Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Physics Communications 165 (2005) 10–14

Computer Physics Communications

www.elsevier.com/locate/cpc

# Effective computation of matrix elements between polynomial basis functions

Igor N. Kozin [a], Jonathan Tennyson [b], Mark M. Law [a,*]

[a] *Department of Chemistry, University of Aberdeen, Meston Walk, Aberdeen AB24 3UE, UK*
[b] *Department of Physics and Astronomy, University College London, London WC1E 6BT, UK*

Received 10 May 2003; accepted 9 December 2003

---

**Abstract**

Two methods of evaluating matrix elements of a function in a polynomial basis are considered: the expansion method, where the function is expanded in the basis and the integrals are evaluated analytically, and the numerical method, where the integration is performed directly using numerical quadrature. A reduced grid is proposed for the latter which makes use of the symmetry of the basis. Comparison of the two methods is presented in the context of evaluation of matrix elements in a non-direct product basis. If high accuracy of all matrix elements is required then the expansion method is the best choice. If however the accuracy of high order matrix elements is not important (as in variational ro-vibrational calculations where one is typically interested only in the lowest eigenstates), then the method based on the reduced grid offers sufficient accuracy and is much quicker than the expansion method.

---

In many physical applications the problem of effective evaluation of integrals of a multi-dimensional function often arises. For example, the typical situation in the calculation of molecular rotation-vibration spectra is evaluation of matrix elements, which are integrals of the product of two general polynomial functions (basis functions) and an arbitrary function (molecular potential). Depending on the dimensionality and the number of matrix elements such calculations are often computationally very demanding. It is often the case that the properties of the polynomial functions are well known and there are analytical formulae for the integrals of these functions. Ref. [1] considered the case when the basis functions are a direct product of standard polynomials and advocated the expansion method, i.e. expansion of the potential in the basis and subsequent evaluation of the integrals analytically. In this paper, we propose a method of numerical integration on a reduced grid which makes use of the symmetry properties of the basis functions. Considering an example of a non-direct product basis we argue that our method

---

[*] Corresponding author.
*E-mail address:* M.M.Law@abdn.ac.uk (M.M. Law).

doi:10.1016/j.cpc.2003.12.007

of direct numerical integration can be very competitive to the expansion method and may even be the preferable choice.

Our starting point is a model for floppy four-atomic molecules as implemented in the new program WAVR4 [2]. The general description of the approach can be found elsewhere [3–6] and more details about the implementation in Refs. [2,7]. Of the six internal coordinates, the three radial coordinates are treated using the discrete variable representation (DVR) [8]. The DVR approximation for the potential reduces six-dimensional integrals to three-dimensional. The other three coordinates are the angles $(\theta_1, \theta_2, \varphi)$ which are represented by a non-direct product basis

$$|\, j\, l\, k\, J\, K\rangle = \bar{P}_j^{|k-K|}(\theta_1)\,\bar{P}_l^k(\theta_2)\,\frac{e^{ik\varphi}}{\sqrt{2\pi}}|J\, K\rangle, \tag{1}$$

where $J$ and $K$ are the usual rotational quantum numbers, $j$ and $l$ are angular momenta associated with internal coordinates, $k$ is projection of $l$ on the quantization axis, $\bar{P}_l^k$ are normalized associated Legendre functions, and $|J\, K\rangle$ are symmetric top eigenfunctions. All the quantum numbers take only non-negative integer values.

