IIPBF - a Matlab toolbox for computing infinite integrals of products of Bessel functions of the 1st and 2nd kind

Tilak Ratnanather

The Johns Hopkins University, Baltimore, MD 21218

Joint work with G. M. Gunter

Institute for Computational Medicine

Center for Imaging Science
Johns Hopkins University

THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

Introduction

$$\int_0^\infty f(x) B_{a,b,\rho,\tau}(x)dx \text{ where } B_{a,b,\rho,\tau}(x) = \begin{cases} J_a(\rho x) J_b(\tau x) \\ J_a(\rho x) Y_b(\tau x) \\ Y_a(\rho x) Y_b(\tau x) \end{cases}$$

occurs in elasticity, biophysics, electrodynamics, geology, ...

- `MATLAB` toolbox[1] for a, b: non-negative integer; $f(x)$: real valued, smooth and monotonic as $x \to \infty$
- adaptation of ISE - integration, summation, extrapolation - (Longman, 1956) for $J_a(\rho x) J_b(\tau x)$ (Lucas, 1995)
- extended[1] to $J_a(\rho x) Y_b(\tau x)$ and $Y_a(\rho x) Y_b(\tau x)$ for a fluid jet on a planar wall[2] with applications in cochlear physiology

[1] Ratnanather JT, Kim JH, Zhang S, Davis AMJ, Lucas SK (2013) Algorithm XXX: IIPBF, a MATLAB toolbox for infinite integral of products of two Bessel functions, ACM TOMS, in press

[2] Davis AMJ, Kim JH, Ceritoglu C, Ratnanather JT (2012) A Stokesian analysis of a submerged viscous fluid jet impinging on a planar wall. J. Fluid Mech. 712:531-551

New Applications

- kernels with spherical Bessel functions occur in MRI applications (Hosseinbor et al., 2013)
- general kernels with Bessel functions in image registration for biomedical shape analysis (Micheli and Glaunès, 2013)
- tomography imaging (Dodd and Deeds, 1968) involve complex valued kernels with Bessel functions
- oscillating jet on a planar wall involves complex valued kernels[3] with applications in cochlear mechanics
- Goal: effectiveness of `IIPBF` for Bessel functions with real-valued order and complex valued kernels

[3] Davis AM, Kim JH, Gunter GM, Ratnanather JT (2013) The Stokesian flow field of an oscillatory submerged viscous jet impinging on a planar wall. Proc. Roy. Soc. Lond. A, in press

Algorithm

- Adaptation of `FORTRAN77` algorithm (Lucas, 1995)

  Integration: `MATLAB` conversion of quadrature routines in `SLATEC` (Barrowes, 2009)

  Summation: accelerates convergence by decomposing product as a sum of high and low frequency components and subdivision via zeros of components

  Extrapolation: ε-algorithm (`dqelg`) from `QUADPACK` and mW transform (Sidi, 1988, 2012) to deal with oscillatory components and acceleration

- `dqage` (definite integrals), `dqagie` (infinite integrals), and `dqelg` (ε-algorithm) from `SLATEC`
- Adaptive Gauss-Kronrod quadrature - `quadgk` - (Shampine, 2008) instead of `dqk15` and `dqk15i`

$$B_{a,b,\rho,\tau}(x) = h_1(x; a, b, \rho, \tau) + h_2(x; a, b, \rho, \tau)$$

$J_a(\rho x) J_b(\tau x)$

$$h_1(x; a, b, \rho, \tau) = \frac{1}{2}(J_a(\rho x)J_b(\tau x) - Y_a(\rho x)Y_b(\tau x)) \sim \frac{1}{\pi\sqrt{\rho\tau}x}\cos\left((\rho+\tau)x - \frac{(a+b+1)\pi}{2}\right)$$

$$h_2(x; a, b, \rho, \tau) = \frac{1}{2}(J_a(\rho x)J_b(\tau x) + Y_a(\rho x)Y_b(\tau x)) \sim \frac{1}{\pi\sqrt{\rho\tau}x}\cos\left((\rho-\tau)x - \frac{(a-b)\pi}{2}\right)$$

$J_a(\rho x) Y_b(\tau x)$

$$h_1(x; a, b, \rho, \tau) = \frac{1}{2}(J_a(\rho x)Y_b(\tau x) + Y_a(\rho x)J_b(\tau x)) \sim \frac{1}{\pi\sqrt{\rho\tau}x}\sin\left((\rho+\tau)x - \frac{(a+b+1)\pi}{2}\right)$$

$$h_2(x; a, b, \rho, \tau) = \frac{1}{2}(J_a(\rho x)Y_b(\tau x) - Y_a(\rho x)J_b(\tau x)) \sim -\frac{1}{\pi\sqrt{\rho\tau}x}\sin\left((\rho-\tau)x - \frac{(a-b)\pi}{2}\right)$$

