

High-order analytic translation matrix elements for real-space six-dimensional polar Fourier correlations

David W. Ritchie

Department of Computing Science, University of Aberdeen, Aberdeen AB24 3UE, Scotland.
Correspondence e-mail: dritchie@csd.abdn.ac.uk

Analytic expressions are presented for calculating translations of high-order three-dimensional expansions of orthonormal real spherical harmonic and Gaussian-type or exponential-type radial basis functions. When used with real spherical harmonic rotation matrices, the resulting translation matrices provide a fully analytic method of calculating six-dimensional real-space rotational-translational correlations. The correlation algorithm is demonstrated by using an exhaustive search to superpose the steric density functions of a pair of similar globular proteins in a matter of seconds on a contemporary personal computer. It is proposed that the techniques described could be used to accelerate the calculation of *e.g.* real-space electron density correlations in molecular replacement, docking proteins into electron microscopy density maps, and searching the Protein Data Bank for structural homologues.

© 2005 International Union of Crystallography
Printed in Great Britain – all rights reserved

1. Introduction

Solving protein crystal structures by molecular replacement (MR) (Rossmann, 1990, 2001), fitting protein structures into electron microscopy (EM) density maps (Roseman, 2000; Rossmann, 2000; Frank, 2002), and predicting the docking mode of a pair of proteins (Katchalski-Katzir *et al.*, 1992; Ritchie & Kemp, 2000) are three examples of tasks which typically involve searching for the coordinate transformation that maximizes a six-dimensional correlation. Often, such correlations can be expressed as the overlap between one or more pairs of three-dimensional scalar functions, such as electron densities, and it is primarily this type of correlation that is considered here. For example, if some phase information is available from multiple anomalous dispersion (MAD) or multiple isomorphous replacement (MIR) data, or from a partial model, the phased translation function may be used to correlate a low-resolution electron density map with the density from a model structure to help locate the model in the crystal unit cell (Read & Schierbeek, 1988). This type of translational correlation may be accelerated using a three-dimensional fast Fourier transform (FFT). However, the correct solution can sometimes be missed if the rotational solution is in error. This can often be corrected by varying the rotational orientations (Fujinaga & Read, 1987) or by translating multiple rotational peaks (Navaza, 2001), but both these techniques entail additional computational cost. Hence a more thorough but economical method of correlating electron densities could be beneficial. We note that combined six-dimensional rotational-translational Patterson correlation (PC) approaches have recently been successful in solving difficult MR cases without using phase information (Chang &

Lewis, 1997; Sheriff *et al.*, 1999). However, these approaches are based on point-wise evaluations of the PC in reciprocal space (Chang & Lewis, 1997; Kissinger *et al.*, 1999; Jamrog & Zhang, 2003) and hence appear not to be amenable to the type of correlation discussed here. Correlating electron densities may also be used to locate secondary structure fragments in density maps (Cowtan, 1998), and has been proposed as a novel way to search the Protein Data Bank (PDB) (Sussman *et al.*, 1998) for candidate molecular replacement models (Cowtan, 2001). Similarly, protein-protein docking algorithms often use a three-dimensional translational FFT to search for good docking orientations (Katchalski-Katzir *et al.*, 1992; Vakser, 1995). However, in each of these examples the translational FFTs must be repeated for a large number of rotational orientations in order to cover the remaining three degrees of freedom. Hence, more efficient correlation techniques would be useful in these areas as well.

The FFT may also be used to accelerate rotational searches. For example, the MR fast rotation function represents a Patterson map of self-vectors as an expansion of spherical Bessel and spherical harmonic functions and uses a two-dimensional FFT to orient a search model with respect to the observed diffraction data (Navaza, 1987). Vagin & Isupov (2001) adapted the fast rotation function to implement a phased rotation function (rotational density correlation) which allows a spherically averaged search model to be located in the unit cell using the phased translation function before solving its rotational orientation. However, this approach still requires rotational and translational correlations to be treated separately. More recently, Kovacs *et al.* (2003) employed an elegant Euler angle factorization in order to fit proteins into EM density maps using a five-dimensional

rotational FFT. However, their approach entails calculating certain residual overlap integrals which must be computed numerically.

In our own areas of interest, namely protein docking and shape comparison, we developed a six-dimensional spherical polar Fourier (SPF) correlation approach (Ritchie & Kemp, 2000) to try to address the limitations of the Cartesian FFT docking methods. In the SPF representation, each three-dimensional scalar property of interest is written as an expansion of real spherical harmonic plus orthonormal radial basis functions. This allows the correlation between a pair of proteins to be expressed naturally as a search over one translational and five rotational degrees of freedom. Because the rotational properties of the spherical harmonics have been described extensively elsewhere (see *e.g.* Rose, 1957; Biedenharn & Louck, 1981*a*), this article focuses on calculating translations for SPF expansions. Formerly, we calculated translations by numerical integration in the (r, θ) plane (Ritchie & Kemp, 2000). Here, we give analytic expressions for efficiently calculating high-order translation matrix elements for both harmonic oscillator (HO) and orthogonal Coulomb-type radial basis functions using spherical Bessel transforms. The HO functions, sometimes referred to as Gaussian-type orbitals (GTOs), have long been used in molecular orbital calculations because of the ease of calculating overlap integrals between pairs of such functions (Boys, 1950). The Coulomb-type functions, sometimes referred to as exponential-type orbitals (ETOs) (Guseinov, 2001, 2002), have an exponential factor in place of the Gaussian and correspond to orthogonalized Slater-type atomic orbitals (STOs) (Barnett & Coulson, 1951) and B-functions (Filter & Steinborn, 1978), or equivalently Bessel-type orbitals (BTOs) (Ozdogan *et al.*, 2005).

In what follows, the abbreviations GTO and ETO will be used to refer to the radial functions used here for consistency with the recent quantum mechanics literature. Despite much prior work, there is still considerable interest in developing efficient and stable methods of calculating integrals over GTOs (Arakane & Matsuoka, 1999; Chiu & Moharerrzadeh, 1999), and especially STOs (Filter & Steinborn, 1980; Weniger & Steinborn, 1983; Hierse & Oppeneer, 1994; Ozdogan *et al.*, 2005; Rico *et al.*, 2005). Nonetheless, to our knowledge, the analytic expressions presented here to translate ETO basis functions and to relate ETOs to BTOs are novel. We also describe non-orthogonal expansions of GTOs and ETOs which give at least a twofold speed-up compared with calculating translations in the orthogonal bases.

Our expressions have been tested for numerical accuracy up to order $N = 32$, which is sufficient for high-resolution protein–protein docking correlations (Ritchie, 2003). We find it is necessary to use an extended-precision arithmetic library to calculate high-order translation matrix elements analytically, although all subsequent calculations may be performed in ordinary double-precision arithmetic. However, because much of our docking algorithm has been described previously (Ritchie & Kemp, 2000), the calculations are illustrated here by superposing a pair of similar superantigen proteins. In

contrast to multi-dimensional FFT approaches, our correlation is calculated as a combinatorial search over pairs of rotated and translated coefficient vectors. With low-order polynomial expansions there is no advantage in using an FFT, although for high-order correlations our algorithm can be accelerated using a one-dimensional real fast Hartley transform (FHT) (Bracewell, 1999). In the example given, a good superposition is achieved using low-order expansions to $N = 6$, with a total calculation time of under 8 s for an exhaustive six-dimensional search on a 2 GHz Pentium Xeon processor. This corresponds to evaluating approximately 6×10^6 trial orientations s^{-1} . This demonstrates that the SPF correlation technique offers a novel and powerful approach for problems that involve calculating real-space six-dimensional rotational–translational correlations.

2. Methods

2.1. Polar Fourier expansions

In the SPF approach, each scalar property of interest, $A(\mathbf{r})$, is represented as an infinite expansion, truncated to order N as

$$A(\mathbf{r}) = \sum_{n=1}^N \sum_{l=0}^{n-1} \sum_{m=-l}^l a_{nlm} R_{nl}(r) y_{lm}(\theta, \phi), \quad (1)$$

where a_{nlm} are the expansion coefficients, $y_{lm}(\theta, \phi) = \vartheta_{|l|}(\theta) \varphi_m(\phi)$ are real normalized spherical harmonics, and $R_{nl}(r)$ are orthonormal GTO or ETO radial basis functions. Generally, we use GTOs to represent steric shape and the more diffuse ETOs to represent electrostatic properties. We follow the quantum chemistry convention in which the radial index n , or principal quantum number, counts from unity. Hence the highest harmonic order and highest polynomial power in any individual coordinate is $L = N - 1$. An expansion to order N involves $N(N + 1)(2N + 1)/6$ coefficients. These are calculated just once for each property as described previously (Ritchie & Kemp, 2000).

