NAG Library Chapter Introduction

s – Approximations of Special Functions

Contents

1 Scope of the Chapter

This chapter is concerned with the provision of some commonly occurring physical and mathematical functions.

2 Background to the Problems

The majority of the functions in this chapter approximate real-valued functions of a single real argument, and the techniques involved are described in Section 2.1. In addition the chapter contains functions for elliptic integrals (see Section 2.2), Bessel and Airy functions of a complex argument (see Section 2.3), complementary error function of a complex argument, hypergeometric functions and various option pricing functions for use in financial applications.

2.1 Functions of a Single Real Argument

Most of the functions provided for functions of a single real argument have been based on truncated Chebyshev expansions. This method of approximation was adopted as a compromise between the conflicting requirements of efficiency and ease of implementation on many different machine ranges. For details of the reasons behind this choice and the production and testing procedures followed in constructing this chapter see Schonfelder (1976).

Basically, if the function to be approximated is $f(x)$, then for $x \in [a, b]$ an approximation of the form

$$f(x) = g(x)\sum_{r=0} C_r T_r(t)$$

is used ($\sum$ denotes, according to the usual convention, a summation in which the first term is halved), where $g(x)$ is some suitable auxiliary function which extracts any singularities, asymptotes and, if possible, zeros of the function in the range in question and $t = t(x)$ is a mapping of the general range $[a, b]$ to the specific range $[-1, +1]$ required by the Chebyshev polynomials, $T_r(t)$. For a detailed description of the properties of the Chebyshev polynomials see Clenshaw (1962) and Fox and Parker (1968).

The essential property of these polynomials for the purposes of function approximation is that $T_n(t)$ oscillates between ± 1 and it takes its extreme values $n + 1$ times in the interval $[-1, +1]$. Therefore, provided the coefficients C_r decrease in magnitude sufficiently rapidly the error made by truncating the Chebyshev expansion after n terms is approximately given by

$$E(t) \simeq C_n T_n(t).$$

That is, the error oscillates between $\pm C_n$ and takes its extreme value $n + 1$ times in the interval in question. Now this is just the condition that the approximation be a minimax representation, one which minimizes the maximum error. By suitable choice of the interval, $[a, b]$, the auxiliary function, $g(x)$, and the mapping of the independent variable, $t(x)$, it is almost always possible to obtain a Chebyshev expansion with rapid convergence and hence truncations that provide near minimax polynomial approximations to the required function. The difference between the true minimax polynomial and the truncated Chebyshev expansion is seldom sufficiently great enough to be of significance.

The evaluation of the Chebyshev expansions follows one of two methods. The first and most efficient, and hence the most commonly used, works with the equivalent simple polynomial. The second method, which is used on the few occasions when the first method proves to be unstable, is based directly on the truncated Chebyshev series, and uses backward recursion to evaluate the sum. For the first method, a suitably truncated Chebyshev expansion (truncation is chosen so that the error is less than the ***machine precision***) is converted to the equivalent simple polynomial. That is, we evaluate the set of coefficients b_r such that

$$y(t) = \sum_{r=0}^{n-1} b_r t^r = \sum_{r=0}^{n-1} C_r T_r(t).$$

The polynomial can then be evaluated by the efficient Horner's method of nested multiplications,

$$y(t) = (b_0 + t(b_1 + t(b_2 + \ldots t(b_{n-2} + t b_{n-1}))) \ldots).$$

This method of evaluation results in efficient functions but for some expansions there is considerable loss of accuracy due to cancellation effects. In these cases the second method is used. It is well known that if

$$\begin{aligned} b_{n-1} &= C_{n-1} \\ b_{n-2} &= 2tb_{n-1} + C_{n-2} \\ b_j \quad &= 2tb_{j+1} - b_{j+2} + C_j, \quad j = n-3, n-4, \ldots, 0 \end{aligned}$$

then

$$\sum_{r=0} C_r T_r(t) = \tfrac{1}{2}(b_0 - b_2)$$

and this is always stable. This method is most efficiently implemented by using three variables cyclically and explicitly constructing the recursion.

