (1,1)$ and k is even is false, and is zero otherwise, under the overall assumption that $m >= 0$.

A further test referred to the cases O_b on OX or OY or OZ. By rotating both sets of axes (in the first two cases) to make OZ and OZ_b collinear and then using the translation formulas for $l <= 10$, $0 <= m <= l$, we constructed equations that matched the classical formula containing binomial coefficients (eq 25 in section 89 of ref 15).

To produce the electronic file that was used to typeset the manuscript, we used a variation of the methods reported previously.[22,23] Verbatim code was extracted from the Mathematica files that controlled the calculation and embedded in the LaTeX file mechanically. The LaTeX coded formulas were extracted from the typesetting file and converted mechanically to Mathematica representations that were checked against the input and immediate output of the calculations.

DISCUSSION

The calculation that has been described illustrates several ways in which computer algebra is impacting the conduct of scientific research and the dissemination of results. These include

1. the ability to use particular mathematical methods in circumstances that make them too unwieldy for manual application,
2. the opportunity for increased accuracy in the application of these methods,
3. the creation of audit trails,
4. the opportunity for extensive checks,
5. the collegial development of validated machine-readable material, and
6. the accurate dissemination of results for visual inspection, free of possible transcription and proof reading errors (in the final typeset version that the author submitted to the journal, an example page of which is printed in the Appendix to this paper).

To reap these benefits, we have to spell out the details of mathematical operations at a level that is not customary in computational chemistry. This really requires little more than providing the commentary on a derivation during a lecture; however, it is the *modus operandi* in logic, foundations of mathematics, and proof theory. The identification and cataloging of the details, moreover, points the way to greater mechanization, and it suggests topics for mathematical abstraction. Most of the elementary operations are obvious, but several only came to light during the development of Mathscape applications.

For example, the procedure to check the P_l^m to (x,y,z) equations illustrates some tactics that were evolved in earlier work and which we have used frequently. At one point in this procedure, we reduce even (odd) powers of $\sin\theta$ to even powers of $\cos\theta$ (times $\sin\theta$), using script (M-7). The Mathematica patterns mu_?OddQ and mu_?EvenQ are matched by odd and even integers, respectively. The first item in the pipe converts $\sin^{\mu}\theta$ to $\sin^{\mu-1}\theta \times \sin\theta$ when μ is odd. The divideAndMultiplyBy function sequesters the solitary factor $\sin\theta$ by putting HoldForm around it. The second item in the pipe converts the unaltered (reduced) power of $\sin\theta$ when the original power was even (odd). A similar tactic uses step-up/step-down operations that convert $n!$ to $n \times (n-1)!$ and to $(n+1)!/(n+1)$. Correspondingly, the double factorial $n!! = n \times (n-2) \times ... \times 4 \times 2$ when n is even, and $n \times (n-2) \times ... \times 3 \times 1$ when n is odd is converted to $n \times (n-2)!!$ and to $(n+2)!!/(n+2)$. These operations are used in the inductive proof of equations that involve binomial expansions, such as eq 18.

To use built-in Mathematica functions of several variables as unary functions (operators) that can be composed, we wrap them in "compound heads". For example, the present calculation uses script (M-8) and script (M-9). We also use wrappers to connote options, as in script (M-10). The construction *lhs*==*rhs* // solveLinear[x] solves $lhs = rhs$ for x, when the equation is linear.

```
pipe[
 toEach[sin[theta]^mu_?OddQ][
  divideAndMultiplyBy[sin[theta]]],
 toEach[sin[theta]^mu_?EvenQ][replaceSin],
 ReleaseHold, Expand]
```
(M-7)

```
drop[n_][s_] := Drop[s, n]
```
(M-8)

```
findItsFirstPositionIn[s_][v_] :=
 Position[s, v][[1, 1]]
```
(M-9)

```
trigExpand[s_] := Expand[s, Trig -> True]
```
(M-10)

```
eqn[sinSqdPlusCosSqd] =
  a*cos[x]^2 + a*sin[x]^2 == a
```
(M-11)

```
assume[assumption][
  expression1 == expression2]
```
(M-12)

```
expression3==
  if[test, expression4, expression5]
```
(M-13)

In an effort to avoid errors caused by mental slips and inaccurate transcription, each of the formulas that we have used in CA sessions and reported over the past several years was either

1. copied just once from a reference source and validated mechanically by consistency tests,
2. the result of executing a Mathscape script that we wrote to perform the derivation, or
3. the keyboarded version of a result that we derived manually and then wrote a Mathscape script to prove by induction.

We never retype the formulas in either the canonical or modified forms. Instead, they are retrieved and modified algorithmically whenever they are needed. For example, the present calculation uses the rule to replace $1 - \cos^2\theta$ by $\sin^2\theta$ and the rule to replace $\sin\theta$ by $\sqrt{1-\cos^2\theta}$. Both rules are constructed from the canonical form (M-11) in the Mathscape library. The representation of particular forms of reasoning constitutes a further challenge. For example, the proofs of eqs 13−16 and 18 require the consolidation of

1. assumptions that are made during a proof that deals with several cases in turn, for example, k is even or odd combined with l is even or odd, and
2. conditional identities, used in the proof, that take different forms when, for example, $k + l$ is even and when it is odd.

