Pergamon

Computers Math. Applic. Vol. 27, No. 7, pp. 41–57, 1994
Elsevier Science Ltd
Printed in Great Britain
0898-1221/94 $6.00 + 0.00

0898-1221(94)E0025-F

Ultra-High Precision Computations

H. G. KHAJAH* AND E. L. ORTIZ
Department of Mathematics, Imperial College
London SW7 2BZ, U.K.

(Received June 1993; accepted July 1993)

Abstract—We describe a machine independent Fortran subroutine which performs the four basic arithmetic operations with a degree of accuracy prescribed by the user. Tables of Chebyshev expansions of orders 48 and 50 for some basic mathematical functions are obtained as a result of applying this subroutine in conjunction with the recursive formulation of the Tau Method. A recently devised technique for the sharp determination of upper and lower error bounds for Tau Method approximations (see [1]) enables us to find the degree n required to achieve a prescribed accuracy ϵ over a given interval $[a, b]$. A number of practical illustrations are given.

1. INTRODUCTION

The advent of new supercomputers has put new demands on the question of designing subroutines for the mathematical functions that are used in scientific computations. Their requirements have also extended the set of functions that today are regarded as basic. These functions need to be computed at, what is now called, an ultra-high precision. In this paper, we attempt to describe one such subroutine, written in Fortran-77, with the facility of constructing the relevant approximants with a degree of accuracy prescribed by the user. In our examples, we have chosen a background accuracy of 100-D for such subroutine and several levels of accuracy for the coefficients of the approximations we generate. However, our work allows the user to fix such background accuracy according to his needs.

Our algorithm is suitable only for computations involving the four basic arithmetic operations. Nevertheless, this is sufficient enough in view of our plan to obtain polynomial approximations of functions using algebraic techniques which require only the four basic operations. For the construction of high precision approximations of any required degree, we use the recursive formulation of the Tau Method given in [2,3]. This approach enables us to construct the Tau Method polynomial (or rational) approximation in a purely recursive form.

For other works of the same nature written in Fortran, we cite [4–6]. We also refer to [7] for an arbitrary precision package in Algol 68. The work of Hull and Abrham [8] is based on an extension of the Turing language, called Numerical Turing; this is a Pascal-like language described in [9]. These works, in general, do not utilize the Tau Method in the form that will be presented here, in their computations of mathematical functions. The only exception is the subsequent work of Schonfelder [10,11], where he obtains Chebyshev expansions for several of these functions using Clenshaw's method—see [12].

In this paper, however, we incorporate recursively generated canonical polynomials for the Tau Method, as in [2,3], into our algorithm, in order to be able to generate tables of coefficients of Chebyshev approximations in a straightforward manner, which takes full advantage of recursivity.

*Permanent address: Applied Sciences Department, College of Technological Studies, P.O. Box 42325, Shuwaikh 70654, Kuwait.

Thus, to obtain an approximation of a certain function, the user has to specify the degree of the approximation together with the relevant recursive expression for the canonical polynomials. Such recursive expression can be found through a simple and well-defined computational procedure for which software is available—see [13]. The computational process involves repeated calls to the high precision subroutines, and once the tables of approximation coefficients are determined, they can be stored for future referencing.

There is, finally, a further and important advantage in using this formulation of the Tau Method. Detailed error estimations recently derived for this method and, in particular, a technique for the determination of sharp upper and lower error bounds discussed by Khajah and Ortiz in [1], enable us to find the degree n required to achieve a prescribed accuracy ϵ over a given interval $[a, b]$. A number of practical examples are discussed in Section 4.

2. ULTRA-HIGH PRECISION COMPUTATIONS

Machine precision limits the degree of accuracy of a computation, whether it is carried out in floating point or in fixed point arithmetic. In some cases, this limitation may not have a significant effect on the final result of a computaion, but there are examples where intermediary truncation or rounding off of numbers can be damaging in the end. One striking example is the behaviour of chaotic trajectories, see [14], where periodicity will inevitably occur as a result of limited machine precision. We shall discuss an alternative way of performing the four basic arithmetic operations by which means it is possible to specify any number of digits that we require in a computation, subject of course to the computer storage capacity, and hence, obtain the desired degree of accuracy. In other words, the user controls the upper level of computational error. In some ways, our algorithm resembles pencil-and-paper calculations and involves no numbers of type `real`; in fact, all variables are declared as 2-byte integers (`integer*2`).