That is,

$$\begin{aligned} \alpha &= C_{n-1} \\ \beta &= 2t\alpha + C_{n-2} \\ \gamma &= 2t\beta - \alpha + C_{n-3} \\ \alpha &= 2t\gamma - \beta + C_{n-4} \\ \beta &= 2t\alpha - \gamma + C_{n-5} \\ &\vdots \\ \text{say } \alpha &= 2t\gamma - \beta + C_2 \\ \beta &= 2t\alpha - \gamma + C_1 \\ y(t) &= t\beta - \alpha + \tfrac{1}{2}C_0 \end{aligned}$$

The auxiliary functions used are normally functions compounded of simple polynomial (usually linear) factors extracting zeros, and the primary compiler-provided functions, sin, cos, ln, exp, sqrt, which extract singularities and/or asymptotes or in some cases basic oscillatory behaviour, leaving a smooth well-behaved function to be approximated by the Chebyshev expansion which can therefore be rapidly convergent.

The mappings of $[a, b]$ to $[-1, +1]$ used range from simple linear mappings to the case when b is infinite, and considerable improvement in convergence can be obtained by use of a bilinear form of mapping. Another common form of mapping is used when the function is even; that is, it involves only even powers in its expansion. In this case an approximation over the whole interval $[-a, a]$ can be provided using a mapping $t = 2(x/a)^2 - 1$. This embodies the evenness property but the expansion in t involves all powers and hence removes the necessity of working with an expansion with half its coefficients zero.

For many of the functions an analysis of the error in principle is given, namely, if E and ∇ are the absolute errors in function and argument and ϵ and δ are the corresponding relative errors, then

$$\begin{aligned} E &\simeq |f'(x)|\nabla \\ E &\simeq |xf'(x)|\delta \\ \epsilon &\simeq \left|\frac{xf'(x)}{f(x)}\right|\delta. \end{aligned}$$

If we ignore errors that arise in the argument of the function by propagation of data errors, etc., and consider only those errors that result from the fact that a real number is being represented in the computer in floating-point form with finite precision, then δ is bounded and this bound is independent of the magnitude of x. For example, on an 11-digit machine

$$|\delta| \leq 10^{-11}.$$

(This of course implies that the absolute error $\nabla = x\delta$ is also bounded but the bound is now dependent on x.) However, because of this the last two relations above are probably of more interest. If possible the relative error propagation is discussed; that is, the behaviour of the error amplification factor $|xf'(x)/f(x)|$ is described, but in some cases, such as near zeros of the function which cannot be

extracted explicitly, absolute error in the result is the quantity of significance and here the factor $|xf'(x)|$ is described. In general, testing of the functions has shown that their error behaviour follows fairly well these theoretical error behaviours. In regions where the error amplification factors are less than or of the order of one, the errors are slightly larger than the above predictions. The errors are here limited largely by the finite precision of arithmetic in the machine, but ϵ is normally no more than a few times greater than the bound on δ. In regions where the amplification factors are large, of order ten or greater, the theoretical analysis gives a good measure of the accuracy obtainable.

It should be noted that the definitions and notations used for the functions in this chapter are all taken from Abramowitz and Stegun (1972). You are strongly recommended to consult this book for details before using the functions in this chapter.

2.2 Approximations to Elliptic Integrals

Four functions provided here are symmetrised variants of the classical (Legendre) elliptic integrals. These alternative definitions have been suggested by Carlson (1965), Carlson (1977b) and Carlson (1977a) and he also developed the basic algorithms used in this chapter.

The symmetrised elliptic integral of the first kind is represented by

$$R_F(x,y,z) = \frac{1}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)}},$$

where $x, y, z \geq 0$ and at most one may be equal to zero.

The normalization factor, $\frac{1}{2}$, is chosen so as to make

$$R_F(x,x,x) = 1/\sqrt{x}.$$

If any two of the variables are equal, R_F degenerates into the second function

$$R_C(x,y) = R_F(x,y,y) = \frac{1}{2}\int_0^\infty \frac{dt}{(t+y).\sqrt{t+x}},$$

where the argument restrictions are now $x \geq 0$ and $y \neq 0$.

