A recursive algorithm for an efficient and accurate computation of incomplete Bessel functions

Richard M. Slevinsky1 · Hassan Safouhi2

Received: 25 March 2022 / Accepted: 9 October 2022 / Published online: 5 November 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
In a previous work, we developed an algorithm for the computation of incomplete Bessel functions, which pose as a numerical challenge, based on the $G_n^{(1)}$ transformation and Slevinsky-Safouhi formula for differentiation. In the present contribution, we improve this existing algorithm for incomplete Bessel functions by developing a recurrence relation for the numerator sequence and the denominator sequence whose ratio forms the sequence of approximations. By finding this recurrence relation, we reduce the complexity from $O(n^4)$ to $O(n)$. We plot relative error showing that the algorithm is capable of extremely high accuracy for incomplete Bessel functions.

Keywords Incomplete Bessel functions · Extrapolation methods · The $G$ transformation · Numerical integration · The Slevinsky-Safouhi formulae

Mathematical subject classification (2010) 65B05 · 65D30

1 Introduction
Incomplete Bessel functions, which are a computational challenge, were a subject of significant research. These functions appear when describing a plethora of phenomena in hydrology, statistics, and quantum mechanics [1–9]. Incomplete Bessel functions of zero order are also involved in a numerous applications to electromagnetic waves [10–14]. By introducing a recursive algorithm for the $G$ transformation
for tail integrals and applying it to incomplete Bessel functions, the article [15] has received some attention recently [16–21]. Most of this attention has been due to the algorithm for incomplete Bessel functions.

Integral representations of the incomplete Bessel functions are given by [22]:

\[ K_v(x, y) = \int_1^\infty \frac{e^{-x t - y/t}}{t^{v+1}} \, dt, \]  

(1)

The algorithm for the G transformation [23–26] applied to incomplete Bessel functions [15] appears as:

\[ \tilde{G}^{(1)}_n(x, y, \nu) = x^\nu \tilde{N}_n(x, y, \nu) D_n(x, y, \nu), \]  

(2)

where:

\[ D_n(x, y, \nu) = (-xy)^n e^{x+y} \sum_{r=0}^{n} \binom{n}{r} (-y)^{-r} \sum_{i=0}^{r} A_i^r x^i, \]  

(3)

where the \( A_i^r \) are the coefficients in the Slevinsky-Safouhi formula I [27] and are given by:

\[ A_i^k = \begin{cases} 
1 & \text{for } i = k, \\
(n - \nu - (k - 1)(\mu + 1))A_{k-1}^0 & \text{for } i = 0, k > 0, \\
(n - \nu + i(m + 1) - (k - 1)(\mu + 1))A_{k-1}^i + A_{k-1}^{i-1} & \text{for } 0 < i < k,
\end{cases} \]  

(4)

with \((\mu, \nu, m, n) = (-2, -\nu - 1, 0, 0)\).

The numerator \( \tilde{N}_n(x, y, \nu) \) in (2) is given by:

\[ \tilde{N}_n(x, y, \nu) = \frac{e^{-x y}}{x^{\nu} y^{\nu}} \sum_{r=1}^{n} \binom{n}{r} D_{n-r}(x, y, \nu) (xy)^r \sum_{s=0}^{r-1} \binom{r-1}{s} y^{-s} \sum_{i=0}^{s} A_i^s (-x)^i, \]  

(5)

where the \( A_i^s \) are the coefficients in the Slevinsky-Safouhi formula I with \((\mu, \nu, m, n) = (-2, \nu - 1, 0, 0)\).

Therefore, the original algorithm involves sums over sums over sums, effectively making the complexity of the computation of the numerator an \( \mathcal{O}(n^3) \) process and the complexity of the computation of the denominator an \( \mathcal{O}(n^2) \) process. Therefore, calculating a sequence of approximations \( \{ G_n \}_{n \geq 0} \) results in an \( \mathcal{O}(n^4) \) algorithm.

