Applied Numerical Mathematics 60 (2010) 1411–1417

Contents lists available at ScienceDirect

Applied Numerical Mathematics

www.elsevier.com/locate/apnum

A recursive algorithm for the G transformation and accurate computation of incomplete Bessel functions

Richard M. Slevinsky, Hassan Safouhi *

Mathematical Section, Campus Saint-Jean, University of Alberta, 8406, 91 Street, Edmonton (AB) T6C 4G9, Canada

ARTICLE INFO

Article history:
Received 2 December 2009
Received in revised form 18 April 2010
Accepted 23 April 2010
Available online 28 April 2010

Keywords:
The Slevinsky–Safouhi formulae
Extrapolation methods
Nonlinear transformations
Incomplete Bessel functions

ABSTRACT

In the present contribution, we develop an efficient algorithm for the recursive computation of the $G_n^{(1)}$ transformation for the approximation of infinite-range integrals. Previous to this contribution, the theoretically powerful $G_n^{(1)}$ transformation was handicapped by the lack of an algorithmic implementation. Our proposed algorithm removes this handicap by introducing a recursive computation of the successive $G_n^{(1)}$ transformations with respect to the order n. This recursion, however, introduces the $(x^2\frac{\mathrm{d}}{\mathrm{d}x})$ operator applied to the integrand. Consequently, we employ the Slevinsky–Safouhi formula I for the analytical and numerical developments of these required successive derivatives.

Incomplete Bessel functions, which pose as a numerical challenge, are computed to high pre-determined accuracies using the developed algorithm. The numerical results obtained show the high efficiency of the new method, which does not resort to any numerical integration in the computation.

1. Introduction

The recent articles [27,28] have shown the high accuracy that can be achieved by the $D_n^{(m)}$ transformation [21] and its confluent form the $G_n^{(m)}$ transformation [9] applied to semi-infinite highly oscillatory integrals. The positive integer m stands for the order of the linear differential equation satisfied by the integrand and n stands for the order of the transformation and as n gets large, the approximations obtained by the transformations converge to the exact values of the integrals. Although these examples have shown the tremendous potential of the $G_n^{(m)}$ transformation as a computational tool, application of the $G_n^{(m)}$ transformation has been hampered by the lack of an algorithmic implementation and thus requires the solution of large systems of linear equations. Combining this with the fact that oscillatory integrals satisfy linear homogeneous differential equations of high order, the application of the $G_n^{(m)}$ transformation becomes extremely challenging.

Previous to [27,28], the $G_n^{(m)}$ transformation has also been applied in [23], where it was used to approximate probability density functions, whose integrands all satisfied first order linear homogeneous differential equations. This simplicity led the authors to some analytical expressions for the anti-derivatives of infinite-range integral tails. Once again, however, symbolic and numerical computation were hampered by the lack of algorithms for the implementation of the $G_n^{(m)}$ transformation.

In this work, we used the Slevinsky–Safouhi formula I to derive an analytic expression of the approximations $G_n^{(1)}$. The obtained analytical expression can be computed recursively with respect to the order n and here again the Slevinsky–Safouhi formula I is used for the analytic development of the terms involved in the computation.

* Corresponding author.
E-mail address: hassan.safouhi@ualberta.ca (H. Safouhi).

0168-9274/$30.00
doi:10.1016/j.apnum.2010.04.005

Incomplete Bessel functions were a subject of significant research and we refer the interested reader to these articles [18,19,10–13,16,20,14,15] for a rich history of these functions. Of the many applications of incomplete Bessel functions, we note that they appear when Ewald-type summation acceleration procedures [8] are applied to electronic-structure calculations for systems described in terms of Gaussian-type atomic orbitals, with periodicity in one, two, or all three physical dimensions. Incomplete Bessel functions of zero order are also involved in a numerous applications to electromagnetic waves [2,22,4,7,24].

Due to their integral representation [14]:

$$K_\nu(x,y)=\int_1^\infty \frac{e^{-xt-y/t}}{t^{\nu+1}}\,\mathrm{d}t, \tag{1}$$

incomplete Bessel functions are a computational challenge. Equipped with the developed algorithm, we apply the $G_n^{(1)}$ transformation to compute incomplete Bessel functions to high pre-determined accuracies. Our method does not require any numerical integration in the computation of incomplete Bessel functions. We also demonstrate that this algorithm allows for a broad range of incomplete Bessel computation.