This function is related to the logarithm or inverse hyperbolic functions if $0 < y < x$, and to the inverse circular functions if $0 \leq x \leq y$.

The symmetrised elliptic integral of the second kind is defined by

$$R_D(x,y,z) = \frac{3}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)^3}}$$

with $z > 0$, $x \geq 0$ and $y \geq 0$, but only one of x or y may be zero.

The function is a degenerate special case of the symmetrised elliptic integral of the third kind

$$R_J(x,y,z,\rho) = \frac{3}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)}(t+\rho)}$$

with $\rho \neq 0$ and $x, y, z \geq 0$ with at most one equality holding. Thus $R_D(x,y,z) = R_J(x,y,z,z)$. The normalization of both these functions is chosen so that

$$R_D(x,x,x) = R_J(x,x,x,x) = 1/\left(x\sqrt{x}\right).$$

The algorithms used for all these functions are based on duplication theorems. These allow a recursion system to be established which constructs a new set of arguments from the old using a combination of arithmetic and geometric means. The value of the function at the original arguments can then be simply related to the value at the new arguments. These recursive reductions are used until the arguments differ from the mean by an amount small enough for a Taylor series about the mean to give sufficient accuracy when retaining terms of order less than six. Each step of the recurrences reduces the difference from the mean by a factor of four, and as the truncation error is of order six, the truncation error goes like $(4096)^{-n}$, where n is the number of iterations.

The above forms can be related to the more traditional canonical forms (see Section 17.2 of Abramowitz and Stegun (1972)), as follows.

If we write $q = \cos^2\phi, r = 1 - m\,\sin^2\phi, s = 1 - n\,\sin^2\phi$, where $0 \leq \phi \leq \frac{1}{2}\pi$, we have

the classical elliptic integral of the first kind:

$$F(\phi \mid m) = \int_0^{\phi} \left(1 - m\,\sin^2\theta\right)^{-\frac{1}{2}} d\theta = \sin\phi\, R_F(q, r, 1);$$

the classical elliptic integral of the second kind:

$$\begin{aligned} E(\phi \mid m) &= \int_0^{\phi} \left(1 - m\,\sin^2\theta\right)^{\frac{1}{2}} d\theta \\ &= \sin\phi\, R_F(q, r, 1) - \tfrac{1}{3} m \sin^3\phi\, R_D(q, r, 1) \end{aligned}$$

the classical elliptic integral of the third kind:

$$\begin{aligned} \Pi(n;\phi \mid m) &= \int_0^{\phi} \left(1 - n\,\sin^2\theta\right)^{-1} \left(1 - m\,\sin^2\theta\right)^{-\frac{1}{2}} d\theta \\ &= \sin\phi\, R_F(q, r, 1) + \tfrac{1}{3} n \sin^3\phi\, R_J(q, r, 1, s). \end{aligned}$$

Also the classical complete elliptic integral of the first kind:

$$K(m) = \int_0^{\frac{\pi}{2}} \left(1 - m\,\sin^2\theta\right)^{-\frac{1}{2}} d\theta = R_F(0, 1 - m, 1);$$

the classical complete elliptic integral of the second kind:

$$E(m) = \int_0^{\frac{\pi}{2}} \left(1 - m\sin^2\theta\right)^{\frac{1}{2}} d\theta = R_F(0, 1 - m, 1) - \tfrac{1}{3} m\, R_D(0, 1 - m, 1).$$

For convenience, Chapter s contains functions to evaluate classical **and** symmetrised elliptic integrals.

2.3 Bessel and Airy Functions of a Complex Argument

The functions for Bessel and Airy functions of a real argument are based on Chebyshev expansions, as described in Section 2.1. The functions provided for functions of a complex argument, however, use different methods. These functions relate all functions to the modified Bessel functions $I_\nu(z)$ and $K_\nu(z)$ computed in the right-half complex plane, including their analytic continuations. I_ν and K_ν are computed by different methods according to the values of z and ν. The methods include power series, asymptotic expansions and Wronskian evaluations. The relations between functions are based on well known formulae (see Abramowitz and Stegun (1972)).