2.2. Overlap Integrals as translation matrix elements

First, we consider the correlation between a fixed ‘body’ A, or scalar function $A(\mathbf{r})$, and a moving body B, or function $B(\mathbf{r})$, under an active translation of B by $\mathbf{T} = (R, 0, 0)$ along the positive z axis:

$$C(\mathbf{T}) = \int A(\mathbf{r}) B(\mathbf{T}^{-1}\mathbf{r}) dV. \quad (2)$$

Substituting the expansions of $A(\mathbf{r})$ and $B(\mathbf{r})$ gives

$$C(\mathbf{T}) = \sum_{nlm} \sum_{n'l'm'} a_{nlm} b_{n'l'm'} \times \int R_{nl}(r) y_{lm}(\theta, \phi) R_{n'l'}(r') y_{l'm'}(\theta', \phi') dV, \quad (3)$$

where the shorthand notation \sum_{nlm} *etc.* is used to indicate summation over the subscript ranges given in equation (1), and where $\mathbf{r} = (r, \theta, \phi)$ and $\mathbf{r}' = \mathbf{r} - \mathbf{T} = (r', \theta', \phi')$. In this case, ϕ and ϕ' remain coincident, so the circular functions, $\varphi_m(\phi)$, may be integrated out and equation (3) reduces to a sum over two-

dimensional integrals in the (r, θ) plane. Because these integrals clearly depend only on the distance R , we write

$$C(R) = \sum_{nlm} \sum_{n'l'm'} a_{nlm} b_{n'l'm'} T_{nl,n'l'}^{(|m|)}(R) \delta_{mm'} \quad (4)$$

and interpret each $T_{nl,n'l'}^{(|m|)}(R)$ as a matrix element of the translation operator. For example, from equation (4) it can be seen that the two sums

$$b_{nlm}^R = \sum_{n'l'} T_{nl,n'l'}^{(|m|)}(R) b_{n'l'm} \quad (5)$$

and (after re-labeling the subscripts)

$$a_{nlm}^R = \sum_{n'l'} T_{n'l',nl}^{(|m|)}(R) a_{n'l'm} \quad (6)$$

represent a positive translation of the body B, or equivalently a negative translation of the body A, respectively. The translation matrices are obviously five-dimensional quantities. However, because they do not depend on the sign of m (see below), it is useful to consider each matrix as being composed of $\sum_{m=0}^{N-1} = N$ two-dimensional arrays, each indexed by $\sum_{nl} = N(N+1)/2$ possible values for each pair of nl subscripts. The matrix elements vanish trivially where $|m| > l$. The notation used here is intended to be consistent with the usual convention for the complex and real spherical harmonic rotation matrix elements, $D_{m'm}^{(l)}(\alpha, \beta, \gamma)$ and $R_{m'm}^{(l)}(\alpha, \beta, \gamma)$, respectively, in the sense that a positive z translation of the basis functions is expressed as

$$R_{nl}(r') y_{lm}(\theta', \phi) = \sum_{n'=1}^{\infty} \sum_{l'=0}^{n'-1} T_{n'l',nl}^{(|m|)}(R) R_{n'l'}(r) y_{l'm}(\theta, \phi). \quad (7)$$

2.3. Spherical Bessel transform method of calculation

If the spherical Bessel transform of $R_{nl}(r)$ is defined as

$$\tilde{R}_{nl}(\beta) = (2/\pi)^{1/2} \int_0^{\infty} R_{nl}(r) j_l(\beta r) r^2 dr, \quad (8)$$

where $j_l(z)$ is the spherical Bessel function, then the inverse transform is given by (Hochstadt, 1971):

$$R_{nl}(r) = (2/\pi)^{1/2} \int_0^{\infty} \tilde{R}_{nl}(\beta) j_l(\beta r) \beta^2 d\beta. \quad (9)$$

It is shown in Appendix A that the translation matrix elements for SPF basis functions may be calculated as a sum over one-dimensional inverse Bessel transforms

$$T_{n'l',nl}^{(|m|)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(|l'|, |m|)} \int_0^{\infty} \tilde{R}_{n'l'}(\beta) \tilde{R}_{nl}(\beta) j_k(\beta R) \beta^2 d\beta, \quad (10)$$

where the coefficients $A_k^{(|l'|, |m|)}$ are given by

$$A_k^{(|l'|, |m|)} = (-1)^{(k+l'-l)/2+m} (2k+1) [(2l+1)(2l'+1)]^{1/2} \times \begin{pmatrix} l'l'k \\ 000 \end{pmatrix} \begin{pmatrix} l'l'k \\ m\bar{m}0 \end{pmatrix}. \quad (11)$$

From the permutational symmetries of the second 3- j symbol, the right-hand side is independent of the sign of m , hence justifying the use of $|m|$ to label the matrix elements. Because the first 3- j symbol vanishes whenever $l+l'+k$ is odd, it can be seen that the non-vanishing coefficients are always real and that the summation in equation (10) need only be calculated for even increments: $k = |l-l'|, |l-l'|+2, \dots, l+l'$. By similar arguments (see Appendix A), it can also be shown that

$$T_{nl,n'l'}^{(|m|)}(R) = T_{n'l',nl}^{(|m|)}(-R) = (-1)^{l-l'} T_{n'l',nl}^{(|m|)}(R). \quad (12)$$

Consequently, nearly half of all matrix elements can be found by symmetry. Given that the original basis functions form a complete orthonormal set, it is straightforward to show that the translation matrices are also orthonormal in the sense that

$$\sum_{n'=1}^{\infty} \sum_{l'=0}^{n'-1} T_{n'l',nl}^{(|m|)}(R) T_{n'l',n''l''}^{(|m|)}(R) = \delta_{nn''} \delta_{ll''}. \quad (13)$$

Evaluating this equation provides a convenient way to verify the following calculations.

2.4. GTO translation matrix elements

The normalized GTO radial functions are given by (Biedenharn & Louck, 1981b)

$$R_n(r) = \left[\frac{2}{\lambda^{3/2} \pi^{1/2}} \frac{(n-l-1)!}{(1/2)_n} \right]^{1/2} \exp(-\rho^2/2) \rho^l L_{n-l-1}^{(l+1/2)}(\rho^2), \quad (14)$$

where $\rho^2 = r^2/\lambda$ with scale factor λ . For protein shape representations we set $\lambda = 20$. Throughout this article $(x)_n = x(x+1) \dots (x+n-1) = \Gamma(x+n)/\Gamma(x)$ denotes a rising factorial and the substitution $\Gamma(n+1/2) = \pi^{1/2} (1/2)_n$ is frequently employed for numerical accuracy. The generalized Laguerre polynomials, $L_k^{(\alpha)}(x)$, are defined by the binomial expansion (Erdelyi *et al.*, 1953a)

$$L_k^{(\alpha)}(x) = \sum_{j=0}^k \binom{k+\alpha}{k-j} \frac{(-x)^j}{j!}. \quad (15)$$

High-order polynomials may be calculated efficiently using the stable recursion

$$(k+1)L_{k+1}^{(\alpha)}(x) = (2k+\alpha+1-x)L_k^{(\alpha)}(x) - (k+\alpha)L_{k-1}^{(\alpha)}(x), \quad (16)$$

along with the identities $L_0^{(\alpha)}(x) = 1$ and $L_1^{(\alpha)}(x) = \alpha+1-x$. The GTO functions are eigenfunctions of the spherical Bessel transform (derived from Erdelyi *et al.*, 1953b, p. 42, equation 3 therein):

$$\tilde{R}_{nl}(\beta) = (-1)^{n-l-1} \left[\frac{2\lambda^{3/2} (n-l-1)!}{\pi^{1/2} (1/2)_n} \right]^{1/2} \times \exp(-x^2/2) x^l L_{n-l-1}^{(l+1/2)}(x^2), \quad (17)$$

where $x^2 = \lambda\beta^2$. Here, it is convenient to use equation (15) to expand equation (17) as a power series:

$$\tilde{R}_{nl}(\beta) = \left[\frac{4\lambda^{3/2}}{\pi^{1/2}} \right]^{1/2} \sum_j X_{nlj} \exp(-x^2/2)x^{2j+l} \quad (18)$$

where \sum_j serves as shorthand notation for $\sum_{j=0}^{n-l-1}$ and where the coefficients X_{nlj} are given by

$$X_{nlj} = \left[\frac{(n-l-1)!(1/2)_n}{2} \right]^{1/2} \frac{(-1)^{n-l-j-1}}{j!(n-l-j-1)!(1/2)_{l+j+1}}. \quad (19)$$