For a particular machine, let N be its precision and let its floating point variable range be $[10^{-|q|-1}, 10^{|q|})$. Then the magnitude of the relative error involved in a certain arithmetic operation, called the *round-off unit*, is at most $0.5 \times 10^{-N+1}$ in the case of rounding, and 10^{-N+1} when truncation is used. For large values of the input-output data and when the intermediary calculations involve large numbers, e.g., factorials or high order Chebyshev polynomials, errors are bound to occur due to the limitations on N and q, which lead to rounding, chopping or exponent spills. Our algorithm, however, allows for much greater flexibility with N and q. In the following, we shall replace q by x_{-1} for notational convenience.

Let N be the required precision and let x be a real number, expressed by its normalized floating point representation

$$x = \pm \underbrace{0.x_1 x_2 \dots x_N}_{\text{mantissa}} \times 10^{x_{-1}},$$

where $0.1 \leq 0.x_1 x_2 \dots x_N < 1$. We write x as an $(N+2)$-vector

$$x := (x_{-1}, x_0, x_1, \dots, x_N),$$

where x_{-1} and x_0 represent the exponent and the sign of x, respectively. By convention, let $x_0 = 0$ when $x \geq 0$ and $x_0 = 1$ otherwise, while x_{-1} can assume any integral value. For example, when $N = 8$, we have

$$\begin{array}{rcrlll} & & & 0.50092 & & := & (0,0,5,0,0,9,2,0,0,0) \\ -123.4567 & = & - & 0.1234567 & \times 10^3 & := & (3,1,1,2,3,4,5,6,7,0) \\ 0.0000123 & = & & 0.123 & \times 10^{-4} & := & (-4,0,1,2,3,0,0,0,0,0). \end{array}$$

As we truncate after N digits, the round-off unit of our algorithm (*not the machine's*) becomes 10^{-N+1}.

3. THE PROCEDURE

The procedure is best illustrated by way of an example. If we want to perform the four arithmetic operations on two numbers, say $x = 5620.328$ and $y = 42.17$, with an 8-D accuracy, then we put $N = 8$ and

$$x = 0.56203280 \times 10^4 := (4,0,5,6,2,0,3,2,8,0),$$
$$y = 0.42170000 \times 10^2 := (2,0,4,2,1,7,0,0,0,0).$$

3.1. Addition and Subtraction

Addition and subtraction are performed in the same manner; if $x-y$ is sought, we simply change the sign of y (i.e., put $y_0 = 1 - y_0$) and take the sum $x + (-y)$. There are three subroutines involved; the main one is **sum** which calls the other two, **add** and **sub**. Thus, to compute $x + y$, subroutine **sum** is called first. Define ν to be $|x_{-1} - y_{-1}|$ and equate the exponents by shifting the operand with the lower exponent ν number of times to the right, then raise its exponent by the same amount; this is merely an alignment of the two operands. In the example at hand, y has a lower exponent and, with $\nu = 2$, it becomes

$$y = 0.00421700 \times 10^4 := (4,0,0,0,4,2,1,7,0,0).$$

To account for all the possibilities $x > 0$, $y > 0$; $x < 0$, $y > 0$; ..., etc., and whether x is greater or less than y in magnitude, we first compare the sign terms $\{x_0, y_0\}$, then we seek the smallest index i for which $x_i \neq y_i$ to determine which operand is larger, and hence, the result's sign is determined. Finally, one of the subroutines **add** or **sub** is called, and addition or subtraction is carried out, term by term, starting from the right-most position. It should be pointed out at this stage that it is the case where $\nu \geq N$ which causes one of the operands to become zero, since its mantissa is moved beyond the prescribed vector length. This is quite acceptable, since our accuracy requirement was set for N digits only.