2.4 Option Pricing Functions

The option pricing functions evaluate the closed form solutions or approximations to the equations that define mathematical models for the prices of selected financial option contracts. These solutions can be viewed as special functions determined by the underlying equations. The terminology associated with these functions arises from their setting in financial markets and is briefly outlined below. See Joshi (2003) for a comprehensive introduction to this subject. An option is a contract which gives the holder the right, but not the obligation, to buy (if it is a call) or sell (if it is a put) a particular asset, S. A European option can be exercised only at the specified expiry time, T, while an American option can be exercised at any time up to T. For Asian options the average underlying price over a pre-set time period determines the payoff.

The asset is bought (if a call) or sold (if a put) at a pre-specified strike price X. Thus, an option contract has a payoff to the holder of $\max\{(S_T - X), 0\}$ for a call or $\max\{(X - S_T), 0\}$, for a put, which depends on whether the asset price at the time of exercise is above (call) or below (put) the strike, X. If at any moment in time a contract is currently showing a theoretical profit then it is deemed ‘in-the-money’; otherwise it is deemed ‘out-of-the-money’.

The option contract itself therefore has a value and, in many cases, can be traded in markets. Mathematical models (e.g., Black–Scholes, Merton, Vasicek, Hull–White, Heston, CEV, SABR, ...) give theoretical prices for particular option contracts using a number of assumptions about the behaviour of financial markets. Typically the price S_t of the underlying asset at time t is modelled as the solution of a stochastic differential equation (SDE). Depending on the complexity of this equation, the model may admit closed form formulae for the prices of various options. The options described in this chapter introduction are detailed below. We let $\mathbb{E}$ denote expectation with respect to the risk neutral measure and we define $\mathbb{I}_A$ to be 1 on the set A and 0 otherwise.

– The price of a standard European call option is $\mathbb{E}\big(e^{-rT}\max\{S_T - X, 0\}\big)$ and the price of a standard European put option is $\mathbb{E}\big(e^{-rT}\max\{X - S_T, 0\}\big)$.

– For continuously averaged geometric Asian options define

$$G(T) = \exp\left(\int_0^T \log(S_t)dt\right).$$

Then the price of an Asian call option is $\mathbb{E}\big(e^{-rT}\max\{G(T) - X, 0\}\big)$ and the price of an Asian put option is $\mathbb{E}\big(e^{-rT}\max\{X - G(T), 0\}\big)$.

– For a binary asset-or-nothing option the price of a call is $\mathbb{E}\big(e^{-rT}S_T\mathbb{I}_{\{S_T>X\}}\big)$ and the price of a put is $\mathbb{E}\big(e^{-rT}S_T\mathbb{I}_{\{S_T<X\}}\big)$.

– For a binary cash-or-nothing option the price of a call is $\mathbb{E}\big(e^{-rT}X\mathbb{I}_{\{S_T>X\}}\big)$ and the price of a put is $\mathbb{E}\big(e^{-rT}X\mathbb{I}_{\{S_T<X\}}\big)$.

– For a floating-strike lookback option the price of a call is $\mathbb{E}\big(e^{-rT}(S_T - \min_{0\le t\le T}S_t)\big)$ and the price of a put is $\mathbb{E}\big(e^{-rT}(\max_{0\le t\le T}S_t - S_T)\big)$.

– For an up-and-in barrier option with barrier level H and cash rebate K, set $A = \{\max_{0\le t\le T}S_t > H\}$. Then the price of a call is

$$\mathbb{E}\big(e^{-rT}\max\{S_T - X, 0\}\mathbb{I}_A + e^{-rT}K(1-\mathbb{I}_A)\big)$$

and the price of a put is

$$\mathbb{E}\big(e^{-rT}\max\{X - S_T, 0\}\mathbb{I}_A + e^{-rT}K(1 - \mathbb{I}_A)\big)$$

– For a down-and-in barrier option with barrier level H and cash rebate K, set $A = \{\min_{0\le t\le T}S_t < H\}$. Then the price of a call is

$$\mathbb{E}\big(e^{-rT}\max\{S_T - X, 0\}\mathbb{I}_A + e^{-rT}K(1 - \mathbb{I}_A)\big)$$

and the price of a put is

$$\mathbb{E}\big(e^{-rT}\max\big(\{X - S_T, 0\}\mathbb{I}_A + e^{-rT}K(1 - \mathbb{I}_A)\big)\big)$$