Substituting equation (18) twice into equation (10) and collecting coefficients of x^{2k} using

$$C_k^{(nl,n'l')} = \sum_j \sum_{j'} \delta_{k,j+j'} X_{nlj} X_{n'l'j'} \quad (20)$$

(where δ_{kj} is the Kronecker delta) gives for the GTO translation matrix elements:

$$T_{n'l',nl}^{(lm)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(l'|m)} \sum_{j=0}^{n-l+n'-l'-2} C_j^{(nl,n'l')} \times \frac{4}{\pi^{1/2}} \int_0^\infty x^{2j+l+l'} j_k(xR/\lambda^{1/2})x^2 dx. \quad (21)$$

Applying the relation (from Erdelyi *et al.*, 1953b, p. 30, equation 13 therein)

$$\frac{4}{\pi^{1/2}} \int_0^\infty \exp(-x^2)x^{2m+k} j_k(xy)x^2 dx = m! \exp(-y^2/4)(y^2/4)^{k/2} L_m^{(k+1/2)}(y^2/4), \quad (22)$$

then gives the final analytic result

$$T_{n'l',nl}^{(lm)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(l'|m)} \sum_{j=0}^{n-l+n'-l'-2} C_j^{(nl,n'l')} M! \times \exp(-R^2/4\lambda)(R^2/4\lambda)^{k/2} L_M^{(k+1/2)}(R^2/4\lambda), \quad (23)$$

where $M = j + (l + l' - k)/2$.

2.5. ETO translation matrix elements

The normalized ETO radial functions are given by (Biedenharn & Louck, 1981b)

$$S_{nl}(r) = \left[(2\Lambda)^3 \frac{(n-l-1)!}{(n+l+1)!} \right]^{1/2} \exp(-\rho/2)\rho^l L_{n-l-1}^{(2l+2)}(\rho), \quad (24)$$

where $\rho = 2\Lambda r$ with scale factor Λ . We set $\Lambda = 1/2$ for protein-protein electrostatic calculations (Ritchie & Kemp, 2000). Using an argument based on orthogonality, Keister & Polyzou (1997) recently proved that the spherical Bessel transform of these functions may be written in terms of the Jacobi polynomials, $P_k^{(\mu,\nu)}(t)$:

$$\tilde{S}_{nl}(\beta) = \frac{2}{(1/2)_n} \left[\frac{(n-l-1)!(n+l+1)!}{\pi\Lambda^3} \right]^{1/2} \times \frac{s^l}{(s^2+1)^{l+2}} P_{n-l-1}^{(l+3/2,l+1/2)}\left(\frac{s^2-1}{s^2+1}\right), \quad (25)$$

where $s = \beta/\Lambda$. Following a similar treatment to the GTO case, the shifted series expansion (Keister & Polyzou, 1997)

$$P_k^{(\mu,\nu)}(t) = \sum_{j=0}^k \frac{\Gamma(k+\mu+1)}{\Gamma(k+\mu+\nu+1)} \frac{(-1)^j \Gamma(k+j+\mu+\nu+1)}{j!(k-j)!\Gamma(j+\mu+1)} \left(\frac{1-t}{2}\right)^j \quad (26)$$

may be used to collect factors of $1/(s^2+1) = (1-t)/2$ to write equation (25) as a power series:

$$\tilde{S}_{nl}(\beta) = \left(\frac{2}{\pi\Lambda^3}\right)^{1/2} \sum_j Y_{nlj} \frac{s^l}{(s^2+1)^{l+j+2}}, \quad (27)$$

where

$$Y_{nlj} = \left[\frac{1(n-l-1)!}{2(n+l+1)!} \right]^{1/2} \frac{(-1)^j (2n+1)(n+l+j+1)!}{j!(n-l-j-1)!(1/2)_{l+j+2}}. \quad (28)$$

Substituting equation (27) twice into equation (10) and collecting coefficients of $1/(s^2+1)^k$ using

$$D_k^{(nl,n'l')} = \sum_j \sum_{j'} \delta_{k,j+j'} Y_{nlj} Y_{n'l'j'}, \quad (29)$$

gives for the ETO translation matrix elements:

$$U_{n'l',nl}^{(lm)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(l'|m)} \sum_{j=0}^{n-l+n'-l'-2} D_j^{(nl,n'l')} \times \frac{2}{\pi} \int_0^\infty \frac{s^{2M+k}}{(s^2+1)^{J+2}} j_k(s\Lambda R)s^2 ds, \quad (30)$$

where $M = (l + l' - k)/2$ and $J = j + l + l' + 2$. It is shown in Appendix B that the remaining integral may be calculated as

$$\frac{2}{\pi} \int_0^\infty \frac{s^{2M+k}}{(s^2+1)^{J+2}} j_k(s\Lambda R)s^2 ds = \sum_{q=0}^M \binom{M}{q} \frac{(-1)^{M+q}}{2^{J+1-q}(J+1-q)!} (\Lambda R)^k \hat{k}_{J-k-q+1/2}(\Lambda R), \quad (31)$$

where $\hat{k}_\sigma(z)$ is a reduced Bessel function of the second kind. For half-integral degree, these functions may be calculated using the recurrence relations (Weniger & Steinborn, 1983):

$$\hat{k}_{1/2}(z) = \exp(-z), \quad (32)$$

$$\hat{k}_{3/2}(z) = (1+z)\exp(-z), \quad (33)$$

and

$$\hat{k}_{n+3/2}(z) = (2n+1)\hat{k}_{n+1/2}(z) + z^2\hat{k}_{n-1/2}(z). \quad (34)$$

Thus, the ETO translation matrix elements may also be calculated analytically, although compared with the GTO basis an additional inner summation is necessary.

2.6. Non-orthogonal translation matrices

Translations of SPF expansions in both the GTO and ETO bases can be computed more economically by eliminating the inner summation on the subscript j in equations (23) and (30). This is equivalent to calculating overlap integrals that correspond to expansions of non-orthogonal radial basis functions. For example, substituting equation (18) into equation (10) and applying equation (22) directly gives the factorization

$$T_{n'l',nl}^{(lm)}(R) = \sum_j \sum_j X_{n'l'j} \bar{T}_{j'l',jl}^{(lm)}(R) X_{nlj}, \quad (35)$$

where each $\bar{T}_{j'l',jl}^{(lm)}(R)$ is an overlap integral in a non-orthogonal basis,

$$\begin{aligned} \bar{T}_{j'l',jl}^{(lm)}(R) &= \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|lm)} M! \exp(-R^2/4\lambda) \\ &\times (R^2/4\lambda)^{k/2} L_M^{(k+1/2)}(R^2/4\lambda), \end{aligned} \quad (36)$$

now with $M = j + j' + (l + l' - k)/2$. This corresponds to expanding $R_{nl}(r)$ as a sum of non-orthogonal functions, $\bar{R}_{nl}(r)$:

$$R_{nl}(r) = (-1)^{n-l-1} \sum_j X_{nlj} \bar{R}_{jl}(r). \quad (37)$$

It can be shown that these functions correspond to the non-orthogonal Laguerre–Gaussian basis proposed by Chiu & Moharerrzadeh (1999). With this factorization, translated expansion coefficients, a_{nlm}^R , in the original orthogonal basis may be calculated using the sequence

$$\bar{a}_{jlm} = \sum_n X_{nlj} a_{nlm}, \quad (38)$$

$$\bar{a}_{jlm}^R = \sum_{j'l'} \bar{T}_{j'l',jl}^{(lm)}(R) \bar{a}_{j'l'm}, \quad (39)$$

$$\bar{a}_{nlm}^R = \sum_j X_{nlj} \bar{a}_{jlm}^R. \quad (40)$$

In a similar manner, ETO translations may be calculated in a non-orthogonal basis by substituting equation (27) into equation (10) and collecting powers of $1/(s^2 + 1)$ directly to give

$$U_{n'l',nl}^{(lm)}(R) = \sum_j \sum_j Y_{n'l'j} \bar{U}_{j'l',jl}^{(lm)}(R) Y_{nlj}, \quad (41)$$

where the non-orthogonal matrix elements, $\bar{U}_{j'l',jl}^{(lm)}(R)$, are given by

$$\begin{aligned} \bar{U}_{j'l',jl}^{(lm)}(R) &= \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|lm)} \sum_{q=0}^M \binom{M}{q} \frac{(-1)^{M+q}}{2^{J+1-q}(J+1-q)!} \\ &\times (\Lambda R)^k \hat{k}_{J-k-q+1/2}(\Lambda R) \end{aligned} \quad (42)$$

with $M = (l + l' - k)/2$ and now $J = j + j' + l + l' + 2$.