2. A few properties

Integration by parts of the integral representation of $K_\nu(x,y)$ in (1) leads to the inhomogeneous recurrence formula [30,5]:

$$xK_{\nu-1}(x,y)+\nu K_\nu(x,y)-yK_{\nu+1}(x,y)=e^{-x-y}. \tag{2}$$

Defining the modified Bessel function using the formula [1]:

$$K_\nu(z)=\frac{1}{2}\int_0^\infty \frac{e^{-(z/2)(t+1/t)}}{t^{\nu+1}}\,\mathrm{d}t, \tag{3}$$

and defining $u=\sqrt{xy}$ and $v=\sqrt{\frac{x}{y}}$, we have another important functional relation [14]:

$$K_\nu(x,y)+K_{-\nu}(y,x)=2v^\nu K_\nu(2u). \tag{4}$$

By interchanging $x\leftrightarrow y$, Eq. (4) can effectively double the applicable region of any algorithm, provided modified Bessel functions can be calculated. In terms of u and v, Eq. (1) can be expressed as [14]:

$$K_\nu(u,v)=\int_v^\infty \frac{v^\nu e^{-u(t+1/t)}}{t^{\nu+1}}\,\mathrm{d}t, \tag{5}$$

or, it can be expressed as a generalized incomplete gamma function [6,5]:

$$K_\nu(x,y)=x^\nu\Gamma(-\nu;x;xy)\quad\text{where }\Gamma(\alpha;x;b)=\int_x^\infty t^{\alpha-1}e^{-t-b/t}\,\mathrm{d}t. \tag{6}$$

Lastly, we present the Slevinsky–Safouhi formula I [29]:

Theorem 2.1. *(See [29].) Let $G(x)$ be a function kth differentiable with the term $(\frac{\mathrm{d}}{x^m\mathrm{d}x})^k(x^{-n}G(x))$ well defined. The Slevinsky–Safouhi formula I is given by*:

$$\left(\frac{\mathrm{d}}{x^\mu\mathrm{d}x}\right)^k\left(x^{-\nu}G(x)\right)=\sum_{i=0}^k A_k^i x^{n-\nu+i(m+1)-k(\mu+1)}\left(\frac{\mathrm{d}}{x^m\mathrm{d}x}\right)^i\left(x^{-n}G(x)\right), \tag{7}$$

with coefficients:

$$A_k^i=\begin{cases}1 & \text{for } i=k,\\ (n-\nu-(k-1)(\mu+1))A_{k-1}^0 & \text{for } i=0,\ k>0,\\ (n-\nu+i(m+1)-(k-1)(\mu+1))A_{k-1}^i+A_{k-1}^{i-1} & \text{for } 0<i<k.\end{cases} \tag{8}$$

3. The $G_n^{(m)}$ transformation

Theorem 3.1. *(See [21].) Let $f(x)$ be integrable on $[0,\infty)$ (i.e. $\int_0^\infty f(t)\,\mathrm{d}t$ exists) and satisfy a linear differential equation of order m of the form $f(x)=\sum_{k=1}^{m} p_k(x) f^{(k)}(x)$ where p_k for $k=1,2,\ldots,m$ have asymptotic expansions as $x\to\infty$, of the form*:

$$p(x)\sim x^{i_k}\left(a_0+\frac{a_1}{x}+\frac{a_2}{x^2}+\cdots\right)=x^{i_k}\sum_{i=0}^{\infty}\frac{a_i}{x^i}\quad \text{with } i_k\leqslant k. \tag{9}$$

If for $1\leqslant i\leqslant m$ and $i\leqslant k\leqslant m$, we have $\lim_{x\to\infty} p_k^{(i-1)}(x) f^{(k-i)}(x)=0$ and for every integer $l\geqslant -1$, we have $\sum_{k=1}^{m} l(l-1)\cdots(l-k+1)p_{k,0}\neq 1$ where $p_{k,0}=\lim_{x\to\infty} x^{-k}p_k(x)$ for $1\leqslant k\leqslant m$, then as $x\to\infty$, we have:

$$\int_x^\infty f(t)\,\mathrm{d}t\sim\sum_{k=0}^{m-1} x^{j_k} f^{(k)}(x)\left(\beta_{0,k}+\frac{\beta_{1,k}}{x}+\frac{\beta_{2,k}}{x^2}+\cdots+\right), \tag{10}$$

where $j_k\leqslant \max(i_{k+1}, i_{k+2}-1,\ldots,i_m-m+k+1)\quad$ for $\quad k=0,1,\ldots,m-1$.