The most straightforward approach to evaluate integrals is to use numerical integration and the best way to do it is by using a Gaussian quadrature [9,10]. The idea is to replace the integral of a function by the sum of its functional values multiplied by weighting coefficients. The grid points are choosen so that they are the roots of a certain type of polynomial of order $N$. Then the quadrature is exact for all polynomials up to degree $(2N-1)$. If we take, for instance, Legendre type polynomials $P_l(x)$ up to order $l_{\max}$, we need $N = l_{\max} + 1$ or more quadrature points to compute the integrals

$$\int_{-1}^{1} P_{l_1}(x)\, f(x)\, P_{l_2}(x)\, dx \approx \sum_{i=1}^{N} P_{l_1}(x_i)\, f(x_i)\, P_{l_2}(x_i). \tag{2}$$

For the basis functions given in Eq. (1) we can use an equidistant (Fourier) grid for $\exp(ik\varphi)$ functions (in $\varphi$) and ordinary Legendre quadrature for associated Legendre functions (in $\theta_1$ and $\theta_2$). The advantage of the quadrature based on Legendre polynomials is that the same grid can be used throughout, that is for all $k$ of the associated Legendre functions. However if arbitrary $k$ is allowed then the associated Legendre functions require double the number of quadrature points which otherwise would be sufficient for Legendre polynomials of the same order. So if we characterize the size of our basis using $(j_{\max}, l_{\max}, k_{\max})$ we can define a *minimal* grid size $(2j_{\max} + 1) \times (2l_{\max} + 1) \times (2k_{\max} + 1)$, i.e. the minimal grid on which our basis is still orthogonal. Evaluation of all matrix elements using multi-dimensional quadrature can be a major bottle-neck in molecular rotation-vibration calculations.

One way to reduce the computational cost is to use an expansion of the arbitrary function involved in the integration [1]. In our case we can use the following expansion for the molecular potential

$$V(\theta_1, \theta_2, \varphi) = \sum_{jlk} a_{jlk}\,\bar{P}_j^k(\theta_1)\,\bar{P}_l^k(\theta_2)\,\frac{e^{ik\varphi}}{\sqrt{2\pi}}. \tag{3}$$

Because the potential function of a molecule is rotationally invariant, only $J = K = 0$ basis functions are required in the expanded potential. Ref. [11] provides an elegant way of expressing the matrix elements of the expanded potential (3) through integrals of products of spherical harmonics. The latter can be evaluated analytically through $3j$-symbols. Here, we take a shortcut and simply define

$$\int \bar{P}_{l_1}^{k_1}(x)\,\bar{P}_{l_2}^{k_2}(x)\,\bar{P}_{l_3}^{k_3}(x)\, dx = A(l_1, l_2, l_3; k_1, k_2, k_3), \tag{4}$$

where

$$\begin{aligned} &A(l_1, l_2, l_3; k_1, k_2, k_3) \\ &\quad = (-1)^{(k_1+k_2+k_3)/2}\sqrt{(2l_1+1)(l_2+1/2)(2l_3+1)}\begin{pmatrix} l_1 & l_2 & l_3 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} l_1 & l_2 & l_3 \\ m_1 & m_2 & m_3 \end{pmatrix}. \end{aligned} \tag{5}$$

Since $k_i \geqslant 0$ by definition, the $m_i$ are chosen so that $|m_i| = k_i$ and the $3j$-symbol rule holds: $m_1 + m_2 + m_3 = 0$. A second $3j$-symbol rule must be satisfied too: $|l_1 - l_3| \leqslant l_2 \leqslant l_1 + l_3$.

Then the matrix elements of the potential function are

$$\begin{aligned} &\langle j_1 l_1 k_1 \, J \, K | V | j_3 l_3 k_3 \, J \, K \rangle \\ &\quad = \frac{1}{\sqrt{2\pi}} \sum_{j_2 l_2 k_2} a_{j_2 l_2 k_2} A\big(j_1, j_2, j_3; |K - k_1|, k_2, |K - k_3|\big) A(l_1, l_2, l_3; k_1, k_2, k_3). \end{aligned} \tag{6}$$

The $3j$-symbol rules provide the upper limit for the expansion size: twice the maximum for each quantum number used in the basis. However the sum in Eq. (6) is smaller than the sum over the minimal grid for the same basis not only because of the non-direct product nature of our basis but also because many terms in Eq. (6) are zero due to $3j$-symbol rules (see Ref. [1] for more details).