3.2. Division

This is a series of subtractions, whereby the denominator is subtracted from the numerator a number of times equal to the integral part of the quotient, after which an 'equivalent of a decimal point' is added, and the process is repeated to obtain the remainder. The main subroutine is **quo**, but **sum** is also called during the iteration process. More precisely, to find $x \div y$, we first compare their signs and hence determine the sign of the quotient q, then we put $x_0 = y_0 = 0$. Starting with $x_{\text{old}} = x$, we define $x_{\text{new}} = x_{\text{old}} - y$ and continue this iterative process for as long as x_{new} is positive, taking $x_{\text{old}} = x_{\text{new}}$ at the end of each step. If the number of iterations is k, then we have

$$k \quad \begin{cases} = 0, & \text{if } x < y, \\ \geq 1, & \text{otherwise,} \end{cases}$$

and k is taken to be the integral part of the quotient. In the final iteration k, if $x_{\text{new}} = 0$, we stop the process. Otherwise, for $x_{\text{new}} < 0$, we raise its exponent by one unit (i.e., add a decimal point), put $x_{\text{old}} = x_{\text{new}} \times 10$ and repeat the same procedure $N - n(k)$ times, where $n(k)$ is the number of digits in the integral part k. The result is written as

$$x \div y = q := (3,0,1,3,3,2,7,7,8,7),$$

and we have a truncation error of 5×10^{-6}, which decreases by increasing N. Note that the exponent term q_{-1} is determined once we have the integral part of the quotient.

3.3. Multiplication

There is only one subroutine for multiplication, namely **prd**. Since the product $p = x \times y$ is of length at most $2N$, we define t_k, for $k = 2, \ldots, 2N$ as the sum of the terms $x_i \times y_j$ with $i, j = 1, \ldots, N$ subject to $k = i + j$. The idea is then to take the sums $t_{k-1} + t_k + t_{k+1}$ and break them up into their constituent digits, from which we obtain the digits in the actual product $p = (p_{-1}, p_0, p_1, \ldots, p_{2N})$. Thus, each t_k is reduced to a 2-D number by carrying the 'hundreds' digit to the previous one (t_{k-1}), then adding in the following manner:

$$z_k = \mathrm{mod}(t_{2k-1}, 10) \times 10 + t_{2k} + (t_{2k+1} - \mathrm{mod}(t_{2k+1}, 10)) \times 10^{-1},$$

where $k = 1, 2, \ldots, N$, $t_1 = t_{2N+1} = 0$. Similarly, each z_k is reduced to a 2-D number to represent $p_{2k-1}p_{2k}$, that is $z_1 \to p_1p_2$, $z_2 \to p_3p_4, \ldots$, $z_N \to p_{2N-1}p_{2N}$. The sign and exponent of p are determined in the usual manner, and the product is truncated to the prescribed length N, causing a loss of accuracy. To overcome this shortcoming, we choose N to be sufficiently large compared to the magnitudes of the numbers in question. In our example, $N = 8$ and the actual product

$$x \times y = p := (6, 0, 2, 3, 7, 0, 0, 9, 2, 3, 1, 7, 6, 0, 0, 0, 0, 0)$$

is of length $18 = 2 + 2 \times 8$. Since we are concerned with only 8 significant figures, the result becomes

$$(6, 0, 2, 3, 7, 0, 0, 9, 2, 3).$$

Therefore, had we chosen a larger value for N at the start, say $N = 12$, the accuracy of the result would not have been affected by truncation. This is important when we are computing, for example, the coefficients of Chebyshev polynomial of order 80—some of which being 30 digits long. This requires N to be at least 60 to allow for accuracy in multiplication.