Finally, using equation (31) with $M = 0$ and $J = l + j$, it is straightforward to apply the inverse Bessel transform to equation (27) to give

$$\begin{aligned} S_{nl}(r) &= \Lambda^{3/2} \sum_j Y_{nlj} \frac{1}{2^{l+j+1}(l+j+1)!} (\Lambda r)^j \hat{k}_{j+1/2}(\Lambda r) \\ &= \Lambda^{3/2} \sum_j Y_{nlj} B_{jl}(\Lambda r), \end{aligned} \quad (43)$$

where the $B_{jl}(\Lambda r)$ are the well known B-functions, or BTOs (Filter & Steinborn, 1978). In other words, equation (43) shows that ETOs correspond to orthogonalized BTOs.

2.7. Correlation algorithm and complexity analysis

To illustrate an application of our approach, we demonstrate the superposition of a pair of similar proteins, A and B, with each protein initially centred at the origin and represented as a single three-dimensional function [equation (1)], e.g. steric density or electrostatic charge. Letting $A(\mathbf{r})$ and $B(\mathbf{r})$ represent the pair of properties to be correlated, it is then convenient to express the correlation as

$$\begin{aligned} C(\beta_1, \gamma_1, R, \alpha_2, \beta_2, \gamma_2) &= \int [\hat{T}_z(-R) \hat{R}(0, \beta_1, \gamma_1) A(\mathbf{r})] \\ &\times [\hat{R}(\alpha_2, \beta_2, \gamma_2) B(\mathbf{r})] dV, \end{aligned} \quad (44)$$

where $\hat{R}(\alpha, \beta, \gamma)$ represents an Euler rotation operator, $\hat{R}(\alpha, \beta, \gamma) \equiv \hat{R}_z(\alpha) \hat{R}_y(\beta) \hat{R}_z(\gamma)$, and where $\hat{T}_z(-R)$ translates the rotated $A(\mathbf{r})$ by R along the negative z axis. We could have chosen instead to translate the rotated $B(\mathbf{r})$ along the positive z axis. We use an icosahedral tessellation of the sphere to generate near-regular (β, γ) rotational samples for each molecule (Ritchie & Kemp, 1999). If the proteins are similar, the translational part of the search will normally be small. In any case, we seek to find the maximum of the expansion

$$\begin{aligned} C(\beta_1, \gamma_1, R, \alpha_2, \beta_2, \gamma_2) &= \sum_{n'l'n'l'm'm''} T_{n'l',nl}^{(lm')}(R) R_{mm'}^{(l')}(0, \beta_1, \gamma_1) \\ &\times a_{n'l'n'l'm''}^{(l)}(\alpha_2, \beta_2, \gamma_2) b_{nlm''}. \end{aligned} \quad (45)$$

The real rotation matrix elements, $R_{mm'}^{(l)}(\alpha, \beta, \gamma)$, may be calculated efficiently using recursion formulae (Navaza, 1990; Ritchie & Kemp, 1999). However, the cost of computing equation (45) as it stands scales in the order of $O(N^7)$ operations for each trial orientation. A much more efficient strategy is to compute the sum in stages using precalculated rotation and translation matrix elements. For example, vectors of coefficients representing different three-dimensional (β_1, γ_1, R) orientations of molecule A may be calculated in $O(N^3) \times [O(N) + O(N^2)] = O(N^5)$ operations per vector (orientation) using

$$a_{nlm}^{\beta_1\gamma_1} = \sum_{m'} R_{mm'}^{(l)}(0, \beta_1, \gamma_1) a_{nlm'} \quad (46)$$

and

$$a_{nlm}^{R\beta_1\gamma_1} = \sum_{n'l'} T_{n'l',nl}^{(lm)}(-R) a_{n'l'm}^{\beta_1\gamma_1}. \quad (47)$$

Similarly, vectors of rotated instances of molecule B may be calculated in $O(N^3) \times O(N) = O(N^4)$ operations per orientation using

$$b_{nlm}^{\beta_2\gamma_2} = \sum_{m'} R_{mm'}^{(l)}(0, \beta_2, \gamma_2) b_{nlm'}. \quad (48)$$

The final degree of freedom is a twist rotation about the z axis, which could be applied to molecule B as

$$b_{nlm}^{\alpha_2\beta_2\gamma_2} = \sum_{m'} R_{mm'}^{(l)}(\alpha_2, 0, 0) b_{nlm'}^{\beta_2\gamma_2} \quad (49)$$

$$= \sum_{m'} b_{nlm'}^{\beta_2\gamma_2} \cos m' \alpha_2 + b_{nl\bar{m}'}^{\beta_2\gamma_2} \sin \bar{m}' \alpha_2. \quad (50)$$

However, it is more efficient to complete the correlation by iterating combinatorially over all pairs of the above-calculated A and B orientations in $O(N) \times O(N^2) = O(N^3)$ operations per iteration using

$$P_m^{\beta_1\gamma_1 R \beta_2\gamma_2} = \sum_{nl} a_{nlm}^{R\beta_1\gamma_1} b_{nlm}^{\beta_2\gamma_2} \quad (51)$$

and

$$Q_m^{\beta_1\gamma_1 R \beta_2\gamma_2} = \sum_{nl} a_{nlm}^{R\beta_1\gamma_1} b_{nl\bar{m}}^{\beta_2\gamma_2}, \quad (52)$$

followed by an inner iteration over α_2 using

$$C(\beta_1, \gamma_1, R, \alpha_2, \beta_2, \gamma_2) = \sum_m P_m^{\beta_1\gamma_1 R \beta_2\gamma_2} \cos m \alpha_2 + Q_m^{\beta_1\gamma_1 R \beta_2\gamma_2} \sin \bar{m} \alpha_2, \quad (53)$$

which clearly scales as $O(N)$ per trial orientation. The calculation of equation (53) for multiple angular samples, M , where $M \geq N$, may be performed in $O(M \log M)$ operations by using a one-dimensional FFT. However, because all quantities here are real, when $N \geq 16$, we use a single real FHT instead of the complex FFT to obtain around a 10% speed-up. Otherwise, it is faster to compute equation (53) explicitly for each value of α_2 in $O(MN)$ time. Thus, despite the relatively high cost of rotating [$O(N^4)$] and translating [$O(N^5)$] individual three-dimensional coefficient vectors, it can be seen that the above scheme reduces the cost of the combinatorial part of a six-dimensional correlation to just two inexpensive nested iterations, with the inner cycle costing only $O(N)$ operations or less per trial orientation.

3. Results

3.1. Verification and numerical precision of expressions

In order to verify our implementation, numerical results from the analytic translation matrix element expressions were compared with those from a two-dimensional numerical integration of equation (3) and a one-dimensional integration of equation (10). The two-dimensional integration was calculated over a regular 200×200 grid in the $[r, \cos(\theta)]$ plane with r ranging from 0 to 50 Å. The one-dimensional integration used 200 steps in β in a log-numerical scheme using

$$\int_0^\infty \tilde{R}_{nl}(\beta) \tilde{R}_{n'l'}(\beta) j_k(\beta R) \beta^2 d\beta \simeq \sum_{\beta_1=10^{-4}}^{\beta_{200}=5 \times 10^8} \tilde{R}_{nl}(\beta_i) \tilde{R}_{n'l'}(\beta_i) j_k(\beta_i R) \beta_i^2 \exp(\gamma_i) \Delta\gamma_i \quad (54)$$

with $\gamma_i = \ln \beta_i$ and $\Delta\gamma_i = \gamma_i - \gamma_{i-1}$. In both cases, increasing the number of integration steps or integration range gave a negligible effect on the numerical results. When calculating the angular coefficients $A_k^{(l'l|m|)}$ [equation (11)] for the one-dimensional integration, we found it became essential to use extended-precision arithmetic for $N \geq 16$ due to the many large factorials in Wigner's formula for the $3-j$ symbols (Biedenharn & Louck, 1981*a*). Hence an array of coefficients was precalculated using 256-bit arithmetic using the GMP arbitrary precision math library (<http://www.swox.com/gmp/>). We note that Tuzan *et al.* (1998) describe a procedure for calculating $3-j$ symbols using only single-precision arithmetic, but this requires significantly more programming than our current approach. Nonetheless, the remaining calculations do not require such high-precision arithmetic. For example, when using 128-bit arithmetic, the analytic calculations agreed with the numerical integration results to within three decimal digits or more for all matrix elements up to $N = 32$ for all translations up to $R = 9$ Å in the GTO basis, and up to $R = 90$ Å in the ETO basis. As an additional check, the orthonormality of both the GTO and ETO translation matrices was verified by evaluating the orthogonality formula equation (13) at a reduced distance of $\rho = 0.1$. This gave zero or unity, as appropriate, to within at least eight decimal digits up to $N = 20$, and to within at least two decimal digits for all translation matrices up to $N = 32$. Hence we are confident that our formulae and implementation are correct.