$Y_a(\rho x) Y_b(\tau x)$

$$h_1(x; a, b, \rho, \tau) = -\frac{1}{2}(J_a(\rho x)J_b(\tau x) - Y_a(\rho x)Y_b(\tau x)) \sim -\frac{1}{\pi\sqrt{\rho\tau}x}\cos\left((\rho+\tau)x - \frac{(a+b+1)\pi}{2}\right)$$

$$h_2(x; a, b, \rho, \tau) = \frac{1}{2}(J_a(\rho x)J_b(\tau x) + Y_a(\rho x)Y_b(\tau x)) \sim \frac{1}{\pi\sqrt{\rho\tau}x}\cos\left((\rho-\tau)x - \frac{(a-b)\pi}{2}\right)$$

Zeros of h_1 and h_2 for mW transform and ε-algorithm

- ISE applied to $f(x)h_1(x)$ and $f(x)h_2(x)$
- $h_{1,1}$ and $h_{2,1}$: 1st zeros of $h_1(x)$ and $h_2(x)$ respectively
- $Y_a(\rho x)$ and $Y_b(\tau x)$ are singular near 0 so use approximate zeros (Olver et al., 2010)
- Two scenarios
  - $\rho = \tau$: split $[0, \infty)$ into $[0, ymax]$ and $[ymax, \infty)$ used for $f(x)B_{a,b,\rho,\tau}(x)$ and $f(x)h_1(x) + f(x)h_2(x)$ respectively

    h_1: $ymax$ from 1st zero of Y_a or Y_b and asymptotic form used for subsequent zeros

    h_2: non-oscillating and monotonic decreasing so zeros not used
  - $\rho \neq \tau$: split $[0, \infty)$ into $[0, ymin]$, $[ymin, ymax]$ and $[ymax, \infty)$ with $ymin = min(h_{1,1}, h_{2,1})$ and $ymax = max(h_{1,1}, h_{2,1})$ where $h_{1,1}$ and $h_{2,1}$ are obtained from the 1st zeros of Y_a or Y_b

    h_1: asymptotic form used for subsequent zeros

    h_2: zeros after $h_{2,1}$ obtained via stepwise increments $\pi / |\rho - \tau|$

mW transform: Generalized Richardson Extrapolation Process

Following Sidi (1988, 2012), begin with $S_0 = \int_a^{x_0} f(x)h_1(x)dx$, $T_0 = \int_{x_0}^{x_1} f(x)h_1(x)dx$ and iterate until $W_t < \mathrm{TOL}$:

$$\begin{aligned} S_{t+1} &= S_t + T_t = \int_a^{x_{t+1}} f(x)h_1(x)dx \\ T_{t+1} &= \int_{x_{t+1}}^{x_{t+2}} f(x)h_1(x)dx \\ M_{t+1,-1} &= S_{t+1}/T_{t+1}, N_{t+1,-1} = 1/T_{t+1} \end{aligned}$$

and for $s = t, t-1, \ldots, 0$

$$\begin{aligned} M_{s,t-s} &= (M_{s,t-s-1} - M_{s+1,t-s-1})/(1/x_s - 1/x_{t+1}) \\ N_{s,t-s} &= (N_{s,t-s-1} - N_{s+1,t-s-1})/(1/(x_s - 1/x_{t+1}) \\ W_t &= M_{0,t}/N_{0,t} \end{aligned}$$

ε-algorithm: Aitken, Euler, Padé, Shanks, Stieltjes, Wynn

- nonlinear transformation of a slowly converging sequence which identifies and removes oscillatory transients
- given partial sum $\{A_n\}_{n=0}^{\infty}$, $\varepsilon_n^{(-1)} = 0$ and $\varepsilon_n^{(0)} = A_n$:

$$\varepsilon_n^{(p)} = \varepsilon_n^{(p-2)} + \left[\varepsilon_{n+1}^{(p-1)} - \varepsilon_n^{(p-1)}\right]^{-1}$$

- $\varepsilon^{(2k)}$ is the kth Shanks' transform of the sequence $\{A_n\}$
- approximate zeros permits ignorance of early poor results before h_2 settles to a simple oscillation