In spite of the attractions of the expansion method we have found it valuable to reconsider direct numerical integration. It turns out we can do much better if we make use of properties of our basis. It is frequently the case that the integration interval is symmetric and the basis functions can be symmetric or anti-symmetric on the interval. Consider again the integral (2). The function $f(x)$ may be expressed as a sum of symmetric $f_s(x) = f_s(-x)$ and anti-symmetric $f_a(x) = -f_a(-x)$ parts. If the product $P_{l_1}(x)P_{l_2}(x)$ is symmetric we need to integrate only the symmetric part of $f(x)$ because the anti-symmetric part gives zero. This allows us to sum only over positive quadrature points and then double the sum. The analogous argument applies if $P_{l_1}(x)P_{l_2}(x)$ is anti-symmetric. Assuming $N$ is even we can write

$$\int_{-1}^{1} P_{l_1}(x) f(x) P_{l_2}(x)\,\mathrm{d}x \approx \begin{cases} 2\sum_{i=1}^{N/2} P_{l_1}(x_i) f_s(x_i) P_{l_2}(x_i) & \text{if } l_1 + l_2 \text{ is even,} \\ 2\sum_{i=1}^{N/2} P_{l_1}(x_i) f_a(x_i) P_{l_2}(x_i) & \text{if } l_1 + l_2 \text{ is odd.} \end{cases} \tag{7}$$

Therefore the knowledge of the symmetry properties of our basis can be used to reduce the computational cost.

Note, that the symmetry properties of the basis may have nothing to do with the symmetry of the molecule in question. Even if the symmetry $x \to -x$ above is absent in the molecular system one can still get a twofold computational gain due to the decomposition of the function to be integrated. If symmetry is present in the molecular system then $f_a(x) = 0$ and the whole problem splits in two: one involving only a symmetric basis and another involving an anti-symmetric basis. In this case, it is usual to use the symmetry of coordinate space and reduce the number of quadrature points. This reduction can be viewed as a sub-case of the more general case of a system without symmetry and therefore we suggest always implementing the more general algorithm described above.

There are more elaborate versions of reducing the cost of integration like Fast Fourier Transformation (FFT) for cos/sin functions. Recently a combination of FFT with recursive methods for associated Legendre functions was used to compute expansions of spherical harmonics [12,13]. However our simple algorithm offers a reasonable compromise between algorithm complexity and considerable computational savings. It is very easy to implement and generalize in many dimensions where it gives $2^{\mathrm{dim}}$ saving. Thus for the basis (1) the minimal grid on which the basis is still orthogonal is $(j_{\max} + 1) \times (l_{\max} + 1) \times (k_{\max} + 1)$ and therefore represents an eightfold saving. It turns out that using the reduced grid is very competitive compared to the expansion method. First of all the analytical integration requires $3j$-symbols. Unfortunately it is usually not practical to precompute and store them because the required storage size is too big. Therefore they need to be computed every time they are needed. Secondly the implementation of analytical integration requires some logic based on the $3j$-symbol rules but this typically degrades processor performance. On the other hand for the numerical integration we need only to precompute the basis functions and potential on the grid, and perform a direct multiplication-summation. The memory requirements

Table 1
Comparison of CPU times (in seconds) used to calculate the matrix elements given by Eq. (6) using analytical and numerical integrals

| Basis $(j_{\max}, l_{\max}, k_{\max})$ | Basis size | Minimal grid | |
|---|---|---|---|
| | | analytical | numerical |
| (15, 15, 15) | 1496 | 13.3 | 4.7 |
| (20, 20, 20) | 3311 | 81.2 | 36.8 |
| (25, 25, 15) | 5816 | 301.3 | 135.1 |

are small (for example, for $j_{\max} = l_{\max} = k_{\max} = 30$ one needs less than 1 MB) and the method performs very well. Note that there is an additional factor which makes the numerical integration quicker in our implementation, namely caching of the $\varphi$ integrals.