3.4. Storage Economy

In order to use the computer's memory resources more efficiently, we may modify our subroutine by reducing the length of the vector which represents a real number. Seeking N-digit accuracy, we let the real number x be given as before

$$x = \pm 0.x_1x_2 \ldots x_N \times 10^{x_{-1}}.$$

For a fixed integer n such that $0 < n < N$, we have $N = mn + r$, where $0 \leq r < n$. Clearly, $0 < m < N$, and we may represent x as an $(m+3)$-vector:

$$x := (\xi_{-1}, \xi_0, \xi_1, \xi_2, \ldots, \xi_m, \xi_{m+1}),$$

where $\xi_{-1} = x_{-1}$ and $\xi_0 = x_0$ are the exponent and sign terms as defined earlier; the remaining terms are strings, each of length n, written as

$$\begin{aligned} \xi_i &= x_{in-n+1}x_{in-n+2} \cdots x_{in}, \qquad 1 \leq i \leq m, \\ \xi_{m+1} &= x_{mn+1}x_{mn+2} \cdots x_N \underbrace{00\ldots0}_{n-r}. \end{aligned}$$

The choice of n is dependent on the type of integer declaration; since the multiplication subroutine **prd** involves products $\xi_i\xi_j$, a large value for n may result in these products being too large for the chosen integer type. It is also possible to adapt our algorithm to a larger, e.g., hexadecimal, base instead of the decimal one. This will reduce the length of vectors used to represent the real numbers.

4. APPLICATIONS TO SOME BASIC FUNCTIONS

We have applied our algorithm to evaluate some basic functions with the Tau Method using, for the sake of comparison with the existing tables, the regular (T_n) and shifted (T_n^*) Chebyshev polynomials as bases for the required expansions. Furthermore, since Legendre polynomial expansions yield comparable results, we have not attempted to compute the approximations in these terms. We have adopted the canonical polynomial approach discussed by Ortiz in [2,3]. In this respect, we make extensive use of the identities concerning the derivatives of Chebyshev polynomials found in [15, Volume I, Chapter VIII]. We have computed coefficients of the various Chebyshev expansions for the functions under consideration to 100 decimals, but in their tabulations they have been truncated to the levels of expansion errors involved. In the process of error estimation, we have used Lanczos' estimates [16] and our upper bound estimates, given in [1], for the validity of the first three expansions given below. However, instead of taking $\delta_{n,n+k}$, we use a conservative estimate of our upper bound given by

$$\sum_{j=0}^{k} \delta_{n,n+j} \geq \delta_{n,n+k}.$$

In [17], Namasivayam and Ortiz give the end point error estimate of a Chebyshev expansion of order n in the interval $[-r, r]$ to be

$$e_n := O\left[\frac{r^n}{(n+1)!\, n\, 2^n}\right],$$

and we check our end point errors against the above for the first two expansions. For the remaining expansions, we have resorted to Mathematica to compute their maximum errors. We cite [18] for references on recent literature on the evaluation of mathematical functions.

Finally, we should emphasize at this point that our tabulated results are presented here only for comparison with others in the literature. They are not intended for transfer into the computer memory since they may be directly generated as part of a Tau Method software.

4.1. The Function e^x

This satisfies the linear differential equation

$$Dy(x) := y'(x) - y(x) = 0, \qquad y(0) = 1. \tag{1}$$

We shall confine x to the unit interval $[0,1]$ and define the canonical polynomial $Q_k(x)$ such that $DQ_k(x) = T_k^*(x)$ for $k \geq 0$. The linear differential operator D is applied to $T_n^*(x)$ as follows:

$$DT_n^*(x) = \sum_{k=0}^{n-1} \beta_k T_k^*(x) - T_n^*(x) = \sum_{k=0}^{n-1} \beta_k DQ_k(x) - DQ_n(x),$$

so that

$$T_n^*(x) = \sum_{k=0}^{n-1} \beta_k Q_k(x) - Q_n(x),$$

and the coefficients β_k are determined from the identities in [15]. Hence,

$$\begin{aligned} Q_{2n}(x) &= 8n \sum_{k=0}^{n-1} Q_{2k+1}(x) - T_{2n}^*(x), \\ Q_{2n+1}(x) &= 2(2n+1)Q_0(x) + 4(2n+1) \sum_{k=1}^{n} Q_{2k}(x) - T_{2n+1}^*(x), \end{aligned}$$

Table 1. The exponential function e^x.