However, the above tests indicate that numerical precision can fall off significantly at high order. Table 1 presents further details on the overall computational cost and numerical precision for calculating GTO translations. Table 2 gives the corresponding results for the ETO basis. These tables show that log-numerical integration is the fastest method and that this gives better numerical precision than using 64-bit (*i.e.* Fortran or C 'double precision') arithmetic to evaluate the analytic expressions for high-order expansions. The tables also show that the numerical accuracy of the analytic calculations falls off markedly with high-order expansions unless high-precision arithmetic is used. This is presumably due to summing many terms of very different magnitudes. Overall, it can be seen that the ETO basis gives somewhat longer calculation times and slightly larger rounding errors than the corresponding GTO functions. The final columns of each table show that the non-orthogonal expansion method is at least twice as fast as using the orthogonal analytic formulae and gives equal or better precision. To achieve double-precision programming accuracy for calculations to $N = 32$, Tables 1 and 2 indicate that translation matrix elements should be calculated using 160-bit and 192-bit arithmetic for the GTO and ETO bases, respectively. Our correlation algorithm has the

Table 1

GTO basis vector translation times and accuracy.

This table lists the total number of distinct non-trivial matrix elements (integrals) for GTO translation matrices to order N , along with the time to calculate each matrix and translate a unit coefficient vector by 1 Å along the positive z axis. The remaining columns are labeled as 2D64: matrix elements calculated by two-dimensional numerical integration of equation (3) using a 200×200 grid in the (r, θ) plane, working in 64-bit arithmetic; 1D64: one-dimensional log-numerical integration of equation (54) using 200 steps in $\ln \beta$ and 64-bit arithmetic; A64: analytic integration [equation (23)] using 64-bit arithmetic, etc; N128: non-orthogonal analytic integration [equations (36) and (38)–(40)] in 128-bit arithmetic, etc. All times are in CPU s on a 2 GHz Pentium Xeon processor. Numbers in brackets give a measure of the overall precision of the result. For example, an entry of (16) means that the root mean squared (RMS) error in the translated vector is no more than $O(10^{-16})$. This is derived from the RMS deviation between a vector translated by a matrix calculated using the given number of bits of arithmetic and one for which 512-bit (effectively infinite precision) arithmetic was used.

N	Integrals	2D64	1D64	A64	A128	N128	A160	N160
6	504	0.3 (4)	0.00 (12)	0.03 (16)	0.03 (16)	0.01 (16)	0.03 (16)	0.01 (16)
9	2871	1.2 (3)	0.01 (12)	0.07 (12)	0.10 (16)	0.02 (16)	0.07 (16)	0.03 (16)
12	10374	3.6 (3)	0.04 (12)	0.21 (12)	0.19 (16)	0.10 (16)	0.20 (16)	0.09 (16)
16	38760	12.2 (3)	0.15 (11)	0.58 (9)	0.77 (16)	0.38 (16)	0.82 (16)	0.39 (16)
20	109802	37.2 (2)	0.45 (8)	1.90 (5)	2.54 (16)	1.19 (16)	2.73 (16)	1.26 (16)
25	315315	102.4 (2)	1.36 (5)	6.13 (1)	8.47 (16)	3.97 (16)	9.30 (16)	4.24 (16)
30	752928	233.1 (2)	3.65 (3)	17.41 (0)	23.38 (15)	10.83 (15)	25.78 (16)	11.72 (16)
32	1026256	315.1 (2)	5.39 (2)	24.49 (0)	33.49 (13)	15.57 (14)	36.82 (16)	16.74 (16)

Table 2

ETO basis vector translation times and accuracy.

This table lists the computation times in s and gives in brackets the number of digits of precision for calculating and using ETO translation matrices to order N , as described in the legend for Table 1. For the final two columns, higher precision arithmetic was used than in Table 1.

N	Integrals	2D64	1D64	A64	A128	N128	A192	N192
6	504	0.3 (3)	0.01 (7)	0.03 (14)	0.04 (16)	0.01 (16)	0.04 (16)	0.01 (16)
9	2871	1.4 (3)	0.02 (6)	0.07 (11)	0.08 (16)	0.03 (16)	0.08 (16)	0.04 (16)
12	10374	4.3 (2)	0.05 (6)	0.17 (6)	0.21 (16)	0.10 (16)	0.24 (16)	0.12 (16)
16	38760	14.8 (3)	0.17 (5)	0.61 (1)	0.79 (16)	0.41 (16)	0.98 (16)	0.47 (16)
20	109802	45.1 (2)	0.46 (5)	1.85 (0)	2.47 (14)	1.29 (14)	3.23 (16)	1.50 (16)
25	315315	123.3 (2)	1.40 (5)	6.23 (0)	8.43 (7)	4.20 (7)	10.87 (16)	4.92 (16)
30	752928	287.5 (2)	3.89 (5)	17.17 (0)	22.80 (0)	11.24 (0)	30.35 (16)	13.42 (16)
32	1026256	391.6 (1)	5.89 (4)	24.62 (0)	33.11 (0)	16.12 (0)	44.27 (15)	19.00 (15)

option to use either non-orthogonal expansions for ‘on the fly’ translations or to calculate and store orthogonal translation matrices. In the latter case, the matrix elements may be stored and subsequently used in ordinary double-precision arithmetic.

3.2. Protein superposition correlation example

Fig. 1(A) shows some GTO steric density representations of a pair of globular protein domains, namely the superantigens *Streptococcal* pyrogenic exotoxin A1 (PDB code 1B1Z) (Papageorgiou *et al.*, 1995) and the *Staphylococcus aureus* exotoxin SEC3 (PDB code 1JCK) (Fields *et al.*, 1996). These globular proteins have a relatively low sequence identity of 46%, but share a highly similar fold. Hence they may be superposed well by conventional least-squares fitting of conserved C_α coordinates. However, in this illustration the superposition was performed by maximizing the overlap between the respective GTO steric density expansions using correlations to $N = 6$. A near-identical superposition (not shown) was also achieved by correlating electrostatic charge densities in the ETO basis. Fig. 1(B) shows the corresponding backbone traces in the calculated superposition, along with the molecular surfaces from which the GTO expansions were derived. It can be seen that the high-order expansions capture the detailed shape of each protein remarkably well, although the low-order expansions still encode sufficient steric infor-

mation to allow a very good global superposition to be calculated. The superposition shown was calculated by searching over some 21×10^6 trial orientations generated from 162 (β, γ) icosahedral angular samples for each protein, 128 twist samples in α_2 , and 40 distance steps of 0.25 Å. After each cycle over the twist angle, the best overlap score was saved along with the corresponding coordinate values in a block of memory sufficient to store around 10^6 orientations. As the search progressed, this memory block was periodically sorted and culled. Hence there is essentially no practical limit on the number of trial orientations that may be evaluated and subsequently stored. In this example, the total memory usage never exceeded 15 Mbyte and the overall correlation time was under 8 s on a 2 GHz Pentium Xeon processor. This corresponds to a rate of 6×10^6 trial orientations s^{-1} , or just a few hundred arithmetic operations per orientation.

4. Discussion

Conceptually, our approach follows in the spirit of Crowther’s original spherical harmonic plus spherical Bessel method of correlating Patterson maps (Crowther, 1972). However, by replacing the Bessel functions with a set of orthonormal radial basis functions which are eigenfunctions of the spherical Bessel transform, our SPF approach extends this rotation-centric representation in a way which also allows the effect of