Modifications

- $|\rho/\tau - 1| < 10^{-10}$: h_2 is almost monotone so standard infinite integration
- $|\rho/\tau| > 10^2$ or $|\tau/\rho| > 10^2$: asymptotic forms of h_1 and h_2 have almost the same frequency so use ε-algorithm
- small values of ρ or τ can cause the 1st zeros of Y_a and Y_b to be large so the minimum of these zeros is used when the ratio of the larger zero to the smaller one is too large; otherwise the maximum is used as recommended (Lucas, 1995)

Flow Chart

if $|\rho/\tau - 1| < 10^{-10}$ **then**

$\cdots$

else if $|\rho/\tau| > 10^2$ *or* $|\tau/\rho| > 10^2$ **then**

$\cdots$

else

Calculate $h_{1,1}$ and $h_{2,1}$;

$I_2 = \int_{h_{1,1}}^{\infty} f(x)h_1(x; a, b, \rho, \tau)dx$ using mW transform;

$I_3 = \int_{h_{2,1}}^{\infty} f(x)h_2(x; a, b, \rho, \tau)dx$ using ε-algorithm;

if $h_{1,1} < h_{2,1}$ **then**

$I_1 = \int_{0}^{h_{1,1}} f(x)B_{a,b,\rho,\tau}(x)dx$ using `quadgk`;

$I_4 = \int_{h_{1,1}}^{h_{2,1}} f(x)h_2(x; a, b, \rho, \tau)dx$ using `quadgk`

else

$I_1 = \int_{0}^{h_{2,1}} f(x)B_{a,b,\rho,\tau}(x)dx$ using `quadgk`;

$I_4 = \int_{h_{2,1}}^{h_{1,1}} f(x)h_1(x; a, b, \rho, \tau)dx$ using `quadgk`

end

end

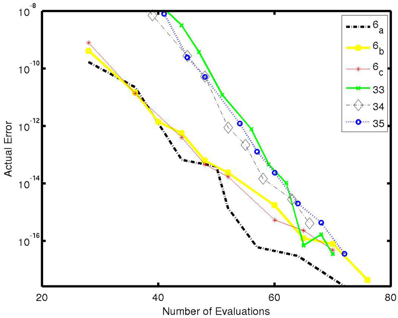
$I_1 + I_2 + I_3 + I_4$

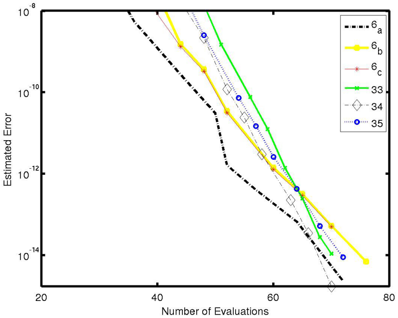
Test Cases

- 32 test cases in `IIPBF` including 23 already tested[1]
- so consider complex-valued kernel for one of the the 23 cases plus 3 new ones from Gradshteyn and Ryzhik (2007)

Case	Integrand	Value
6	$xK_0(xc)J_0(\rho x)J_0(\tau x)$	$1/\left(\left(c^2+\rho^2+\tau^2\right)^2-4\rho^2\tau^2\right)^{1/2}$
33	$\frac{x}{x^2+c^2}J_a(\rho x)J_a(\tau x)$	$I_a(\tau c)K_a(\rho c)$ for $0<\tau<\rho$ and $I_a(\rho c)K_a(\tau c)$ for $0<\rho<\tau$ $\operatorname{Re} c>0, \operatorname{Re} a>-1$
34	$x^{-c}J_a(\rho x)J_b(\tau x)$	$\dfrac{\tau^a\Gamma\left(\frac{a+b-c+1}{2}\right)}{2^c\rho^{b-c+1}\Gamma\left(\frac{a-b+c+1}{2}\right)\Gamma(a+1)}$ $\times F\left(\frac{a+b-c+1}{2}, \frac{-a+b-c+1}{2}; b+1; \frac{\tau^2}{\rho^2}\right)$ $\operatorname{Re}(a+b-c+1)>0, \operatorname{Re} c>-1, 0<\tau<\rho$
35	$x^{-c}J_a(\rho x)Y_b(\tau x)$	$\dfrac{2}{\pi}\sin\left(\dfrac{\pi(a-b-c)}{2}\right)\dfrac{\rho^a\Gamma(\frac{1}{2}-\frac{c}{2}+\frac{b}{2}+\frac{a}{2})\Gamma(\frac{1}{2}-\frac{c}{2}-\frac{b}{2}+\frac{a}{2})}{2^{c+1}\Gamma(a+1)\tau^{-c+a+1}}$ $\times F(\frac{1}{2}-\frac{c}{2}+\frac{b}{2}+\frac{a}{2}, \frac{1}{2}-\frac{c}{2}-\frac{b}{2}+\frac{a}{2}; a+1; \frac{\rho^2}{\tau^2})$ $\operatorname{Re}(a\pm b-c+1)>0, \operatorname{Re} c>-1, \tau<\rho$