To deal with this, we write the equation to be proved under a particular assumption as (M-12) and the conditional identity as (M-13) or in related forms that express multiple cases. Typically, *expression*2 contains (or is converted to a form that contains) *expression*3. This is replaced by the if construction. Then the Mathscape procedure coalesceAssumeAndIf is applied. This propagates *assumption* downward and resolves *test*, producing a new conditional identity.

The mechanical exhaustion of alternative cases is a real need. All the work in refs 7, 12, and 13 depends in part on eq 18 of ref 11. We derived an erroneous version of this recurrence formula by hand, some 50 years ago.[24] The recent mechanized reconstruction dealt with a limit containing Bessel functions separately, when the order is non-negative and negative, and found a conditionalterm that had been omitted.

APPENDIX

TRANSFORMATION OF HARMONICS 3

where ν_r,ν_x,ν_y and ν_z are non-negative integers, $\nu_r+\nu_x+\nu_y+\nu_z = l$, $\nu_x+\nu_y = m$ and ν_r takes values $0, 2, \ldots, 2\lfloor(l-m)/2\rfloor$. Consequently, the harmonics (3) for $l = 0$ to some $\mathbf{l}$, $0 \le m \le l$ are linear in the $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ for $\nu_x + \nu_y + \nu_z \le \mathbf{l}$. Hence, $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ for any triple of non-negative integers (ν_x, ν_y, ν_z) is linear in the harmonics (3) with $l \le \nu_x + \nu_y + \nu_z$. Also, the l and m in these harmonics have the same parities as $\nu_x + \nu_y + \nu_z$ and $\nu_x + \nu_y$, respectively.

SUMMARY OF THE CALCULATION

The calculation consists of four steps.

1. Single-term multinomials $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ are expressed as sums of solid spherical harmonics (3) for $l = 0$ to some $\mathbf{l}$, $m = 0$ to l — see Eq. (5).
2. A subset of these equations, that constitutes a left-triangular system, is solved for the surface harmonics $P_l^m(\cos\theta)(\cos|\sin)m\phi$, leading to the list of equations (6) for the solid harmonics.
3. In this second set of equations, we
   (a) replace (x, y, z, ϕ, θ) by $(x_b, y_b, z_b, \theta_b, \phi_b)$,
   (b) replace (x_b, y_b, z_b) by $(x-b_x, y-b_y, z-b_z)$ and expand,
   (c) replace the multinomial terms that contain (x, y, z) by harmonics that contain (r, θ, ϕ), using the equation list (5).

   This gives a set of equations (7) that relate harmonics in the (r_b, θ_b, ϕ_b) system to harmonics in (r, θ, ϕ).
4. The final results are checked by
   (a) reversing the translation,
   (b) evaluating the formulas for spot values of the coordinates,
   (c) comparison with well known formulas when the displaced origin is on an axis of the reference system.

To rotate the coordinates, we simply let $(x_b, y_b, z_b, r_b, \theta_b, \phi_b)$ stand for the coordinates in a system based on the same origin as the $(x, y, z, r, \theta, \phi)$ system, oriented by the Euler angles (α, β, γ). Then step 3(b) is changed to use the equations for the (x_b, y_b, z_b), as just defined, in terms of $(x, y, z, \alpha, \beta, \gamma)$. The rest of the process is unaltered. Any linear transformation can be handled by the corresponding change in step 3(b).

These processes are summarized in the sections that follow. Further details that include complete program listings are in the Supporting Information.

$$\Big\{1 = P_0(\cos\theta), z = rP_1(\cos\theta), \ldots,$$
$$x^2y = -\frac{r^3}{5}P_1^1(\cos\theta)\sin\phi + \frac{r^3}{30}P_3^1(\cos\theta)\sin\phi - \frac{r^3}{60}P_3^3(\cos\theta)\sin 3\phi, \ldots\Big\} \tag{5}$$

$$\Big\{P_0(\cos\theta) = 1, rP_1(\cos\theta) = z, \ldots,$$
$$r^2P_2^2(\cos\theta)\cos 2\phi = -3r^2 + 3(2x^2 + z^2), \ldots\Big\} \tag{6}$$

$$\Big\{P_0(\cos\theta_b) = P_0(\cos\theta),$$
$$r_bP_1(\cos\theta_b) = rP_1(\cos\theta) - b_z, \ldots,$$
$$r_b^3P_3^2(\cos\theta_b)\sin 2\phi_b = r^3P_3^2(\cos\theta)\sin 2\phi + 5\Big(6b_xb_yrP_1(\cos\theta) + 2r^2P_2^1(\cos\theta)(b_x\sin\phi + b_y\cos\phi) - b_z\big(r^2P_2^2(\cos\theta)\sin 2\phi + 6\big(b_xb_y + rP_1^1(\cos\theta)(b_x\sin\phi + b_y\cos\phi)\big)\big)\Big),$$
$$\ldots\Big\} \tag{7}$$

EXPANDING THE MULTINOMIALS

Consider the multinomial in Eq. (8) where the C coefficients are numeric. From the analysis leading to Eq. (4), f has a finite harmonic expansion. Using the orthogonality properties of Legendre and trigonometric functions, we write this as Eq. (9). The equation list (5) is produced by substituting single-term multinomials $x^{\nu_x}y^{\nu_y}z^{\nu_z}$ in place of f, and evaluating the integrals and sums.