To solve for the unknowns $\beta_{k,i}$, we must set up and solve a system of linear equations. To produce this system of linear equations, few methods have been conceived. The first is called the $D_n^{(m)}$ transformation. For this transformation, a set of interpolating points $x_0, x_1,\ldots,x_l$ is used to solve for the unknowns.

The approximation $D_n^{(m)}$ of $\int_0^\infty f(t)\,\mathrm{d}t$, using the nonlinear D transformation, satisfies the $(nm+1)$ equations given by [21]:

$$D_n^{(m)}=\int_0^{x_l} f(t)\,\mathrm{d}t+\sum_{k=0}^{m-1} f^{(k)}(x_l)x_l^{\sigma_k}\sum_{i=0}^{n-1}\frac{\bar\beta_{k,i}}{x_l^i},\quad l=0,1,\ldots,nm, \tag{11}$$

where σ_k for $k=0,1,\ldots,m-1$, are the minima of $k+1$ and s_k where s_k is the largest of the integers s for which $\lim_{x\to+\infty} x^s f^{(k)}(x)=0$. $D_n^{(m)}$ and $\bar\beta_{k,i}$ for $k=0,1,\ldots,m-1$, $i=0,1,\ldots,n-1$ are the $(nm+1)$ unknowns. The x_l, $l=0,1,\ldots$ are chosen to satisfy $0<x_0<x_1<\cdots$ and $\lim_{l\to+\infty} x_l=+\infty$.

The order of the above set of equations can be reduced to $n(m-1)+1$ by choosing x_l for $l=0,1,\ldots$ to be the leading positive zeros of $f(x)$. In this case, $\bar D$ is applied, and the linear system (11) will be re-written as [25]:

$$\bar D_n^{(m)}=\int_0^{x_l} f(t)\,\mathrm{d}t+\sum_{k=1}^{m-1} f^{(k)}(x_l)x_l^{\sigma_k}\sum_{i=0}^{n-1}\frac{\bar\beta_{k,i}}{x_l^i},\quad l=0,1,\ldots,n(m-1). \tag{12}$$

If we take the limiting case as all the points coalesce, we achieve the first confluent form [26] of the $D_n^{(m)}$ transformation, known as the $G_n^{(m)}$ transformation [9]. The approximation $G_n^{(m)}$ to $\int_0^\infty f(t)\,\mathrm{d}t$ is given as the solution of the system of $mn+1$ linear equations [9]:

$$\frac{\mathrm{d}^l}{\mathrm{d}x^l}\left\{G_n^{(m)}-\int_0^x f(t)\,\mathrm{d}t-\sum_{k=0}^{m-1} x^{\sigma_k} f^{(k)}(x)\sum_{i=0}^{n-1}\frac{\bar\beta_{k,i}}{x^i}\right\}=0,\quad l=0,1,\ldots,mn, \tag{13}$$

where it is assumed that $\frac{\mathrm{d}^l}{\mathrm{d}x^l}G_n^{(m)}\equiv 0,\forall l>0$. In the above system (13), $\sigma_k=\min(s_k,k+1)$ where s_k is the largest of the integers s such that $\lim_{x\to\infty} x^s f^{(k)}(x)=0$ holds, $k=0,1,\ldots,m-1$. Also, $G_n^{(m)}$ and $\bar\beta_{k,i}$ are the respective set of $mn+1$ unknowns.

3.1. Algorithm for the $G_n^{(1)}$ transformation

The $G_n^{(1)}$ transformation can be written as the solution to the linear system (13) with $m=1$.

Instead of solving the system of linear equations each time for each order n, it would be ideal to resolve each approximation $G_n^{(1)}$ in a recursive manner.

By considering the equation for $l=0$:

$$G_n^{(1)}-\int_0^x f(t)\,\mathrm{d}t=x^{\sigma_0} f(x)\sum_{i=0}^{n-1}\frac{\bar\beta_{0,i}}{x^i}, \tag{14}$$

and by isolating the summation on the right hand side, we obtain:

$$\frac{G_n^{(1)} - \int_0^x f(t)\,\mathrm{d}t}{x^{\sigma_0} f(x)} = \sum_{i=0}^{n-1} \frac{\bar{\beta}_{0,i}}{x^i}. \tag{15}$$