Actual and Estimated Errors: $xK_0(xc)J_0(\rho x)J_0(\tau x)$, $\frac{x}{x^2+c^2}J_a(\rho x)J_b(\tau x)$, $x^{-c}J_a(\rho x)J_b(\tau x)$, $x^{-c}J_a(\rho x)Y_b(\tau x)$





- number of function evaluations of h_1 and h_2 based on relative error tolerances from 10^{-4} to 10^{-14}
- for case 6, $\rho = 0.1, \tau = 100$ and $c = \alpha + \imath\beta$ where $(\alpha, \beta) \in \{(10, 2), (2, 2), (2, 10)\}$
- for cases 33, 34 and 35: $\{c, a, b, \rho, \tau\} = \{2, 1, 1, 2, 1\}$, $\{2, 1, 2, 2, 1\}$ and $\{1, 2, 1, 1, 2\}$ respectively

Comparison with BESSELINT (Van Deun and Cools, 2008)

- for $f(x)\Pi_{i=1}^{k} J_{a_i}(\rho_i x)$ where $f(x) = x^s e^{-ux}/(t^2 + x^2)$
- Case 33: $x^{-c} J_a(\rho x) J_b(\tau x)$
- Case 34: $x^{-c} J_a(\rho x) Y_b(\tau x)$

Case	Time (secs)		Estimated Error		Absolute
	BESSELINT	IIPBF	BESSELINT	IIPBF	Difference
33	0.1099	0.1834	1.85 x10^{-12}	1.16 x10^{-15}	2.19 x10^{-16}
34	0.0950	0.1805	4.73 x10^{-15}	1.43 x10^{-15}	9.58 x10^{-16}

- bottleneck in both is due to the infinite integration routine
- MATLAB version 7.14.0.739 (R2012a) on an Intel(R) Xeon(R) CPU E31290 at 3.60GHz which is a 64bit machine

(ρ, τ) parameter tests for Case 35: $x^{-c} J_a(\rho x) Y_b(\tau x)$

$a = 2.5$, $b = 1.5$, $c = 1$

		ρ						
		0.001	0.01	0.1	1	10	100	1000
τ	0.0011	3.18 x10^{-13}	-	-	-	-	-	-
	0.011	4.12 x10^{-13}	3.18 x10^{-13}	-	-	-	-	-
	0.11	1.71 x10^{-13}	4.12 x10^{-13}	3.18 x10^{-13}	-	-	-	-
	1.1	2.91 x10^{-11}	1.99 x10^{-13}	4.12 x10^{-13}	3.18 x10^{-13}	-	-	-
	11	0.00	7.28 x10^{-12}	1.71 x10^{-13}	4.12 x10^{-13}	3.18 x10^{-13}	-	-
	101	0.00	0.00	7.28 x10^{-12}	3.69 x10^{-13}	7.12 x10^{-13}	1.46 x10^{-13}	-
	1001	0.00	0.00	0.00	1.46 x10^{-11}	3.13 x10^{-13}	8.36 x10^{-12}	2.52 x10^{-13}

$a = 4.2$, $b = 0.3$, $c = 2.5$

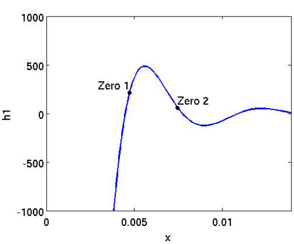
		ρ						
		0.001	0.01	0.1	1	10	100	1000
τ	0.0011	1.35 x10^{-12}	-	-	-	-	-	-
	0.011	1.48 x10^{-12}	6.64 x10^{-13}	-	-	-	-	-
	0.11	1.08 x10^{-11}	1.42 x10^{-12}	1.12 x10^{-12}	-	-	-	-
	1.1	2.21 x10^{-16}	1.13 x10^{-10}	1.11 x10^{-12}	2.11 x10^{-13}	-	-	-
	11	4.41 x10^{-19}	6.98 x10^{-15}	3.84 x10^{-9}	5.80 x10^{-13}	2.01 x10^{-13}	-	-
	101	1.11 x10^{-21}	1.75 x10^{-17}	2.78 x10^{-13}	5.52 x10^{-8}	5.04 x10^{-13}	2.47 x10^{-13}	-
	1001	2.26 x10^{-24}	3.59 x10^{-20}	5.68 x10^{-16}	9.01 x10^{-12}	1.43 x10^{-7}	6.36 x10^{-13}	3.27 x10^{-13}