k	a_k^{48}	
0	1.753387654377090395721946355212090821042278927770743410957428044218541910291739519014075469679 37	
1	0.850391653780810966535234986588273561683176957565744134523363301477205811023644483198002747453 73	
2	0.105208693630936925302952764071087395351850025278510283821402882528260576488901105236139949543 8	
3	0.872210473331556411161287401957439886837675533766186395214024125112119911243564130888315110333	E-2
4	0.543436831150155963598275836194608931328961226567916395719987514806187139673409520542136303 78	E-3
5	0.271154349130686940404606404606559671133757125752016206204410142222048776610888362089702427 1	E-4
6	0.112813288878208278906302698148958906144697506388398331116723036208958645163280536273144938	E-5
7	0.402455822987071029479929049058296386483110419860211524274855320548028219015075034154573 9	E-7
8	0.125658441828390651922564412635917929426588827539104319763546455510743839059526709864226	E-8
9	0.348809136220943327722928623359012318026171735077701031506662913647934024589562589049 6	E-10
10	0.871527888510539423101082266734949371670029111319484211478065974875902072841778063 47	E-12
11	0.197980816727558482495716665032569358160090549907346915436523697573195452851363657 7	E-13
12	0.412294909282100119928940591644195765630691727157783557361705553842080295777969 5	E-15
13	0.792602721504249298251810433553906573585208716108079029050317289969108779382932	E-17
14	0.141494099890484837999166196164347366383194781582462255540563058143730498844 75	E-18
15	0.235762117534205456479735033561321839317939246290398023164164364217985852294	E-20
16	0.368293699615641113251760275542627924312338082234416420644396129389874680 9	E-22
17	0.541497801951439986084572140399677580428736603715139517508414084660564 75	E-24
18	0.751942886619227142512200708471696207971917081215487386745518206906467	E-26
19	0.989235855964434757876303000563106889563052399885990516409756879085	E-28
20	0.123636086256726526210428043734971903997257302134594274102978801 81	E-29
21	0.147165910622548192878650683331657584993982809236323585926464602 3	E-31
22	0.167213337860441923614697363795326023117423760824619247485357 98	E-33
23	0.181733053593000977170031917706846506498997106586481393495721	E-35
24	0.189285548810246182679995050272371383464227650563654692943 4	E-37
25	0.189267351646417972366694453699783733385620453728882700 12	E-39
26	0.181971638282103133005965725876500786071968347719928153	E-41
27	0.168478330307140404900987882229158707733721001574201 1	E-43
28	0.150415503914957128988130690093817195496660197909 24	E-45
29	0.129659223884204342815093240834487774615799158426	E-47
30	0.108042092800913226225307258113769423331741347 2	E-49
31	0.871252310847134472453109796446661770954171 9	E-52
32	0.680625586855164112164335438336373342402 55	E-54
33	0.515596725244088827604353761038926788951	E-56
34	0.379095329668597265884737649899809859 4	E-58
35	0.270768947965460011105571751853801 62	E-60
36	0.188025169532503369371973044875847	E-62
37	0.127038386551592099305672325808 4	E-64
38	0.835743614706239957800267930 5	E-67
39	0.535711624362572003160036 7	E-69
40	0.334807006276328706106698	E-71
41	0.204143204460733893198 1	E-73
42	0.121509607251212616 76	E-75
43	0.706427869669702 47	E-78
44	0.401366802379107	E-80
45	0.2229748247402	E-82
46	0.12117846688	E-84
47	0.64456631	E-87
48	0.335711	E-89

Accuracy of coeff's 95-D; estimated functional error 3.5×10^{-92}.