In the following, we introduce an algorithm for the G transformation that reduces the calculation to an easily programmable and parallel four-term recurrence relation: with one initialization, it computes the numerator; with another initialization, it computes the denominator. As there are only a finite number of arithmetic operations required in the recurrence relation, the resulting algorithm for calculating a sequence of approximations \( \{ G_n \}_{n \geq 0} \) is an \( \mathcal{O}(n) \) algorithm.
The inductive proof of this recurrence relation follows and the notation is heavy because of the two variables $x$ and $y$ and the parameter $\nu$ already included in the incomplete Bessel function. As well, with this new algorithm comes the ability to program the $G$ transformation to unprecedentedly high order. We show some numerical results in the form of figures of the relative error for six different values. The reduction in complexity of the original algorithm comes with the benefit of a stable algorithm for high order.

2 The algorithm

**Theorem 1** [28] Let $f(x)$ be integrable on $[0, \infty)$ (i.e., $\int_0^\infty f(t) \, dt < \infty$) 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:

$$p(x) \sim x^{i_k} \sum_{i=0}^{\infty} \frac{a_i}{x^i} \quad \text{with} \quad i_k \leq k.$$  \hfill (6)

If for $1 \leq i \leq m$ and $i \leq k \leq m$, we have:

$$\lim_{x \to \infty} p_k^{(i-1)}(x)f^{(k-i)}(x) = 0,$$  \hfill (7)

and for every integer $l \geq -1$, we have:

$$\sum_{k=1}^{m} l(l-1) \cdots (l-k+1)p_{k,0} \neq 1,$$  \hfill (8)

where $p_{k,0} = \lim_{x \to \infty} x^{-k} p_k(x)$ for $1 \leq k \leq m$, then as $x \to \infty$, we have:

$$\int_x^\infty f(t) \, dt \sim \sum_{k=0}^{m-1} x^{j_k} f^{(k)}(x) \left( \beta_{0,k} + \frac{\beta_{1,k}}{x} + \frac{\beta_{2,k}}{x^2} + \cdots \right).$$  \hfill (9)

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

To solve for the unknowns $\beta_{k,j}$ we must set up and solve a system of linear equations.

The $G_n^{(m)}$ transformation [23] truncates the asymptotic expansions (10) after $n$ terms each and the system is formed by differentiation. The approximation $G_n^{(m)}$ to $\int_0^\infty f(t) \, dt$ is given as the solution of the system of $mn + 1$ linear equations [23]:

\[ \square \] Springer
\[
\frac{d^l}{dx^l} \left\{ G_n^{(m)} - \int_0^x f(t) \, dt - \sum_{k=0}^{m-1} x^{\sigma_k} f^{(k)}(x) \sum_{i=0}^{n-1} \frac{\beta_{k,i}}{x^i} \right\} = 0, \quad l = 0, 1, \ldots, mn, 
\]

where it is assumed that \( \frac{d^l}{dx^l} G_n^{(m)} \equiv 0, \forall l > 0 \). In the above system (11), \( \sigma_k = \min(s_k, k + 1) \) 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 \( \beta_{k,i} \) are the respective set of \( mn + 1 \) unknowns.

The \( G_n^{(1)} \) transformation can be written as the solution to the linear system (11) 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 [15].

By considering the (11) for \( l = 0 \):

\[
G_n^{(1)} - F(x) = x^{\sigma_0} f(x) \sum_{i=0}^{n-1} \frac{\beta_{0,i}}{x^i} \quad \text{with} \quad F(x) = \int_0^x f(t) \, dt,
\]

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

\[
\frac{G_n^{(1)} - F(x)}{x^{\sigma_0} f(x)} = \sum_{i=0}^{n-1} \frac{\beta_{0,i}}{x^i}. \tag{13}
\]

To eliminate the summation, and consequently all of the unknowns \( \beta_{0,i} \), we apply the \( \left( x^2 \frac{d}{dx} \right) \) operator \( n \) times, obtaining:

\[
\left( x^2 \frac{d}{dx} \right)^n \left[ \frac{G_n^{(1)} - F(x)}{x^{\sigma_0} f(x)} \right] = 0 \quad \implies \quad G_n^{(1)} = \left( x^2 \frac{d}{dx} \right)^n \left( \frac{F(x)}{x^{\sigma_0} f(x)} \right) \left( \frac{1}{x^{\sigma_0} f(x)} \right), \tag{14}
\]

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

1. Set:
   \[
   \mathcal{N}_0(x) = \frac{F(x)}{x^{\sigma_0} f(x)} \quad \text{and} \quad \mathcal{D}_0(x) = \frac{1}{x^{\sigma_0} f(x)}. \tag{15}
   \]