– For an up-and-out barrier option with barrier level H and cash rebate K, set $A = \{\max_{0\le t\le T}S_t > H\}$. Then the price of a call is

$$\mathbb{E}\big(e^{-rT}\max\{S_T - X, 0\}(1 - \mathbb{I}_A) + e^{-rT}K\mathbb{I}_A\big)$$

and the price of a put is

$$\mathbb{E}\big(e^{-rT}\max\{X - S_T, 0\}(1 - \mathbb{I}_A) + e^{-rT}K\mathbb{I}_A\big)$$

– For a down-and-out barrier option with barrier level H and cash rebate K, set $A = \{\min_{0\le t\le T}S_t < H\}$. Then the price of a call is

$$\mathbb{E}\big(e^{-rT}\max\{S_T - X, 0\}(1 - \mathbb{I}_A) + e^{-rT}K\mathbb{I}_A\big)$$

and the price of a put is

$$\mathbb{E}\big(e^{-rT}\max\{X - S_T, 0\}(1 - \mathbb{I}_A) + e^{-rT}K\mathbb{I}_A\big)$$

– The price of an American call option is $\text{esssup}_{0\le\tau\le T}\mathbb{E}(e^{-r\tau}\max\{S_\tau - X, 0\})$ and the price of an American put option is $\text{esssup}_{0\le\tau\le T}\mathbb{E}(e^{-r\tau}\max\{X - S_\tau, 0\})$. Here $\text{esssup}_{0\le\tau\le T}$ denotes the essential supremum over all stopping times τ for the process S which take values in $[0, T]$. If S is a Markov process, then the essential supremum may be replaced with the normal supremum. Note that if the asset S pays no dividends then the price of an American call option is the same as a European call option.

2.4.1 The Black–Scholes Model

The best known model of asset behaviour is the Black–Scholes model. Under the risk-neutral measure, the asset is governed by the SDE

$$\frac{dS_t}{S_t} = (r - q)dt + \sigma dW_t$$

where r is the continuously compounded risk-free interest rate, q is the continuously compounded dividend yield, σ is the volatility of log-asset returns (i.e., $\log(S_{t+dt}/S_t)$) and $W = (W_t)_{t\ge0}$ is a standard Brownian motion. Under this model, the price of any option P must solve the Black–Scholes PDE

$$\frac{\partial P}{\partial t} + \frac{\partial P}{\partial S}(r - q)S + \frac{1}{2}\frac{\partial^2 P}{\partial S^2}\sigma^2 S^2 - rP = 0$$

at all times before the option is exercised. This PDE admits a closed form solution for a number of different options.

2.4.2 The Black–Scholes Model with Term Structure

The simplest extension of the Black–Scholes model is to allow r, q and σ to be deterministic functions of time so that

$$\frac{dS_t}{S_t} = (r_t - q_t)dt + \sigma_t dW_t.$$

In this case one can still obtain closed form solutions for some options, e.g., European calls and puts.

2.4.3 The Heston Model

Heston (1993) proposed a stochastic volatility model with the following form

$$\begin{aligned} \frac{dS_t}{S_t} &= (r - q)dt + \sqrt{v_t}dW_t^{(1)} \\ dv_t &= \kappa(\eta - v_t)dt + \sigma\sqrt{v_t}dW_t^{(2)} \end{aligned}$$

where $W^{(1)}$ and $W^{(2)}$ are two Brownian motions with quadratic covariation given by $d\langle W^{(1)}, W^{(2)}\rangle_t = \rho dt$. In this model r and q are the continuously compounded risk free interest rate and dividend rate respectively, $v = (v_t)_{t\ge0}$ is the stochastic volatility process, η is the long term mean of volatility, κ is the rate of mean reversion, and σ is the volatility of volatility. The prices of European call and put options in the Heston model are available in closed form up to the evaluation of an integral transform (see Lewis (2000)).