To eliminate the summation, and consequently all of the unknowns $\bar{\beta}_{0,i}$, we must apply some type of operator to both sides of the equation. In the non-confluent case, this is achieved by the divided difference operator acting on the different interpolation points of the $D_n^{(1)}$ transformation. This culminates with the conception of the W algorithm [26]. In the confluent case, we require the $(x^2 \frac{\mathrm{d}}{\mathrm{d}x})$ operator, which, applied n times, eliminates the summation. For example, if we apply the $(x^2 \frac{\mathrm{d}}{\mathrm{d}x})$ operator to the summation, we obtain:

$$\left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)\left(\sum_{i=0}^{n-1} \frac{\bar{\beta}_{0,i}}{x^i}\right) = x^2 \sum_{i=1}^{n-1} \frac{-i\bar{\beta}_{0,i}}{x^{i+1}} = \sum_{i=1}^{n-1} \frac{-i\bar{\beta}_{0,i}}{x^{i-1}} \tag{16}$$

and the first unknown $\bar{\beta}_{0,0}$ disappears. Successive application will continue to eliminate the unknowns in this fashion and we obtain:

$$\left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)^n \left[\frac{G_n^{(1)} - \int_0^x f(t)\,\mathrm{d}t}{x^{\sigma_0} f(x)}\right] = 0, \tag{17}$$

which leads to a recursive algorithm for the $G_n^{(1)}$ transformation.

3.2. The algorithm

1. Where $F(x) = \int_0^x f(t)\,\mathrm{d}t$, set:

$$\mathcal{N}_0(x) = \frac{F(x)}{x^{\sigma_0} f(x)} \quad \text{and} \quad \mathcal{D}_0(x) = \frac{1}{x^{\sigma_0} f(x)}. \tag{18}$$

2. For $n = 1, 2, \ldots$, compute $\mathcal{N}_n(x)$ and $\mathcal{D}_n(x)$ recursively from:

$$\mathcal{N}_n(x) = \left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)\mathcal{N}_{n-1}(x) \quad \text{and} \quad \mathcal{D}_n(x) = \left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)\mathcal{D}_{n-1}(x). \tag{19}$$

3. For all n, the approximations $G_n^{(1)}(x)$ to $(\int_0^x + \int_x^\infty) f(t)\,\mathrm{d}t$ are given by:

$$G_n^{(1)}(x) = \frac{\mathcal{N}_n(x)}{\mathcal{D}_n(x)}. \tag{20}$$

The development of an algorithm for the case of $m = 2$, i.e. for the $G_n^{(2)}$ transformation, is of interest as many oscillatory integrals satisfy second order linear homogeneous differential equations of the form required for Theorem 3.1. For the case of $m > 1$, general algorithms could be constructed based on the E algorithm [3] and the FS algorithm [26]. However, as the G transformation is a confluent transformation, in that the linear system (13) is essentially created by differentiation, algorithms stemming from the E algorithm or the FS algorithm would be symbolic in nature, due to the recursive differentiation involved.

4. Computing incomplete Bessel functions

Since incomplete Bessel functions satisfy a first order linear homogeneous differential equation, $m = 1$ in Theorem 3.1, we use the $G_n^{(1)}$ transformation in order to obtain the evaluation of $K_\nu(x, y)$ for a wide range of the involved parameter and variables to a high pre-determined accuracy. The method does not use numerical integration through quadrature routines in any capacity. We begin our numerical discussion with the following equation obtained from (6):

$$K_\nu(x, y) + x^\nu \int_0^x \frac{e^{-t-xy/t}}{t^{\nu+1}}\,\mathrm{d}t = x^\nu \int_0^\infty \frac{e^{-t-xy/t}}{t^{\nu+1}}\,\mathrm{d}t. \tag{21}$$

The integrands $f(t) = \frac{e^{-t-xy/t}}{t^{\nu+1}}$ in (21) satisfy the first order linear homogeneous differential equation:

$$f(t) = -\frac{t^2}{t^2 - xy + (\nu+1)t} f'(t), \tag{22}$$

whereupon the conditions of applicability of Theorem 3.1 can be proved and where we find $\sigma_0 = 0$. Symbolically programming the $G_1^{(1)}$ transformation to the right hand side of (21) through our algorithm gives:

$$G_1^{(1)}(x,y,\nu) = \frac{\mathcal{N}_1(x)}{\mathcal{D}_1(x)} = \frac{x^{\nu+2}}{x^2 - xy + (\nu+1)x} f(x) + x^{\nu} \int_0^x \frac{e^{-t-xy/t}}{t^{\nu+1}}\,\mathrm{d}t. \tag{23}$$