and each Q_r becomes a linear combination of T_k^*'s for $k \leq r$. As such, we let the approximate polynomial solution of degree n be

$$y_n(x) = \tau Q_n(x) = \sum_{k=0}^{n} a_k^n T_k^*(x), \tag{2}$$

where the parameter τ is determined from the boundary condition $y_n(0) = 1$,

$$\tau = \frac{1}{Q_n(0)},$$

and the perturbed form of equation (1) becomes

$$Dy_n(x) = \tau T_n^*(x).$$

Denote the approximation error by $e_n(x) = e^x - y_n(x)$ and let

$$\|e_n\| = \max\{|e_n(x)| : x \in [0,1]\}.$$

For this, Lanczos' upper bound estimate of the error (see [16]) is

$$\|e_n\| \leq \frac{|\tau|}{\sqrt{4n^2+1}},$$

and furthermore, our conservative estimate of the upper bound is

$$\|e_n\| \le \delta_{n,n+1} + \delta_{n+1,n+2},$$

where $\delta_{i,j} = \|y_i - y_j\|$. The coefficients a_k^n for $n = 48$ are presented in Table 1 with 95-D accuracy. Compared with the value of e in NBS [19], the error at $x = 1$ is 2.7×10^{-93}, while the asymptotic estimate of Namasivayam and Ortiz [17] is of order 10^{-79}. With $|\tau| = 3.36 \times 10^{-90}$ in this case, the upper bound estimate of Lanczos is equal to 3.5×10^{-92}, whereas our estimate $\delta_{48,49} + \delta_{49,50}$ becomes 3.63×10^{-92}.

4.2. The Function e^{-x^2}

In this case, we solve the linear differential equation

$$Dy := y'(x) + 2xy(x) = 0, \qquad y(0) = 1, \tag{3}$$

for $x \in [0, 1]$. Following a similar argument as in the previous example, the canonical polynomials become

$$Q_1(x) = T_0^*(x) - Q_0(x)$$

and for $n \ge 1$

$$Q_{2n} = 2T_{2n-1}^* - 8(2n-1)\left\{\frac{Q_0}{2} + \sum_{k=1}^{n-1} Q_{2k}\right\} - 2Q_{2n-1} - Q_{2n-2},$$

$$Q_{2n+1} = 2T_{2n}^* - 16n\sum_{k=0}^{n-1} Q_{2k+1} - 2Q_{2n} - Q_{2n-1},$$

with $Q_0(x)$ being undefined. Then, the canonical polynomial of order r becomes

$$Q_r = \alpha_r Q_0 + \sum_{k=0}^{r-1} \beta_k^r T_k^*.$$

Let the n^{th} order approximation be

$$y_n(x) = \tau_1 Q_{n+1}(x) + \tau_2 Q_n(x), \tag{4}$$

so that the perturbed equation becomes

$$Dy_n(x) = \tau_1 T_{n+1}^*(x) + \tau_2 T_n^*(x).$$

The parameters τ_1 and τ_2 are determined from the initial condition together with the condition that the undefined $Q_0(x)$ be eliminated from the right side of (4), which result in the following two equations:

$$\alpha_{n+1}\tau_1 + \alpha_n\tau_2 = 0$$
$$\zeta_{n+1}\tau_1 + \zeta_n\tau_2 = 1,$$

where $\zeta_r = \sum_{k=0}^{r-1}(-1)^k \beta_k^r$. Following Lanczos [16], the upper bound for $e_n(x)$ is analytically derived as

$$\|e_n\| \le \frac{|\tau_1|}{2}\frac{n}{n^2-1} + \frac{|\tau_2|}{2}\frac{n-1}{n(n-2)}.$$

Table 2. The exponential function e^{-x^2}.