ACKNOWLEDGMENT

We thank L. C. Allen and J. F. Capitani for their ongoing help and advice, C. W. Hillegas for technical support, and an anonymous referee for some very constructive comments.

Supporting Information Available: A supplementary note that (i) describes the control files which performed the calculations and (ii) gives some further details of the mathematical and programming methods; the concatenation of the control files and the files that logged the actual runs; and the concatenation of the tables for the Cartesian to harmonic transformations, the harmonic to Cartesian transformations, and the harmonic translations. This material is available free of charge via the Internet at http://pubs.acs.org.

REFERENCES AND NOTES

(1) Barnett, M. P. The Evaluation of Molecular Integrals by the Zeta-function Expansion. In *Methods of Computational Physics*; Alder, B., Fernbach, S., Rotenberg, M., Eds.; Academic Press: New York, 1963; Vol. 2, pp 95−153.
(2) Shavitt, I. The Gaussian Function in Calculations of Statistical Mechanics and Quantum Mechanics. In *Methods of Computational Physics*; Alder, B., Fernbach, S., Rotenberg, M., Eds.; Academic Press: New York, 1963; Vol. 2, pp 1−44.
(3) Steinborn, E. O.; Ruedenberg, K. *Adv. Quantum Chem.* **1973**, *7*, 1.
(4) Steinborn, E. O.; Filter, E. *Int. J. Quantum Chem. Symp.* **1975**, *9*, 435.
(5) Guseinov, I. I. Spherically Symmetrical Properties of Expansion Coefficients for Translation of Electric Multipole Moment Operator. *J. Mol. Struct. (THEOCHEM)* **1998**, *427*, 263.
(6) Gracey, J. A. Computation of Perturbative Renormalization Group Functions − the Large N-f Algorithm. *Comput. Phys. Commun*. **1998**, *115*, 113−123.
(7) Barnett, M. P. Digital Erosion in the Evaluation of Molecular Integrals. *Theor. Chem. Acc.* **2002**, *107*, 241−245.
(8) Barnett, M. P. Implicit Rule Formation in Symbolic Computation. *Computers Math. Applic.* **1993**, *26*, 35−50.
(9) Barnett, M. P.; Perry, K. R. Hierarchical Addressing in Symbolic Computation. *Computers Math. Applic.* **1994**, *28*, 17−35.
(10) Wolfram, S. *The Mathematica Book*, 4th ed.; Cambridge University Press: 1999.
(11) Barnett, M. P. Symbolic Calculation of Auxiliary Functions for Molecular Integrals over Slater Orbitals. *Int. J. Quantum Chem.* **2000**, *76*, 464−472.
(12) Barnett, M. P. Two-center Non-exchange Integrals over Slater Orbitals. *J. Chem. Phys.* **2000**, *113*, 9419−9428.
(13) Barnett, M. P. Molecular Integrals and Information Processing. *Int. J. Quantum Chem.* **2003**, in course of publication.
(14) Barnett, M. P.; Decker, T.; Krandick, W. Power Series Expansion of the Roots of a Secular Equation Containing Symbolic Elements: Computer Algebra and Moseley's Law. *J. Chem. Phys.* **2001**, *114*, 10265−10269.
(15) Hobson, E. W. *Theory of Spherical and Ellipsoidal Harmonics*; Cambridge University Press: Cambridge, 1931.
(16) Gradshteyn, I. I.; Ryzhik, I. M. *Tables of Integrals, Series and Products*; Academic Press: 1979.
(17) Steinborn, E. O. *Adv. Quantum Chem.* **1973**, *7*, 83.
(18) Cash, G. G. A Simple Program for Computing Characteristic Polynomials with Mathematica. *J. Chem. Inf. Comput. Sci*. **1999**, *39*, 833−834.
(19) Whittaker, E. T.; Watson, G. N. *A Course of Modern Analysis*; Cambridge University Press: 1927.
(20) Levitt, M. H. *Euler Angles* **2002**, www.mhl.soton.ac.uk/research/help/Euler.
(21) Weisstein, E. W. *Euler Angles* **2000**, http://mathworld.wolfram.com/EulerAngles.html. 1999.
(22) Barnett, M. P.; Perry, K. R. Symbolic computation for electronic publishing. *TUGboat* **1994**, *15*, 285−292.
(23) Barnett, M. P. Combining Mathematica and TeX. *TUGboat* **1998**, *19*, 147−158.
(24) Coulson, C. A.; Barnett, M. P. The Evaluation of Unit Molecular Integrals. In *Conference on Quantum Mechanical Methods in Valence Theory*; Mulliken, R. L., Ed.; National Academy Press: Washington, DC, 1951; pp 237−271.

CI0256539