Since incomplete Bessel functions are defined as integral tails rather than complete semi-infinite integrals, the remaining integral appears on both sides of the equation. We can then extract the approximation to the functions $K_\nu(x,y)$:

$$\tilde{G}_1^{(1)}(x,y,\nu) = \frac{xe^{-x-y}}{x^2 - xy + (\nu+1)x}. \tag{24}$$

Low order transformations like (24), however, may not be sufficient to cover the entire relevant range of parameter ν and variables x and y. By expanding the derivations involved in the functions $\mathcal{N}_n(x)$ and $\mathcal{D}_n(x)$ in the algorithm, notably:

$$\mathcal{N}_n(x) = \left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)^n \frac{F(x)}{x^{\sigma_0} f(x)} \quad \text{and} \quad \mathcal{D}_n(x) = \left(x^2 \frac{\mathrm{d}}{\mathrm{d}x}\right)^n \frac{1}{x^{\sigma_0} f(x)}, \tag{25}$$

and by proceeding as above, we are able to develop explicitly the numerator $\tilde{\mathcal{N}}_n(x,y,\nu)$ and denominator $\tilde{\mathcal{D}}_n(x,y,\nu)$ of the approximations $\tilde{G}_n^{(1)}(x,y,\nu)$ for incomplete Bessel functions.

For the development of $\tilde{\mathcal{D}}_n(x,y,\nu)$, we use the Leibniz product rule and the Slevinsky–Safouhi formula I with $(\mu,\nu,m,n) = (-2,-\nu-1,0,0)$ as follows:

$$\begin{aligned}
\tilde{\mathcal{D}}_n(x,y,\nu) &= \left(t^2 \frac{\mathrm{d}}{\mathrm{d}t}\right)^n \left(t^{\nu+1} e^{t+xy/t}\right)\Big|_{t=x} \\
&= \sum_{r=0}^{n} \binom{n}{r} \left(t^2 \frac{\mathrm{d}}{\mathrm{d}t}\right)^{n-r} e^{xy/t}\Big|_{t=x} \left(t^2 \frac{\mathrm{d}}{\mathrm{d}t}\right)^r \left(t^{\nu+1} e^t\right)\Big|_{t=x} \\
&= \sum_{r=0}^{n} \binom{n}{r} (-xy)^{n-r} e^y \left(t^2 \frac{\mathrm{d}}{\mathrm{d}t}\right)^r \left(t^{\nu+1} e^t\right)\Big|_{t=x} \\
&= \sum_{r=0}^{n} \binom{n}{r} (-xy)^{n-r} e^y \sum_{i=0}^{r} A_r^i t^{\nu+1+i+r} \frac{\mathrm{d}^i}{\mathrm{d}t^i} e^t\Big|_{t=x} \\
&= \sum_{r=0}^{n} \binom{n}{r} (-xy)^{n-r} e^y \sum_{i=0}^{r} A_r^i x^{\nu+1+i+r} e^x,
\end{aligned} \tag{26}$$

which upon further simplification leads to:

$$\tilde{\mathcal{D}}_n(x,y,\nu) = (-xy)^n x^{\nu+1} e^{x+y} \sum_{r=0}^{n} \binom{n}{r} (-y)^{-r} \sum_{i=0}^{r} A_r^i x^i. \tag{27}$$

In a similar manner, we develop $\tilde{\mathcal{N}}_n(x,y,\nu)$ by using the Leibniz product rule and the Slevinsky–Safouhi formula I with $(\mu,\nu,m,n) = (-2,\nu-1,0,0)$:

$$\tilde{\mathcal{N}}_n(x,y,\nu) = \frac{e^{-x-y}}{x^\nu y} \sum_{r=1}^{n} \binom{n}{r} \mathcal{D}_{n-r}(x,y,\nu)(xy)^r \sum_{s=0}^{r-1} \binom{r-1}{s} y^{-s} \sum_{i=0}^{s} A_s^i (-x)^i. \tag{28}$$

The coefficients A_r^i in (27) and A_s^i in (28) are given by Eq. (8).