2. For \( n = 1, 2, \ldots \), compute \( \mathcal{N}_n(x) \) and \( \mathcal{D}_n(x) \) recursively from:
   \[
   \mathcal{N}_n(x) = \left( x^2 \frac{d}{dx} \right) \mathcal{N}_{n-1}(x) \quad \text{and} \quad \mathcal{D}_n(x) = \left( x^2 \frac{d}{dx} \right) \mathcal{D}_{n-1}(x). \tag{16}
   \]

3. For all \( n \), the approximations \( G_n^{(1)}(x) \) to \( \left( \int_0^x + \int_x^\infty \right) f(t) \, dt \) are given by:
   \[
   G_n^{(1)}(x) = \frac{\mathcal{N}_n(x)}{\mathcal{D}_n(x)}. \tag{17}
   \]
For the incomplete Bessel functions, \( \sigma_0 = 0 \), and the algorithm for the \( G^{(1)}_n \) transformation takes the form below. Let:

\[
f(t) = \frac{e^{-xt-y/t}}{t^{v+1}} \quad \text{and} \quad F(t) = \int_0^t f(s) \, ds.
\] (18)

1. Set:

\[
\mathcal{N}_n(x, y, \nu, t) = \frac{F(t)}{f(t)} \quad \text{and} \quad \mathcal{D}_n(x, y, \nu, t) = \frac{1}{f(t)}.
\] (19)

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

\[
\mathcal{N}_n(x, y, \nu, t) = \left( t^2 \frac{d}{dt} \right) \mathcal{N}_{n-1}(x, y, \nu, t)
\]
\[
\mathcal{D}_n(x, y, \nu, t) = \left( t^2 \frac{d}{dt} \right) \mathcal{D}_{n-1}(x, y, \nu, t).
\] (20)

3. For all \( n \geq 0 \), the approximations \( \tilde{G}^{(1)}_n(x, y, \nu) \) to \( K_v(x, y) \) are given by:

\[
\tilde{G}^{(1)}_n(x, y, \nu) = \frac{\tilde{\mathcal{N}}_n(x, y, \nu, 1)}{\tilde{\mathcal{D}}_n(x, y, \nu, 1)},
\] (21)

where:

\[
\tilde{\mathcal{N}}_n(x, y, \nu, t) = \mathcal{N}_n(x, y, \nu, t) - F(t)\mathcal{D}_n(x, y, \nu, t).
\] (22)

**Proposition 1** Let:

\[
\tilde{\mathcal{N}}_0(x, y, \nu) = 0 \quad \text{and} \quad \tilde{\mathcal{N}}_1(x, y, \nu) = 1,
\] (23)

\[
\mathcal{D}_0(x, y, \nu) = e^{x+y} \quad \text{and} \quad \mathcal{D}_1(x, y, \nu) = (x + v + 1 - y)\mathcal{D}_0(x, y, \nu),
\] (24)

and:

\[
\tilde{\mathcal{N}}_{-1}(x, y, \nu) = \mathcal{D}_{-1}(x, y, \nu) = 0.
\] (25)

Let also:

\[
(n + 1)Q_{n+1}(x, y, \nu) = (x + v + 1 + 2n - y)Q_n(x, y, \nu)
\]
\[
+ (2y - v - n)Q_{n-1}(x, y, \nu) - yQ_{n-2}(x, y, \nu),
\] (26)

where the \( Q_n(x, y, \nu) \) stand for either the \( \tilde{\mathcal{N}}_n(x, y, \nu) \) or the \( \mathcal{D}_n(x, y, \nu) \).

Then:
\[ \tilde{G}^{(1)}_n(x, y, \nu) = \frac{\tilde{N}_n(x, y, \nu)}{D_n(x, y, \nu)}. \]  

**Proof** In the following, \( Q_n(x, y, \nu) \) stand for either the \( \tilde{N}_n(x, y, \nu) \) or the \( D_n(x, y, \nu) \) and \( \tilde{Q}_n(x, y, \nu, t) \) stand for either the \( \tilde{N}'_n(x, y, \nu, t) \) or the \( D_n(x, y, \nu, t) \).