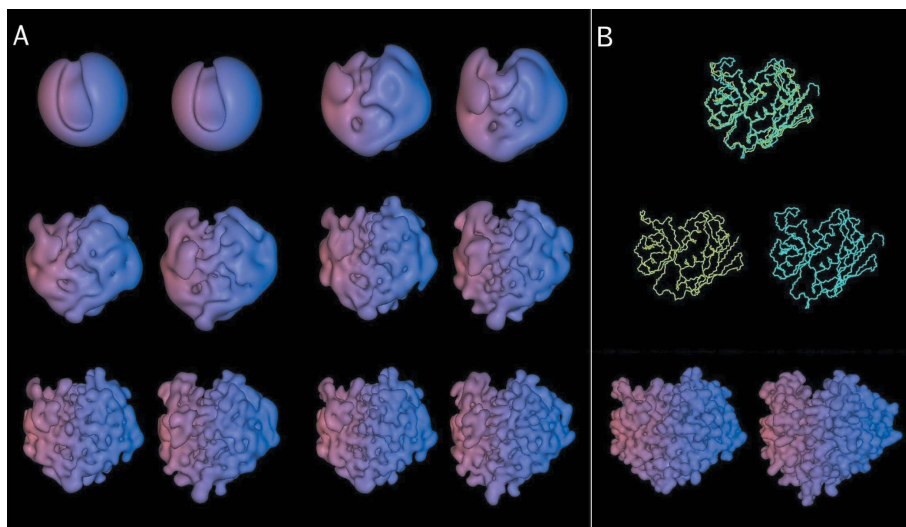


Figure 1
Illustration of the GTO shape representation and superposition of a pair of globular proteins, the superantigens SpeA and SEC3. (A) From top left to bottom right: the steric density functions of SpeA and SEC3 shown at expansion orders $N = 6, 12, 16, 20, 25$ and 30 . Each pair is in the same superposed orientation, separated horizontally for clarity, with SpeA on the left and SEC3 on the right. For visualization, each surface shape was contoured from the three-dimensional density function as described previously (Ritchie, 2003). Expansions to $N = 32$ are visually almost indistinguishable from the $N = 30$ expansions, and are not shown here. (B) The corresponding backbone traces of the superposition with SpeA in yellow and SEC3 in cyan (top), the separated backbones with SpeA on the left and SEC3 on the right (centre), and the original molecular surfaces from which the GTO density representations were derived (bottom).

translations to be calculated analytically. We have presented efficient spherical Bessel transform methods of calculating translation matrix elements for both GTO and ETO radial basis functions. Although it is well known that the Laguerre–Gaussian functions are eigenfunctions of the spherical Bessel transform, we believe the methods of calculation and the expressions presented here, especially for the ETO basis, are straightforward yet novel. However, it is necessary to call upon a significant body of special function theory to derive analytic forms for the SPF translation matrices, and the resulting expressions require considerably more programming to implement than the FFT. Nonetheless, the computational ‘pay-off’ that follows is that once an initial transform into the SPF basis has been calculated for each molecular property of interest, a full six-dimensional correlation search may be performed using only the initial expansion coefficients. Because all subsequent expressions are entirely real, no inverse transform is required. On the other hand, computing high-order matrix elements is relatively expensive. However, this overhead can be alleviated by calculating non-orthogonal expansions or by precalculating and storing orthogonal translation matrices for subsequent use.

There are also close parallels between our technique and the complex five-dimensional FFT-based EM density fitting approach of Kovacs *et al.* (2003). However, their approach is currently limited to relatively low-order ($L = 16$) spherical harmonic expansions in order to stay within the memory limits of contemporary 32-bit computers, and their use of discrete radial envelope functions requires the translational component of the correlation to be computed by two-dimensional

numerical integration (Kovacs *et al.*, 2003). Unlike the translation matrix elements used here, these integrals cannot be pre-computed as a one-off cost because they depend intrinsically on the shapes to be compared.

In general, FFT-based correlation algorithms use a power of two for both the polynomial order of the representation and the number of search increments in each degree of freedom in the correlation. For example, for angular searches, a rotational FFT step size of at least 64 is desirable in order to give reasonably small search increments (e.g. $360^\circ/64 \simeq 5.6^\circ$). Similarly, Cartesian FFT docking algorithms often use grids of 64^3 or 128^3 elements in order to accommodate all possible translations of one protein about the other with a sub-Å step size (Katchalski-Katzir *et al.*, 1992). Hence, unless a truncation technique is employed (Segal & Eisenstein, 2005), most FFT docking algorithms implicitly use a very high-order polynomial representation for each protein.

In contrast, in the SPF approach the polynomial order and search step size may easily be varied independently. In fact, because our polynomial order, N , is generally significantly less than the corresponding FFT order (say M) for any given coordinate, we typically have $O(MN) \lesssim O(M \ln M)$. Thus, in practice, it is not a disadvantage that our correlation algorithm uses an FFT (actually a real FHT) in at most just one of the search coordinates. Indeed, by calculating and reusing vectors of rotated and translated expansion coefficients, the complexity analysis given above shows that the inner iteration of the combinatorial part of a six-dimensional search scales as $O(N)$ or better per trial orientation.

Given the modest memory overhead of under 15 Mbyte of our low-order $N = 6$ superposition calculation (even our most exhaustive high-order $N = 32$ docking correlations use under 200 Mbyte), and considering that our superposition correlation time compares very favourably with the timings in Table I of Kovacs *et al.* (2003), it would seem that the SPF approach could offer significant computational advantages to both EM and MR applications. The straightforward analytic method of calculating ETO translation matrix elements presented here could also provide a useful new alternative to using BTO expansions to calculate STO overlap integrals in quantum mechanics. In principle, GTO and ETO expansions should give comparable results if suitable scale factors are chosen and if a sufficient number of terms are used. However, in practice, the ETOs decay much more gradually with distance than the GTOs. In our experience, the GTO basis works well for both superposing and docking globular protein domains, whereas the ETO basis is more appropriate for rapidly calculating electrostatic interactions (Ritchie & Kemp, 2000). However,

ETOs might also be a good choice for correlating very large low-resolution macromolecular structures. The results here indicate that either type could be used to calculate *e.g.* real-space electron density correlations. Although the example presented here shows that SPF expansions have good computational properties, it is not yet clear how the algorithm might perform when the translational component is large or when very high angular resolution is required. It would also be interesting to test the algorithm on a wider range of examples. Hence a more extensive study of the utility of the approach is planned. Developing a fast and robust six-dimensional correlation algorithm would be useful in several areas including *e.g.* molecular replacement, docking proteins into electron microscopy density maps, and searching the PDB for structural homologues.

5. Conclusion

Spherical Bessel transform formulae have been given for the analytic calculation of translation matrices for SPF expansions using GTO and ETO radial basis functions. To our knowledge, the formulae presented to translate ETO basis functions and to relate ETOs to BTOs are novel. It has been shown that translation matrices for both the GTO and ETO bases may be calculated accurately up to $N = 32$ using an extended-precision arithmetic library. The calculation may be accelerated by over a factor of two by using non-orthogonal translation matrix elements, which give equal or better precision. Although our SPF correlation algorithm uses an FFT in only one of the search coordinates, this is not a practical disadvantage for relatively low-resolution superposition and docking correlations. On the contrary, the SPF approach is very economical in both CPU time and computer memory. By ordering the sub-expressions according to computational cost, an exhaustive six-dimensional superposition correlation of a pair of protein shapes can be performed in a matter of seconds on a contemporary personal computer. This demonstrates the utility of the SPF approach. It is proposed that the techniques described here, which have been implemented in the protein docking and superposition program *Hex* (<http://www.csd.abdn.ac.uk/hex/>), may be of value in other fields, such as MR and EM, that need to calculate real-space six-dimensional correlations.

APPENDIX A

Derivation of equation (10)

The two-coordinate systems $\mathbf{r} = (r, \theta, \phi)$ and $\mathbf{r}' = \mathbf{r} - \mathbf{T} = (r', \theta', \phi')$ may be related functionally by multiplying the vector equation

$$\mathbf{r} = \mathbf{T} + \mathbf{r}' \quad (55)$$

by an arbitrary complex vector $i\mathbf{k}$ and by exponentiating each side to give

$$\exp(i\mathbf{k} \cdot \mathbf{r}) = \exp(i\mathbf{k} \cdot \mathbf{T}) \exp(i\mathbf{k} \cdot \mathbf{r}'). \quad (56)$$

Then, substituting $\mathbf{k} = (\beta, \Theta, \Phi)$, $\mathbf{r} = (r, \theta, \phi)$, $\mathbf{r}' = (r', \theta', \phi')$, and $\mathbf{T} = (R, \gamma, \delta)$ into Raleigh's plane wave equation (Bransden & Joachain, 1997)

$$\exp(i\mathbf{k} \cdot \mathbf{r}) = 4\pi \sum_{l=0}^{\infty} \sum_{m=-l}^l i^l j_l(\beta r) Y_{lm}(\Theta, \Phi)^* Y_{lm}(\theta, \phi), \quad (57)$$

gives

$$\begin{aligned} & \sum_{pq} i^p j_p(\beta r) Y_{pq}(\Theta, \Phi)^* Y_{pq}(\theta, \phi) \\ &= 4\pi \sum_{kj} \sum_{st} i^{k+s} j_k(\beta R) Y_{kj}(\Theta, \Phi)^* Y_{kj}(\gamma, \delta) j_s(\beta r') \\ & \quad \times Y_{st}(\Theta, \Phi)^* Y_{st}(\theta', \phi'), \end{aligned} \quad (58)$$

where $Y_{pq}(\theta, \phi)$ *etc.* are complex spherical harmonics. Here, each summation has an infinite range, subject to $|q| < p$, *etc.* Using the identity $Y_{st}(\Theta, \Phi)^* = (-1)^t Y_{s-t}(\Theta, \Phi)$, multiplying both sides by $Y_{lm}(\Theta, \Phi)$, and integrating over (Θ, Φ) gives

$$\begin{aligned} & j_l(\beta r) Y_{lm}(\theta, \phi) \\ &= 4\pi \sum_{kj} \sum_{st} i^{k+s-l} j_k(\beta R) Y_{kj}(\gamma, \delta) j_s(\beta r') Y_{st}(\theta', \phi') (-1)^t \\ & \quad \times \int Y_{lm}(\Theta, \Phi) Y_{s-t}(\Theta, \Phi) Y_{kj}(\Theta, \Phi)^* d\Omega. \end{aligned} \quad (59)$$