Our approximations to $K_\nu(x,y)$ take the form:

$$\tilde{G}_n^{(1)}(x,y,\nu) = x^\nu \frac{\tilde{\mathcal{N}}_n(x,y,\nu)}{\tilde{\mathcal{D}}_n(x,y,\nu)}. \tag{29}$$

5. Numerical discussion

In [14], four numerical cases are presented and evaluated using a multiplicity of methods. With the approximations $\tilde{G}_n^{(1)}(x,y,\nu)$, we are capable of replicating these cases to the same pre-determined accuracy of $\pm 1 \times 10^{-10}$. We also show results obtained with an accuracy of $\pm 1 \times 10^{-15}$ for which higher order transformations are required to achieve the higher

Table 1
Numerical results.

x	y	ν	n	$\tilde{G}_n^{(1)}(x,y,\nu)$	Error	n	$\tilde{G}_n^{(1)}(x,y,\nu)$	Error
0.01	4	0	10	0.2225310761289636(1)	0.57(−10)	21	0.2225310761266469(1)	0.88(−15)
0.01	4	1	7	0.2138941668493954(0)	0.96(−10)	17	0.2138941668229403(0)	0.36(−15)
0.01	4	2	5	0.5450346981126452(−1)	0.78(−10)	13	0.5450346979970107(−1)	0.31(−15)
0.01	4	3	6	0.2325312150773913(−1)	0.21(−11)	9	0.2325312150770776(−1)	0.75(−15)
0.01	4	4	7	0.1304275099607653(−1)	0.21(−11)	9	0.1304275099607964(−1)	0.13(−16)
0.01	4	5	8	0.8567534990653542(−2)	0.31(−11)	10	0.8567534990648645(−2)	0.52(−17)
0.01	4	6	9	0.6208676806589944(−2)	0.66(−11)	11	0.6208676806600737(−2)	0.10(−16)
0.01	4	7	10	0.4801085238209789(−2)	0.19(−10)	12	0.4801085238177460(−2)	0.34(−16)
0.01	4	8	11	0.3884072049500670(−2)	0.72(−10)	13	0.3884072049626805(−2)	0.11(−15)
0.01	4	9	13	0.3246798003147811(−2)	0.62(−12)	14	0.3246798003148393(−2)	0.58(−15)
4.95	5	2	16	0.1224999251036423(−4)	0.27(−10)	22	0.1224998799706337(−4)	0.63(−15)
10.0	2	6	4	0.4150010642122851(−6)	0.29(−10)	10	0.4150045941916255(−6)	0.17(−15)
3.1	2.6	5	12	0.5285042839881951(−3)	0.62(−10)	27	0.5285043252436447(−3)	0.49(−15)

Table 2
Numerical results.

x	y	ν	n	$\tilde{G}_n^{(1)}(x,y,\nu)$	Error	n	$\tilde{G}_n^{(1)}(x,y,\nu)$	Error
1	1	8	20	0.1642584144215054(−1)	0.91(−10)	48	0.1642584157597500(−1)	0.56(−15)
1	1	16	13	0.8393633398897751(−2)	0.55(−10)	38	0.8393633437083270(−2)	0.68(−15)
5	5	4	8	0.8224340656934549(−5)	0.63(−10)	18	0.8224363011631705(−5)	0.46(−15)
5	5	8	6	0.5034017760297062(−5)	0.71(−10)	16	0.5034054653465547(−5)	0.60(−15)
5	5	16	5	0.2737340663308483(−5)	0.99(−10)	13	0.2737360566898996(−5)	0.71(−15)
10	1	16	4	0.6565373623179823(−6)	0.39(−10)	9	0.6565409733529793(−6)	0.69(−15)
10	5	16	3	0.1410714584838518(−7)	0.15(−10)	8	0.1410826247065302(−7)	0.22(−15)
10	10	16	2	0.1201285853568483(−9)	0.11(−11)	6	0.1204845014455500(−9)	0.20(−16)
1	5	1.6	8	0.4064821996750524(−2)	0.43(−10)	18	0.4064821958669517(−2)	0.40(−15)
1	10	2.1	5	0.2137545203050621(−3)	0.28(−10)	9	0.2137545215106365(−3)	0.56(−15)
5	10	3.5	8	0.1419594335970586(−6)	0.57(−10)	16	0.1419478426782529(−6)	0.17(−15)
0.1	0.1	16	16	0.5113063330630686(−1)	0.74(−10)	37	0.5113063337908691(−1)	0.93(−15)
0.5	0.5	12	18	0.3044667045008242(−1)	0.85(−10)	46	0.3044667055799152(−1)	0.32(−15)

pre-determined accuracy. For the region where $x \geqslant y$, we use the approximations $\tilde{G}_n^{(1)}(x,y,\nu)$ straightforwardly, and for the region where $x < y$, we use the inversion formula (4) and compute the approximations $\tilde{G}_n^{(1)}(y,x,-\nu)$ for the incomplete Bessel function $K_{-\nu}(y,x)$ and compute $K_\nu(2u)$ with the subroutine mikv.for from [17]. In Tables 1 and 2, we show the input variables x and y and parameter ν, the maximal order n of the transformation required, the corresponding approximation $\tilde{G}_n^{(1)}(x,y,\nu)$ of our FORTRAN 77 program, along with an approximation to the absolute error:

$$\text{Error} = \left|\tilde{G}_n^{(1)}(x,y,\nu) - \tilde{G}_{n-1}^{(1)}(x,y,\nu)\right|. \tag{30}$$