Let \( D_n(t) = D_n(x, y, \nu, t) \) for short and let \( \mathcal{D}_{-2}(t) = D_{-1}(t) = 0 \). The sequence \( \{D_n(t)\}_{n \geq 0} \) is generated by the one-term recurrence:

\[ D_n(t) = \left( t^2 \frac{d}{dt} \right) D_{n-1}(t). \]  

Then, we show that \( D_n(t) \) satisfies, for \( n \geq 0 \):

\[ D_{n+1}(t) = (x t^2 + (v + 1 + 2n) t - y) D_n(t) + (2n t y - n(n - 1) t^2 - n(v + 1)t^2) D_{n-1}(t) - n(n - 1)t^2 y D_{n-2}(t). \]  

For \( n = 0 \):

\[ D_1(t) = \left( t^2 \frac{d}{dt} \right) D_0(t) = (x t^2 + (v + 1)t - y) D_0(t). \]  

The induction argument assumes:

\[ D_{k+1}(t) = (x t^2 + (v + 1 + 2k) t - y) D_k(t) + (2k t y - k(k - 1) t^2 - k(v + 1)t^2) D_{k-1}(t) - k(k - 1) t^2 y D_{k-2}(t). \]  

Differentiating and multiplying by \( t^2 \), we obtain:

\[ D_{k+2}(t) = (x t^2 + (v + 1 + 2k) t - y) D_{k+1}(t) + (2x t^3 + (v + 1 + 2k)t^2 + 2k t y - k(k - 1) t^2 - k(v + 1)t^2) D_k(t) + (2k t^2 y - 2k(k - 1) t^3 - 2k(v + 1)t^3 - k(k - 1)t^2 y) D_{k-1}(t) - 2k(k - 1) t^3 y D_{k-2}(t). \]  

Multiplying (31) by \( 2t \) and subtracting it from (32), we obtain, after simplification:

\[ D_{k+2}(t) = (x t^2 + (v + 1 + 2(k + 1)) t - y) D_{k+1}(t) + (2(k + 1) t y - k(k + 1)t^2 - (k + 1)(v + 1)t^2) D_k(t) - k(k + 1) t^2 y D_{k-1}(t). \]  

which proves (29) by induction.

As with the denominator, let \( \mathcal{N}'_n(t) = \mathcal{N}'_n(x, y, \nu, t) \) and \( \tilde{\mathcal{N}}_n(t) = \tilde{\mathcal{N}}_n(x, y, \nu, t) \) for short.

Since \( \mathcal{N}'_0(t) = F(t) D_0(t) \):

\[ \mathcal{N}'_0(t) = F(t) D_0(t). \]
\[ \mathcal{N}_1(t) = F(t) \mathcal{D}_1(t) + t^2 f(t) \mathcal{D}_0(t) = F(t) \mathcal{D}_1(t) + t^2. \]  

But we now write \( F(t) = \frac{N_0(t)}{D_0(t)} \) to conclude that:

\[ \mathcal{N}_1(t) = (t^2 x + (\nu + 1)t - y)N_0(t) + t^2. \]  

(35)

Differentiating and multiplying by \( t^2 \), we obtain:

\[ \mathcal{N}_2(t) = (t^2 x + (\nu + 1)t - y)\mathcal{N}_1(t) + (2x t^3 + (\nu + 1)t^2)N_0(t) + 2t^3. \]  

(36)

Multiplying (35) by \( 2t \) and subtracting it from (36), we obtain, after simplification:

\[ \mathcal{N}_2(t) = (t^2 x + (\nu + 3)t - y)\mathcal{N}_1(t) + (2yt - (\nu + 1)t^2)N_0(t). \]  

(37)

But this is just (29) for \( n = 1 \) with the labels \( \mathcal{N}_n(t) \) interchanged for \( \mathcal{D}_n(t) \). Any further differentiation and multiplication by \( t^2 \) will, therefore, ultimately lead to the same four-term recurrence relation, the difference being different initial conditions.

As a sequence:

\[ \{ \tilde{\mathcal{N}}_n(t) \}_{n \geq 0} = \{ \mathcal{N}_n(t) - F(t)\mathcal{D}_n(t) \}_{n \geq 0}, \]  

(38)

is a linear combination of both solutions \( \mathcal{N}_n(t) \) and \( \mathcal{D}_n(t) \).

Therefore, \( \tilde{\mathcal{N}}_n(t) \) satisfies (29) as well with the labels appropriately interchanged.