Substituting the standard formula for Gaunt's integral on the right gives

$$\begin{aligned} & j_l(\beta r) Y_{lm}(\theta, \phi) \\ &= 4\pi \sum_{kj} \sum_{st} i^{k+s-l} j_k(\beta R) Y_{kj}(\gamma, \delta) j_s(\beta r') Y_{st}(\theta', \phi') (-1)^{t+j} \\ & \quad \times \left[\frac{(2l+1)(2s+1)(2k+1)}{4\pi} \right]^{1/2} \begin{pmatrix} l & s & k \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} l & s & k \\ m & \bar{t} & \bar{j} \end{pmatrix}. \end{aligned} \quad (60)$$

Now, the first 3- j symbol vanishes when $l + s + k$ is odd, so the phase factor for the surviving terms is always real. Furthermore, the second 3- j symbol vanishes unless $m + \bar{t} + \bar{j} = 0$. Hence the expression may be reduced to a triple infinite sum by substituting $j = m - t$:

$$\begin{aligned} & j_l(\beta r) Y_{lm}(\theta, \phi) \\ &= 4\pi \sum_k \sum_{st} (-1)^{(k+s-l)/2} j_k(\beta R) Y_{k,m-t}(\gamma, \delta) j_s(\beta r') Y_{st}(\theta', \phi') (-1)^{m-t} \\ & \quad \times \left[\frac{(2l+1)(2s+1)(2k+1)}{4\pi} \right]^{1/2} \begin{pmatrix} l & s & k \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} l & s & k \\ m & \bar{t} & t-m \end{pmatrix}. \end{aligned} \quad (61)$$

When the translation is only in the z direction, $\delta = 0$ and either $\gamma = 0$ or $\gamma = \pi$, and $\phi' = \phi$, which entails $t = m$. This allows the summation over t to be eliminated, and the expression becomes valid for real as well as complex spherical harmonics. For a positive z translation, the harmonic $Y_{k,0}(\gamma = 0, 0)$ reduces to

$$Y_{k,0}(0, 0) = \left[\frac{2k+1}{4\pi} \right]^{1/2}. \quad (62)$$

Then, collecting the angular factors, relabelling $s \rightarrow l'$, and restricting the range of k according to the triangle rule for the 3- j symbols, gives the plane wave addition theorem:

$$j_l(\beta r)Y_{lm}(\theta, \phi) = \sum_{l'} \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|ml)} j_k(\beta R) j_{l'}(\beta r') Y_{l'm}(\theta', \phi), \quad (63)$$

where the angular coefficient $A_k^{(l'l|ml)}$ is given by

$$A_k^{(l'l|ml)} = (-1)^{(k+l'-l)/2+m} (2k+1) [(2l+1)(2l'+1)]^{1/2} \times \begin{pmatrix} l'l'k \\ 000 \end{pmatrix} \begin{pmatrix} l'l'k \\ m\bar{m}0 \end{pmatrix}. \quad (64)$$

For a negative translation, an additional factor of $(-1)^{l-l'}$ appears in the above expression because $Y_{k,0}(\gamma = \pi, 0) = (-1)^k Y_{k,0}(0, 0)$ and because $k+l'-l$ is even.

Now, multiplying both sides of equation (63) by $(2/\pi)^{1/2} \tilde{R}_{nl}(\beta) \beta^2$ and integrating over β [*i.e.* applying the inverse Bessel transform, equation (9)] gives

$$R_{nl}(r)Y_{lm}(\theta, \phi) = (2/\pi)^{1/2} \sum_{l'} \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|ml)} \times \int_0^\infty \tilde{R}_{nl}(\beta) j_k(\beta R) j_{l'}(\beta r') \beta^2 d\beta Y_{l'm}(\theta', \phi). \quad (65)$$

Then, multiplying each side by $R_{n'l'}(r')Y_{j'm'}(\theta', \phi)$ and integrating over all space in the corresponding variables gives

$$T_{n'l',nl}^{(l|ml)}(R) = (2/\pi)^{1/2} \sum_{l'} \delta_{j'l'} \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|ml)} \times \int_0^\infty \int_0^\infty R_{n'l'}(r') j_{l'}(\beta r') \tilde{R}_{nl}(\beta) j_k(\beta R) \beta^2 d\beta r'^2 dr'. \quad (66)$$

Finally, recognizing the integral in r' as the spherical Bessel transform of $R_{n'l'}(r')$, the result reduces to

$$T_{n'l',nl}^{(l|ml)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(l'l|ml)} \int_0^\infty \tilde{R}_{n'l'}(\beta) \tilde{R}_{nl}(\beta) j_k(\beta R) \beta^2 d\beta. \quad (67)$$

This generalizes the expression given by Danos & Maximon (1965) for translating multipole expansions to the more general case for arbitrary orthonormal radial functions, $R_{nl}(r)$.

APPENDIX B
Derivation of equation (31)

Following Weniger & Steinborn (1983), but correcting a mistake in their working [here, equation (75) onwards], a closed-form expression for the integral

$$I = \int_0^\infty \frac{x^{2m+k}}{(x^2+1)^{n+2}} j_k(xy) x^2 dx \quad (68)$$

may be obtained with the help of the basic relation (Erdelyi *et al.*, 1953*b*, p. 24, equation 20 therein):

$$\int_0^\infty \frac{x^{\nu+1}}{(x^2+a^2)^{\mu+1}} J_\nu(xy) dx = \frac{a^{\nu-\mu} y^\mu}{2^\mu \Gamma(\mu+1)} K_{\nu-\mu}(ay), \quad (69)$$

where $J_\nu(z)$ is a general Bessel function of the first kind, and $K_\sigma(z)$ is a modified Bessel function of the second kind. When the degree σ is half-integral, which is the case here, $K_\sigma(z)$ has a closed form:

$$K_{n+1/2}(z) = (\pi/2z)^{1/2} \exp(-z) \sum_{m=0}^n \left(\frac{1}{2z}\right)^m \frac{(n+m)!}{m!(n-m)!}. \quad (70)$$

However, this function has a singularity at the origin, so it is useful to use instead the reduced Bessel function

$$\hat{k}_{n+1/2}(z) = (2z/\pi)^{1/2} z^n K_{n+1/2}(z), \quad (71)$$

which is well behaved for all z . Equation (69) can then be cast in the desired form using the standard relation (Hochstadt, 1971):

$$(1/z \partial/\partial z)^m z^\sigma J_\sigma(tz) = t^m z^{\sigma-m} J_{\sigma-m}(tz). \quad (72)$$

Hence putting $\sigma = \nu + m$ and applying $(1/y \partial/\partial y)^m y^{\nu+m}$ to each side of equation (69) gives

$$\int_0^\infty \frac{x^{\nu+2m+1}}{(x^2+a^2)^{\mu+1}} J_\nu(xy) dx = \frac{a^{\nu+m-\mu}}{y^\nu 2^\mu \Gamma(\mu+1)} (1/y \partial/\partial y)^m y^{\nu+m+\mu} K_{\nu+m-\mu}(ay), \quad (73)$$

or in terms of the reduced Bessel function [equation (71)]

$$\int_0^\infty \frac{x^{\nu+2m+1}}{(x^2+a^2)^{\mu+1}} J_\nu(xy) dx = (\pi/2)^{1/2} \frac{1}{a^{2\mu} y^\nu 2^\mu \Gamma(\mu+1)} (1/y \partial/\partial y)^m (ay)^{2\mu} \hat{k}_{\nu+m-\mu}(ay). \quad (74)$$

It can be shown (Weniger & Steinborn, 1983) that

$$(1/z \partial/\partial z)^m z^\lambda \hat{k}_\sigma(z) = (-1)^m \sum_{q=0}^m \binom{m}{q} 2^q (-\lambda/2)_q z^{\lambda-2q} \hat{k}_{\sigma-m+q}(z). \quad (75)$$