Our tests showed that the implementation in WAVR4 performs the numerical integration on the minimal grid much quicker than the expansion method (even not including the time to perform the expansion itself). The accuracy provided by the minimal grid was found to be acceptable because in variational calculations we are typically interested in the lowest eigenfunctions and therefore some inaccuracy of high matrix elements is not critical. In the application given below, the lowest 10% of the eigenvalues calculated by the two methods differ by less than 0.0002%. More importantly the errors in the eigenvalues due to convergence with respect to the basis size are usually larger than those due to inaccuracies in the integrations.

In practice our basis (1), is adapted to inversion symmetry which takes $\varphi \to 2\pi - \varphi$ (see Refs. [6,7]). Therefore our expansion is also parity adapted. It is straightforward to apply Eq. (4) again and show that the matrix elements have a similar form. Our grid in $\varphi$ is halved: the non-reduced minimal grid now has $(k_{\max} + 1)$ points; the reduced one has $(k_{\max} + 1)/2$ points if $k_{\max}$ is odd and $k_{\max}/2 + 1$ if $k_{\max}$ is even.

Typically in our implementation the time taken to compute the full matrix given by Eq. (6) using the numerical method is about half that for the analytical method. The results of test runs performed on a Compaq Alpha workstation (667 MHz) are given in Table 1. The test calculations were made on the angular states of acetylene, HCCH, using a literature potential [15] and three representative basis sets. Of the molecular symmetries, only inversion was used. The timings are given for the calculation of pure vibrational $(J = K = 0)$, totally symmetric states.

One may want to use a slightly bigger reduced grid to achieve a suitable compromise between speed and accuracy. For example, it has been suggested to use $(n_{\max} + 4)$ points instead of $(n_{\max} + 1)$ for polynomials of order $n_{\max}$ [14]. Only when high accuracy is vital does the use of an expansion become the preferable choice because comparable accuracy using numerical integration becomes more expensive. Both the expansion and numerical options are implemented in WAVR4 and the user may choose the optimal strategy for the problem of interest.

## Acknowledgements

The authors are grateful to Jeremy Hutson for many useful discussions. This work was funded by the UK Engineering and Physical Sciences Research Council as the “flagship” project of CCP6 (the Collaborative Computational Project on Molecular Quantum Dynamics).

## References

[1] J. Tennyson, Comput. Phys. Comm. 38 (1985) 39.
[2] I.N. Kozin, M.M. Law, J. Tennyson, J.M. Hutson, Comput. Phys. Comm. 163 (2004) 117.
[3] X. Chapuisat, C. Iung, Phys. Rev. A 45 (1992) 6217.
[4] F. Gatti, C. Iung, M. Menou, Y. Justum, A. Nauts, X. Chapuisat, J. Chem. Phys. 108 (1998) 8804.

[5] C. Iung, F. Gatti, A. Viel, X. Chapuisat, Phys. Chem. Chem. Phys. 1 (1999) 3377.
[6] M. Mladenović, J. Chem. Phys. 112 (2000) 1070.
[7] I.N. Kozin, M.M. Law, J.M. Hutson, J. Tennyson, J. Chem. Phys. 118 (2003) 4896.
[8] J.C. Light, I.P. Hamilton, J.V. Lill, J. Chem. Phys. 92 (1985) 1400.
[9] A.H. Stroud, D. Secrest, Gaussian Quadrature Formulas, Prentice-Hall, London, 1966.
[10] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes in FORTRAN: The Art of Scientific Computing, second ed., Cambridge University Press, Cambridge, England, 1992.
[11] J. Tennyson, A. van der Avoird, J. Chem. Phys. 77 (1982) 5664.
[12] J.R. Driscoll, D.M. Healy Jr., Adv. Appl. Math. 15 (1994) 202.
[13] V. Lesur, D. Gubbins, Geophys. J. Int. 139 (1999) 547.
[14] J. Tennyson, B.T. Sutcliffe, J. Chem. Phys. 77 (1982) 4061.
[15] S. Carter, I.M. Mills, J.N. Murrell, Mol. Phys. 41 (1980) 191;
L. Halonen, M.S. Child, S. Carter, Mol. Phys. 47 (1982) 1097.