To complete the proof, we must return to the original sequences:

\[ \{ \tilde{\mathcal{N}}_n(x, y, \nu) \}_{n \geq 0} \quad \text{and} \quad \{ \mathcal{D}_n(x, y, \nu) \}_{n \geq 0}. \]  

(39)

Indeed, the relationship is that:

\[ Q_n(x, y, \nu) = \frac{Q_n(x, y, \nu, 1)}{n!}, \]  

(40)

where the \( Q_n(x, y, \nu) \) stand for either the \( \tilde{\mathcal{N}}_n(x, y, \nu) \) or the \( \mathcal{D}_n(x, y, \nu) \) and the \( Q_n(x, y, \nu, t) \) stand for either the \( \tilde{\mathcal{N}}_n(x, y, \nu, t) \) or the \( \mathcal{D}_n(x, y, \nu, t) \).

### 3 Figures

Figures 1 and 2 show contour plots of an approximation to relative error of the \( G \) transformation:

\[ \text{Relative Error} = \left| \frac{K(x, y) - \tilde{G}^{(1)}(x, y, \nu)}{K(x, y)} \right|. \]

for \( n = 15 \) and \( n = 30 \), respectively. The grid sampling to produce the contour plots is \( x = 0.25, 0.5, \ldots, 10 \) and \( y = -10, -9.5, \ldots, 10 \), numbers that are exactly
representable in binary. In both figures, the top rows show the relative error when computing the $G$ transformation via recurrence relations in extended (256-bit) precision, while the bottom rows show the relative error when the computations are performed in double precision. To preserve a common scale across all contour plots, the relative error is clamped above machine epsilon in double precision, $2^{-63} \approx 2.22 \times 10^{-16}$, and below 1.

Figure 3 shows contour plots for the order of the $G$ transformation for an approximation to $K_\nu(x, y)$ that is obtained when successive transformations differ relatively on the order of machine precision. We choose to illustrate the (integer)
orders on a base-10 log-scale because of the sharp increase in order for small $x$ and large $|y|$.

There are a number of ways to compute approximate values of $K_n(x, y)$ which may be sufficiently accurate for the purposes of Figs. 1, 2, and 3. For example, a Gauss–Laguerre or double exponential quadrature may be used. For simplicity, we used the $G$ transformation computed in extended (256-bit) precision, with a machine epsilon of $2^{-255} \approx 1.73 \times 10^{-77}$.

Heuristically, we qualify the recurrence relations as stable since in Figs. 1 and 2, there are only subtle differences between the resulting computations performed in double precision vis-à-vis those in extended precision. These subtle differences are all at the level of machine epsilon; the rough nature of the contours of relative errors around machine epsilon is due to floating-point rounding errors dominating the truncation error of the algorithm. Moreover, the stability of the $G$ transformation is also qualified by the fact that the region with relative error on the order of machine epsilon increases in size from Figs. 1 to 2 and the rounding errors in the double precision plots do not increase the relative errors by more than one level in the contour plots. When rounding errors do not grow too rapidly with order $n$, the derivation of stopping criteria is rather straightforward.

While Figs. 1 and 2 suggest that small $x$ and large $|y|$ evaluation of $K_n(x, y)$ requires more computational effort, the stability of the algorithm also suggests that good approximations are attainable nevertheless. This is shown in Fig. 3, where the order for machine epsilon is substantially higher in this region.
4 Conclusion

We improve an existing algorithm for incomplete Bessel functions based on the $G$ transformation [15] by developing a recurrence relation the numerator sequence and the denominator sequence whose ratio form the sequence of approximations. By finding this recurrence relation, we reduce the complexity from $O(n^4)$ to $O(n)$. We plot relative error showing that the algorithm is capable of extremely high accuracy for incomplete Bessel functions. The stability and convergence appear to be remarkable.

Funding HS acknowledges the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) — Grant RGPIN-2016-04317. RMS acknowledges the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) — Grant RGPIN-2017-05514.

Declarations Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Conflict of interest The authors declare no competing interests.