Hence setting $\lambda = 2\mu$ and substituting equation (75) into equation (74) gives

$$\int_0^\infty \frac{x^{\nu+2m+1}}{(x^2+a^2)^{\mu+1}} J_\nu(xy) dx = (\pi/2)^{1/2} \frac{a^{2m-2\mu} (-1)^m}{2^\mu \Gamma(\mu+1)} \sum_{q=0}^m \binom{m}{q} 2^q (-\mu)_q y^{2\mu-2q-\nu} \hat{k}_{\nu-\mu+q}(ay). \quad (76)$$

Here, μ is an integer so the rising factorial may be recast as

$$(-\mu)_q = (-1)^q \mu! / (\mu - q)!. \quad (77)$$

We also have $\mu > \nu + 2m$ and so to keep the degree of the Bessel function positive it is convenient to use the identity

$$\hat{k}_{-\sigma}(z) = z^{-2\sigma} \hat{k}_{\sigma}(z). \quad (78)$$

Some further working then gives

$$\begin{aligned} & \int_0^{\infty} \frac{x^{\nu+2m+1}}{(x^2 + a^2)^{\mu+1}} J_{\nu}(xy) dx \\ &= (\pi/2)^{1/2} \frac{a^{2\nu+2m-2\mu}}{2^{\mu}} \sum_{q=0}^m \binom{m}{q} \frac{(-1)^{m+q} 2^q}{(\mu - q)!} y^{\nu} \hat{k}_{\mu-\nu-q}(ay). \end{aligned} \quad (79)$$

Finally, putting $\mu = n + 1$, $\nu = k + 1/2$, $a = 1$, and replacing $J_{k+1/2}(xy)$ by the corresponding spherical Bessel function gives

$$\begin{aligned} & \int_0^{\infty} \frac{x^{2m+k}}{(x^2 + 1)^{n+2}} j_k(xy) x^2 dx \\ &= \frac{\pi}{2} \sum_{q=0}^m \binom{m}{q} \frac{(-1)^{m+q}}{2^{n+1-q}(n+1-q)!} y^k \hat{k}_{n-k-q+1/2}(y). \end{aligned} \quad (80)$$

Part of this work was funded by the BBSRC (Grant Ref. 1/B10454).

References

Arakane, F. & Matsuoka, O. (1999). *Int. J. Quant. Chem.* **66**, 273–279.
 Barnett, M. P. & Coulson, C. A. (1951). *Philos. Trans. R. Soc. London A*, **243**, 221–249.
 Biedenharn, L. C. & Louck, J. C. (1981a). *Angular Momentum in Quantum Physics*. Reading, MA: Addison-Wesley.
 Biedenharn, L. C. & Louck, J. C. (1981b). *The Racah–Wigner Algebra in Quantum Theory*. Reading, MA: Addison-Wesley.
 Boys, S. F. (1950). *Proc. R. Soc. London A*, **200**, 542–554.
 Bracewell, R. N. (1999). *The Fourier Transform and its Applications*, 3rd ed. New York: McGraw-Hill.
 Bransden, B. H. & Joachain, C. J. (1997). *Introduction to Quantum Mechanics*. Harlow, UK: Addison Wesley Longman.
 Chang, G. & Lewis, M. (1997). *Acta Cryst.* **D53**, 279–289.
 Chiu, L. Y. C. & Moharerrzadeh, M. (1999). *Int. J. Quant. Chem.* **73**, 265–273.
 Cowtan, K. (1998). *Acta Cryst.* **D54**, 750–756.
 Cowtan, K. (2001). *Acta Cryst.* **D57**, 1435–1444.
 Crowther, R. A. (1972). *The Molecular Replacement Method*, edited by M. G. Rossmann, pp. 173–178. New York: Gordon and Breach.
 Danos, M. & Maximon, L. C. (1965). *J. Math. Phys.* **6**, 766–778.
 Erdelyi, A., Magnus, W., Oberhettinger, F. & Tricomi, F. G. (1953a). *Higher Transcendental Functions*, Vol 2. New York: McGraw-Hill.

Erdelyi, A., Magnus, W., Oberhettinger, F. & Tricomi, F. G. (1953b). *Tables of Integral Transforms*, Vol. 2. New York: McGraw-Hill.
 Fields, B. A., Malchiodi, E. L., Li, H., Ysern, X., Stauffacher, C. V., Schlievert, P. M., Karjalainen, K. & Mariuzza, R. A. (1996). *Nature (London)*, **384**, 188–192.
 Filter, E. & Steinborn, O. (1978). *Phys. Rev. A*, **18**, 1–11.
 Filter, E. & Steinborn, O. (1980). *J. Math. Phys.* **21**, 2725–2736.
 Frank, J. (2002). *Ann. Rev. Biophys. Biomol. Struct.* **31**, 303–319.
 Fujinaga, M. & Read, R. J. (1987). *J. Appl. Cryst.* **20**, 517–521.
 Guseinov, I. I. (2001). *Int. J. Quant. Chem.* **81**, 126–129.
 Guseinov, I. I. (2002). *Int. J. Quant. Chem.* **90**, 114–118.
 Hierse, W. & Oppeneer, P. M. (1994). *Int. J. Quant. Chem.* **52**, 1249–1265.
 Hochstadt, H. (1971). *The Functions of Mathematical Physics*. New York: Wiley.
 Jamrog, D. C., Zhang, Y. & Philips, G. N. Jr (2003). *Acta Cryst.* **D59**, 304–314.
 Katchalski-Katzir, E., Shariv, I., Eisenstein, M., Friesem, A. A. & Aflalo, C. (1992). *Proc. Natl Acad. Sci.* **89**, 2195–2199.
 Keister, B. D. & Polyzou, W. N. (1997). *J. Comput. Phys.* **134**, 231–235.
 Kissinger, C. R., Gelhaar, D. K. & Fogel, D. B. (1999). *Acta Cryst.* **D55**, 484–491.
 Kovacs, J. A., Chacon, P., Cong, Y., Metwally, E. & Wriggers, W. (2003). *Acta Cryst.* **D59**, 1371–1376.
 Navaza, J. (1987). *Acta Cryst.* **A43**, 645–653.
 Navaza, J. (1990). *Acta Cryst.* **A46**, 619–620.
 Navaza, J. (2001). *Acta Cryst.* **D57**, 1367–1372.
 Ozdogan, T., Orbay, M. & Degirmenci, S. (2005). *J. Math. Chem.* **37**, 27–35.
 Papageorgiou, A. C., Collins, C. M., Gutman, D. M., Kline, J. B., O’Brien, S. M., Tranter, H. S. & Acharya, K. R. (1995). *EMBO J.* **18**, 9–21.
 Read, R. J. & Schierbeek, A. J. (1988). *J. Appl. Cryst.* **21**, 490–495.
 Rico, J. F., López, R., Ema, I. & Ramírez, G. (2005). *J. Comput. Chem.* **26**, 846–855.
 Ritchie, D. W. (2003). *Proteins Struct. Func. Genet.* **52**, 98–106.
 Ritchie, D. W. & Kemp, G. J. L. (1999). *J. Comput. Chem.* **20**, 383–395.
 Ritchie, D. W. & Kemp, G. J. L. (2000). *Proteins Struct. Func. Genet.* **39**, 178–194.
 Rose, M. E. (1957). *Elementary Theory of Angular Momentum*. New York: Wiley.
 Roseman, A. M. (2000). *Acta Cryst.* **D56**, 1332–1340.
 Rossmann, M. G. (1990). *Acta Cryst.* **A46**, 73–82.
 Rossmann, M. G. (2000). *Acta Cryst.* **D56**, 1341–1349.
 Rossmann, M. G. (2001). *Acta Cryst.* **D57**, 1360–1366.
 Segal, D. & Eisenstein, M. (2005). *Proteins Struct. Func. Bioinf.* **59**, 580–591.
 Sheriff, S., Klei, H. E. & Davis, M. E. (1999). *J. Appl. Cryst.* **32**, 98–101.
 Sussman, J. L., Lin, D., Jiang, J., Manning, N. O., Prilusky, J., Ritter, O. & Abola, E. E. (1998). *Acta Cryst.* **D54**, 1078–1084.
 Tuzan, R. E., Burkhardt, P. & Secrest, D. (1998). *Comput. Phys. Commun.* **112**, 112–148.
 Vagin, A. A. & Isupov, M. N. (2001). *Acta Cryst.* **D57**, 1451–1456.
 Vakser, I. A. (1995). *Protein Eng.* **8**, 371–377.
 Weniger, E. J. & Steinborn, E. O. (1983). *J. Chem. Phys.* **78**, 6121–6132.