1st zeros of h_1

Case 35: $x^{-c} J_a(\rho x) Y_b(\tau x)$ for $\rho = 10$, $\tau = 1001$

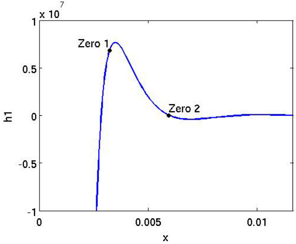
$c = 1$; $a = 2.5$; $b = 1.5$

Error: 3.13×10^{-13}



$c = 2.5$; $a = 4.2$; $b = 0.3$

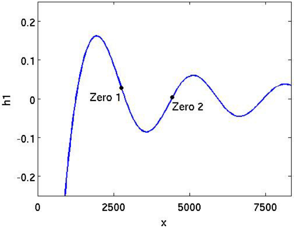
Error: 1.43×10^{-7}



Case 23: $x^{b-a+1} J_a(\rho x) J_b(\tau x)/(x^2 + c^2)$ for $\rho = 0.001$, $\tau = 0.0011$

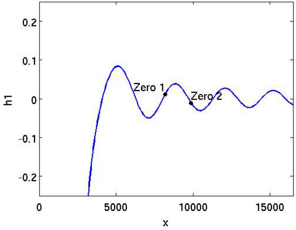
$c = 2$; $a = 1$; $b = 1$

Error: 3.70×10^{-14}



$c = 2$; $a = 3$; $b = 5$

Error: 1.04×10^{-7}



Discussion

- `quadgk` enabled error estimates with significantly less function evaluations than before which is not surprising (Gonnet, 2012) but its structure makes it difficult to vectorize mW transform
- product decomposition avoids loss of precision when mW transform is applied to the product (Sidi, 2012)
- `IIPBF` works for a wide range of ρ and τ with higher orders but for extreme values results should be checked
- $|a - b| > 5$ works (Lucas, 1995) but problems when the Bessel functions become very small (Van Deun and Cools, 2008)
- warnings when the maximum number of intervals in `quadgk` is exceeded due to either the first zero or the spacing between the first two zeros or both becoming too big
- wild idea: best of both `IIPBF` and `BESSELINT` in `Chebfun`?
- `http://www.cis.jhu.edu/software/iipbf/`

References

Barrowes, B. (2009). *The SLATEC library converted into MATLAB functions*. http://www.mathworks.com/matlabcentral/fileexchange/14535.

Dodd, C. V. and Deeds, W. E. (1968). Analytical solutions to eddy-current probe-coil problems. *J. App. Phys.*, 39:2829–2838.

Gonnet, P. (2012). A review of error estimation in adaptive quadrature. *ACM Comput. Surv.*, 44(4):22:1–22:36.

Gradshteyn, I. S. and Ryzhik, I. M. (2007). *Gradshteyn and Ryzhik's Table of Integrals, Series, and Products*. Academic Press, San Diego, 7th edition. A. Jeffrey and D. Zwillinger (Eds.).

Hosseinbor, A. P., Chung, M. K., Wu, Y.-C., and Alexander, A. L. (2013). Bessel Fourier Orientation Reconstruction (BFOR): an analytical diffusion propagator reconstruction for hybrid diffusion imaging and computation of q-space indices. *Neuroimage*, 64:650–670.

Longman, I. M. (1956). Note on a method for computing infinite integrals of oscillatory functions. *Math. Proc. Cambridge Phil. Soc.*, 52:764–768.

Lucas, S. K. (1995). Evaluating infinite integrals involving products of Bessel functions of arbitrary order. *J. Comp. App. Math.*, 64:269–282.

Olver, F. W. J., Lozier, D. W., Boisvert, R. F., and Clark, C. W. (2010). *NIST Handbook of Mathematical Functions: Companion to the Digital Library of Mathematical Functions*. Cambridge University Press, New York.

Shampine, L. F. (2008). Vectorized adaptive quadrature in MATLAB. *J. Comp. App. Math.*, 211:131–140.

Sidi, A. (1988). A user-friendly extrapolation method for oscillatory infinite integrals. *Math. Comp.*, 51:249–266.

Sidi, A. (2012). A user-friendly extrapolation method for computing infinite range integrals of products of oscillatory functions. *IMA J. Num. Anal.*, 32:602–631.

Van Deun, J. and Cools, R. (2008). Integrating products of Bessel functions with an additional exponential or rational factor. *Comp. Phys. Comm.*, 178:578–590.