2.4.4 The Heston Model with Term Structure

The Heston model can be extended by allowing the coefficients to become deterministic functions of time:

$$\begin{aligned} \frac{dS_t}{S_t} &= (r_t - q_t)dt + \sqrt{v_t}dW_t^{(1)} \\ dv_t &= \kappa_t(\eta_t - v_t)dt + \sigma_t\sqrt{v_t}dW_t^{(2)} \end{aligned}$$

where $W^{(1)}$ and $W^{(2)}$ are two Brownian motions with quadratic covariation given by $d\langle W^{(1)}, W^{(2)}\rangle_t = \rho_t dt$. When the coefficients are restricted to being piecewise constant functions of

time, the prices of European call and put options can be calculated as described in Elices (2008) and Mikhailov and NÎgel (2003).

2.5 Hypergeometric Functions

The confluent hypergeometric function $M(a,b,x)$ (or ${}_1F_1(a;b;x)$) requires a number of techniques to approximate it over the whole parameter (a,b) space and for all argument (x) values. For x well within the unit circle $|x| \leq \rho < 1$ (where $\rho = 0.8$ say), and for relatively small parameter values, the function can be well approximated by Taylor expansions, continued fractions or through the solution of the related ordinary differential equation by an explicit, adaptive integrator. For values of $|x| > \rho$, one of several transformations can be performed (depending on the value of x) to reformulate the problem in terms of a new argument x' such that $|x'| \leq \rho$. If one or more of the parameters is relatively large (e.g., $|a| > 30$) then recurrence relations can be used in combination to reformulate the problem in terms of parameter values of small size (e.g., $|a| < 1$).

Approximations to the hypergeometric functions can therefore require all of the above techniques in sequence: a transformation to get an argument well inside the unit circle, a combination of recurrence relations to reduce the parameter sizes, and the approximation of the resulting hypergeometric function by one of a set of approximation techniques. Similar complications arise in the computation of the Gaussian Hypergeometric Function ${}_2F_1$.

All the techniques described above are based on those described in Pearson (2009).

3 Recommendations on Choice and Use of Available Functions

3.1 Vectorized Function Variants

Many functions in Chapter s which compute functions of a single real argument have variants which operate on vectors of arguments. For example, nag_bessel_i0 (s18aec) computes the value of the I_0 Bessel function for a single argument, and nag_bessel_i0_vector (s18asc) computes the same function for multiple arguments. In general it should be more efficient to use vectorized functions where possible, though to some extent this will depend on the environment from which you call the functions. See Section 4 for a complete list of vectorized functions.

3.2 Elliptic Integrals

IMPORTANT ADVICE: users who encounter elliptic integrals in the course of their work are strongly recommended to look at transforming their analysis directly to one of the Carlson forms, rather than to the traditional canonical Legendre forms. In general, the extra symmetry of the Carlson forms is likely to simplify the analysis, and these symmetric forms are much more stable to calculate. Note, however, that this transformation may eventually lead to the following combination of Carlson forms:

$$R_F(0,1-m,1) - \frac{1}{3}mR_D(0,1-m,1)$$

with possibly $m \to 1$, which makes R_F and R_D undefined, although the combination itself remains defined and $\to 1$. The function nag_elliptic_integral_complete_E (s21bjc) returning the Legendre form $E(m)$ through this combination makes provision for such a case, and allows $m = 1$.

The function nag_elliptic_integral_rc (s21bac) for R_C is largely included as an auxiliary to the other functions for elliptic integrals. This integral essentially calculates elementary functions, e.g.,

$$\ln x = (x-1)\,R_C\Big(\big(\tfrac{1+x}{2}\big)^2, x\Big), \quad x > 0;$$

$$\arcsin x = x\,R_C\big(1-x^2, 1\big), |x| \leq 1;$$

$$\operatorname{arcsinh} x = x\,R_C\big(1+x^2, 1\big), \text{etc.}$$

In general this method of calculating these elementary functions is not recommended as there are usually much more efficient specific functions available in the Library. However, nag_elliptic_integral_rc (s21bac) may be used, for example, to compute $\ln x/(x-1)$ when x is close to 1, without the loss of significant figures that occurs when $\ln x$ and $x-1$ are computed separately.