The four cases in [14] are:

Case 1. $x = 0.01,\ y = 4.00,\ \nu = 0(1)9$. We use (4) to invert x and y.

Case 2. $x = 4.95,\ y = 5.00,\ \nu = 2$. We again use (4) to invert x and y.

Case 3. $x = 10,\ y = 2,\ \nu = 6$. For this case, (4) is unnecessary, as $x > y$. We produce $K_6(10,2) = 0.00000\ 04150\ 01064\ 21228\ 51$, which is different from the "Accurate Value" $K_6(10,2) = 0.00023\ 44186\ 32699$ and the "Research of [14]" $K_6(10,2) = 0.00023\ 44186\ 19816$. However, we suspect that there is a typographical error in [14] as numerical integration with Maple gives the: "Accurate Value" $K_6(10,2) = 0.00000\ 04150\ 04594\ 23189\ 99$. Evidently, there is a disagreement between even the two accurate values, which leads us to suspect that the output in [14] does not correspond with $K_6(10,2)$. Our approximation has an absolute error less than 10^{-10} with $n = 4$ and less than 10^{-15} with $n = 10$.

Case 4. $x = 3.1,\ y = 2.6,\ \nu = 5$.

Table 2 corresponds to a new table of values that we have compiled. This table shows the regions where the approximations $\tilde{G}_n^{(1)}(x,y,\nu)$ perform well, and also where a high order transformation is required to attain the desired pre-determined accuracy.

In Table 2, our approximations $\tilde{G}_n^{(1)}(x,y,\nu)$ are demonstrated on a wide range of x, y and ν. Simple values to approximate include rows 6, 7 and 8, where x and y are large. Challenging values to approximate include rows 1, 2, 12 and 13, where x and y are small. We note that in rows 9, 10 and 11, values of ν are real and non-integer. Eqs. (28) and (27)

reveal that the computational complexity of the approximations $\tilde{G}_n^{(1)}(x, y, \nu)$ is independent of ν, which allows for an evaluation of $K_\nu(x, y)$ with real-, or even complex-, valued ν. In Table 2, we emphasize large values of ν. This is because the recurrence relation (2) is more stable in the downward direction for $x \geqslant y$. Therefore, since the values of ν vary from 0 to 16, the computation procedure would be more stable starting at the maximal ν and recurring downwards to maintain the pre-determined accuracy.

6. Conclusion

In this article, we confront the challenging problem of incomplete Bessel function computation. Through the necessity of an algorithm for their computation, we have conceived the much-anticipated recursive algorithm for the $G_n^{(1)}$ transformation. This algorithm opens the door to the numerical application of the $G_n^{(1)}$ transformation to other types of integral functions whose integrands satisfy first order linear homogeneous differential equations. Our algorithm allows, with the help of the Slevinsky–Safouhi formula I for higher order derivatives, for rapid evaluation of high-order $G_n^{(1)}$ transformations, which are required for evaluating integrals to high pre-determined accuracies. We replicate the four numerical cases treated in [14] in Table 1 to pre-determined accuracies of $\pm 1 \times 10^{-10}$ and $\pm 1 \times 10^{-15}$. In Table 2, we compile new values to assess the performance of our algorithm. In this table, our algorithm computes incomplete Bessel functions again to the pre-determined accuracies of $\pm 1 \times 10^{-10}$ and $\pm 1 \times 10^{-15}$.