References

1. Jones, D.S.: Incomplete Bessel functions. i. Proc. Edinb. Math. Soc. 50, 173–183 (2007)
2. Jones, D.S.: Incomplete Bessel functions. II. Asymptotic expansions for large argument. Proc. Edinb. Math. Soc. 50, 711–723 (2007)
3. Hantush, M.S., Jacob, C.E.: Non-steady radial flow in an infinite leaky aquifer. Trans. Amer. Geophys. Union. 36, 95–100 (1955)
4. Harris, F.E.: New approach to calculation of the leaky aquifer function. Int. J. Quantum Chem. 63, 913–916 (1997)
5. Harris, F.E.: More about the leaky aquifer function. Int. J. Quantum Chem. 70, 623–626 (1998)
6. Harris, F.E.: On Kryachko’s formula for the leaky aquifer function. Int. J. Quantum Chem. 81, 332–334 (2001)
7. Hunt, B.: Calculation of the leaky aquifer function. J. Hydrol. 33, 179–183 (1977)
8. Kryachko, E.S.: Explicit expression for the leaky aquifer function. Int. J. Quantum Chem. 78, 303–305 (2000)
9. Harris, F.E., Fripiat, J.G.: Methods for incomplete Bessel evaluation. J. Comp. Appl. Math. 109, 1728–1740 (2009)
10. Agrest, M.M., Maksimov, M.S.: Theory of Incomplete Cylinder Functions and Their Applications. Springer, Philadelphia (1971)
11. Lewin, L.: The near field of a locally illuminated diffracting edge. IEEE Trans. Antennas Propagat. AP19, 134–136 (1971)
12. Chang, D.C., Fisher, R.J.: A unified theory on radiation of a vertical electric dipole above a dissipative earth. Radio Sci. 9, 1129–1138 (1974)
13. Dvorak, S.L.: Applications for incomplete Lipschitz-Hankel integrals in electromagnetics. IEEE Antennas Propagat. Mag. 36, 26–32 (1994)
14. Mechaik, M.M., Dvorak, S.L.: Exact closed form field expressions for a semiinfinite traveling-wave current filament in homogeneous space. J. Electromagn. Waves Appl. 8, 1563–1584 (1994)
15. Slevinsky, R.M., Safouhi, H.: A recursive algorithm for the $G$ transformation and accurate computation of incomplete Bessel functions. App. Num. Math. 60, 1411–1417 (2010)
16. Gaudreau, P., Slevinsky, R.M., Safouhi, H.: Computation of tail probability distributions via extrapolation methods and connection with rational and Padé approximants. SIAM J. Sci. Comput. 34, B65–B85 (2012)
17. Scott, D., Tran, T., Slevinsky, R.M., Safouhi, H.: The incomplete Bessel K function in R Package DistributionUtils. http://cran.r-project.org/web/packages/DistributionUtils/ (2012)
18. Slevinsky, R.M., Safouhi, H.: Useful properties of the coefficients of the Slevinsky-Safouhi formula for differentiation. Numer. Algorithms 66, 457–477 (2014)
19. Creal, D.D.: Exact likelihood inference for autoregressive gamma stochastic volatility models. Research Seminar pp. 1–35 (2012)
20. Creal, D.D.: A class of non-Gaussian state space models with exact likelihood inference: Appendix. Working Paper pp. 1–28 (2013)
21. Nestler, F., Pippig, M., Potts, D.: Fast Ewald summation based on NFFT with mixed periodicity. Working Paper pp. 1–38 (2013)
22. Harris, F.E.: Incomplete Bessel, generalized incomplete gamma, or leaky aquifer functions. J. Comp. Appl. Math. 215, 260–269 (2008)
23. Gray, H.L., Wang, S.: A new method for approximating improper integrals. SIAM J. Numer. Anal. 29, 271–283 (1992)
24. Gray, H.L., Atchison, T.A.: Nonlinear transformation related to the evaluation of improper integrals. I. SIAM J. Numer. Anal. 4, 363–371 (1967)
25. Gray, H.L., Atchison, T.A., McWilliams, G.V.: Higher order G-transformations. SIAM J. Numer. Anal. 8, 365–381 (1971)
26. Joyce, D.C.: Survey of extrapolation processes in numerical analysis. SIAM Rev. 13, 435–490 (1971)
27. Slevinsky, R.M., Safouhi, H.: New formulae for higher order derivatives and applications. J. Comput. App. Math. 233, 405–419 (2009)
28. Levin, D., Sidi, A.: Two new classes of non-linear transformations for accelerating the convergence of infinite integrals and series. Appl. Math. Comput. 9, 175–215 (1981)

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.