3.3 Bessel and Airy Functions

For computing the Bessel functions $J_\nu(x)$, $Y_\nu(x)$, $I_\nu(x)$ and $K_\nu(x)$ where x is real and $\nu = 0$ or 1, special functions are provided, which are much faster than the more general functions that allow a complex argument and arbitrary real $\nu \geq 0$. Similarly, special functions are provided for computing the Airy functions and their derivatives $\mathrm{Ai}(x)$, $\mathrm{Bi}(x)$, $\mathrm{Ai}'(x)$, $\mathrm{Bi}'(x)$ for a real argument which are much faster than the functions for complex arguments.

3.4 Option Pricing Functions

For the Black–Scholes model, functions are provided to compute prices and derivatives (Greeks) of all the European options listed in Section 2.4. Prices for American call and put options can be obtained by calling nag_amer_bs_price (s30qcc) which uses the Bjerksund and Stensland (2002) approximation to the theoretical value. For the Black–Scholes model with term structure, prices for European call and put options can be obtained by calling nag_pde_bs_1d_analytic (d03ndc). The prices of European call and put options in the standard Heston model can be obtained by calling nag_heston_price (s30nac), while nag_heston_term (s30ncc) returns the same prices in the Heston model with term structure.

3.5 Hypergeometric Functions

Two functions are provided for the confluent hypergeometric function ${}_1F_1$. Both return values for ${}_1F_1(a;b;x)$ where parameters a and b, and argument x, are all real, but one variant works in a scaled form designed to avoid unnecessary loss of precision. The unscaled function nag_specfun_1f1_real (s22bac) is easier to use and should be chosen in the first instance, changing to the scaled function nag_specfun_1f1_real_scaled (s22bbc) only if problems are encountered. Similar considerations apply to the Gaussian hypergeometric function functions nag_specfun_2f1_real (s22bec) and nag_specfun_2 f1_real_scaled (s22bfc).

4 Functionality Index

Zeros of Bessel functions $J_\alpha(x)$, $J'_\alpha(x)$, $Y_\alpha(x)$, $Y'_\alpha(x)$,
scalar ..... nag_bessel_zeros (s17alc)

5 Auxiliary Functions Associated with Library Function Arguments

None.

6 Functions Withdrawn or Scheduled for Withdrawal

None.

7 References

Abramowitz M and Stegun I A (1972) *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Bjerksund P and Stensland G (2002) Closed form valuation of American options **Discussion Paper 2002/09** *NHH Bergen Norway* http://www.nhh.no/

Carlson B C (1965) On computing elliptic integrals and functions *J. Math. Phys.* **44** 36–51

Carlson B C (1977a) *Special Functions of Applied Mathematics* Academic Press

Carlson B C (1977b) Elliptic integrals of the first kind *SIAM J. Math. Anal.* **8** 231–242

Clenshaw C W (1962) Chebyshev Series for Mathematical Functions *Mathematical tables* HMSO

Elices A (2008) Models with time-dependent parameters using transform methods: application to Heston's model *arXiv:0708.2020v2*

Fox L and Parker I B (1968) *Chebyshev Polynomials in Numerical Analysis* Oxford University Press

Haug E G (2007) *The Complete Guide to Option Pricing Formulas* (2nd Edition) McGraw-Hill

Heston S (1993) A closed-form solution for options with stochastic volatility with applications to bond and currency options *Review of Financial Studies* **6** 327–343

Joshi M S (2003) *The Concepts and Practice of Mathematical Finance* Cambridge University Press

Lewis A L (2000) Option valuation under stochastic volatility *Finance Press, USA*

Mikhailov S and NÎgel U (2003) Heston's Stochastic Volatility Model Implementation, Calibration and Some Extensions *Wilmott Magazine July/August* 74–79

Pearson J (2009) Computation of hypergeometric functions *MSc Dissertation, Mathematical Institute, University of Oxford*

Schonfelder J L (1976) The production of special function routines for a multi-machine library *Softw. Pract. Exper.* **6(1)**