Acknowledgements

The corresponding author (H.S.) acknowledges the financial support for this research by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] M. Abramowitz, I.A. Stegun, Handbook of Mathematical Functions, Dover, New York, 1965.
[2] M.M. Agrest, M.S. Maksimov, Theory of Incomplete Cylinder Functions and Their Applications, Springer, Philadelphia, 1971.
[3] C. Brezinski, M. Redivo-Zaglia, Extrapolation Methods: Theory and Practice, North-Holland, Amsterdam, 1991.
[4] D.C. Chang, R.J. Fisher, A unified theory on radiation of a vertical electric dipole above a dissipative earth, Radio Sci. 9 (1974) 1129–1138.
[5] M.A. Chaudhry, S.M. Zubair, Generalized incomplete gamma functions with applications, J. Comput. Appl. Math. 55 (1994) 99–124.
[6] M.A. Chaudhry, N.M. Temme, E.J.M. Veling, Asymptotics and closed form of a generalized incomplete gamma function, J. Comput. Appl. Math. 67 (1996) 371–379.
[7] S.L. Dvorak, Applications for incomplete Lipschitz–Hankel integrals in electromagnetics, IEEE Antennas Propagat. Mag. 36 (1994) 26–32.
[8] P.P. Ewald, Die Berechnung optischer und elektrostatischer Gitterpotentiale, Ann. Physik. 64 (1921) 253–287.
[9] H.L. Gray, S. Wang, A new method for approximating improper integrals, SIAM J. Numer. Anal. 29 (1992) 271–283.
[10] M.S. Hantush, C.E. Jacob, Non-steady radial flow in an infinite leaky aquifer, Trans. Amer. Geophys. Union 36 (1955) 95–100.
[11] F.E. Harris, New approach to calculation of the leaky aquifer function, Int. J. Quantum Chem. 63 (1997) 913–916.
[12] F.E. Harris, More about the leaky aquifer function, Int. J. Quantum Chem. 70 (1998) 623–626.
[13] F.E. Harris, On Kryachko's formula for the leaky aquifer function, Int. J. Quantum Chem. 81 (2001) 332–334.
[14] F.E. Harris, Incomplete Bessel, generalized incomplete gamma, or leaky aquifer functions, J. Comput. Appl. Math. 215 (2008) 260–269.
[15] F.E. Harris, J.G. Fripiat, Methods for incomplete Bessel evaluation, J. Comput. Appl. Math. 109 (2009) 1728–1740.
[16] B. Hunt, Calculation of the leaky aquifer function, J. Hydrol. 33 (1977) 179–183.
[17] J. Jin, FORTRAN routines for computation of Special Functions, from MathWorld – A Wolfram Web Resource, http://jin.ece.uiuc.edu/routines/routines.html.
[18] D.S. Jones, Incomplete Bessel functions. I, Proc. Edinburgh Math. Soc. 50 (2007) 173–183.
[19] D.S. Jones, Incomplete Bessel functions. II. Asymptotic expansions for large argument, Proc. Edinburgh Math. Soc. 50 (2007) 711–723.
[20] E.S. Kryachko, Explicit expression for the leaky aquifer function, Int. J. Quantum Chem. 78 (2000) 303–305.
[21] D. Levin, A. Sidi, Two new classes of non-linear transformations for accelerating the convergence of infinite integrals and series, Appl. Math. Comput. 9 (1981) 175–215.
[22] L. Lewin, The near field of a locally illuminated diffracting edge, IEEE Trans. Antennas Propagat. 19 (1971) 134–136.
[23] J. McWilliams, J. Balusek, H.L. Gray, Approximation of tail probabilities using the G_n-transform, Comm. Statist. Simulation Comput. 33 (2004) 1–17.
[24] M.M. Mechaik, S.L. Dvorak, Exact closed form field expressions for a semiinfinite traveling-wave current filament in homogeneous space, J. Electromagn. Waves Appl. 8 (1994) 1563–1584.
[25] A. Sidi, Extrapolation methods for oscillating infinite integrals, J. Inst. Math. Appl. 26 (1980) 1–20.
[26] A. Sidi, Practical Extrapolation Methods: Theory and Applications, Cambridge Univ. Press, Cambridge, 2003.
[27] R.M. Slevinsky, H. Safouhi, Numerical treatment of a twisted tail using extrapolation methods, Numer. Algorithms 48 (2008) 301–316.
[28] R.M. Slevinsky, H. Safouhi, The S and G transformations for computing three-center nuclear attraction integrals, Int. J. Quantum Chem. 109 (2009) 1741–1747.
[29] R.M. Slevinsky, H. Safouhi, New formulae for higher order derivatives and applications, J. Comput. Appl. Math. 233 (2009) 405–419.
[30] G.W. Thomas, R.G. Keys, A.C. Reynolds Jr., The computation of leaky aquifer functions, J. Hydrol. 36 (1978) 173–178.