k	a_k^{48}	
0	+0.731032658938930376031844613891959150533298844649137117562546	
1	−0.333462228141044950346527593583512684838408494681310124605788	
2	−0.47443083061151541909857111644009831093903944890603300602811	E-1
3	+0.17856748608620034828344336604242100775326816245963950003710	E-1
4	+0.333035400106111156231581104251432336663949194073276261116	E-3
5	−0.462498456679420598440900578224793091700692947528478244598	E-3
6	+0.1254387284292594312395805690979141339744770672639624955	E-4
7	+0.7793395969581316023559569813658953832101319518580242885	E-5
8	−0.442408731447404547352457275483911324149417153820735445	E-6
9	−0.95719234120155378943640169341027659824806727550673023	E-7
10	+0.7944437694996698407211304821051635737261374131012876	E-8
11	+0.904774645327148098140460495036404941878079028288576	E-9
12	−0.101733378633728859114149471908594701876836497628507	E-9
13	−0.6723184105491774628737191903564061259351803995548	E-11
14	+0.1029005024007677791350891706482363359932023244407	E-11
15	+0.38731040530462940791809835588440431954371005724	E-13
16	−0.8646236913421621907604260875008947062244257438	E-14
17	−0.156719994022907203019698236464810838748304923	E-15
18	+0.62220138681883167205382131028621550996913635	E-16
19	+0.207239140551031511589417151020658119193932	E-18
20	−0.3914095573622990918093141119106712423691637	E-18
21	+0.3451819340519336276060785785722210383433	E-20
22	+0.2184146336336866518584702017435023372637	E-20
23	−0.42617638991910688168479226305431517830	E-22
24	−0.10928207988090609948034014092938304914	E-22
25	+0.322854657683910126080208578735457657	E-24
26	+0.49415324280211973052441822535378106	E-25
27	−0.1954079870306794610422389287352351	E-26
28	−0.203062698085711744853069534038697	E-27
29	+0.10179654237859823562276806183295	E-28
30	+0.760859923233337384914723638506	E-30
31	−0.4720314120570845013876818227	E-31
32	−0.2601751579708448704712065967	E-32
33	+0.198568470219595101168389054	E-33
34	+0.8099476118426028561608164	E-35
35	−0.767179389092896860820075	E-36
36	−0.22776279192939896226836	E-37
37	+0.2746068821917054437851	E-38
38	+0.56795029710617763443	E-40
39	−0.9166250749775196573	E-41
40	−0.120015578436871297	E-42
41	+0.28678565302376402	E-43
42	+0.185980663802135	E-45
43	−0.84448797083551	E-46
44	−0.47200433996	E-49
45	+0.23483410685	E-48
46	−0.115134558	E-50
47	−0.616794206	E-51
48	+0.6194633	E-53

Accuracy of coeff's 60-D; estimated functional error 3.25×10^{-54}.

For $n = 48$, we have $\tau_1 = 3.1 \times 10^{-54}$ and $\tau_2 = -3.0 \times 10^{-52}$ and this upper bound becomes 3.25×10^{-54}, while our conservative estimate $\delta_{48,49} + \delta_{49,50} = 3.238 \times 10^{-54}$. The value at $x = 1$ was compared with the tabulated value for e^{-1} in NBS [19] and the error was found to be of the order 10^{-56}. The asymptotic estimate of Namasivayam and Ortiz [17] is of order 10^{-34}. The coefficients of the expansion $y_{48}(x)$ are given in Table 2 with 60-D accuracy.

4.3. The Exponential Integral

Consider the linear differential equation

$$Dy(x) := x^2 y'(x) + (1 + x)\, y(x) = 1 \tag{5}$$

over the interval $[0, 1]$. Its solution $y(x)$ is related to the exponential integral by the relation

$$y(x) = -ze^z \mathrm{Ei}\,(-z), \quad \text{where} \quad z = x^{-1} \quad \text{and} \quad \mathrm{Ei}\,(\zeta) = \int_\zeta^\infty t^{-1} e^{-t}\, dt.$$

Note that the solution $y(x)$ has a singularity at the origin. Again, we apply the operator D to the basis elements $T_k^*(x)$ to obtain the canonical polynomials:

$$Q_1 = 2T_0^* - 3Q_0,$$
$$Q_2 = 2T_1^* - 10T_0^* + 13Q_0,$$

and for $n \geq 3$

$$Q_n = \frac{4}{n}\left\{T^*_{n-1} - \frac{2n+1}{2}Q_{n-1} - \frac{7n-6}{4}Q_{n-2} - (n-1)\left(2\sum_{k=1}^{n-3}Q_k + Q_0\right)\right\},$$

with $Q_0(x)$ being undefined. Since $DQ_0(x) = 1$, we express the approximate solution $y_n(x)$ as

$$y_n(x) = Q_0(x) - \frac{Q_{n+1}(x)}{\beta_{n+1}} = \tau \sum_{r=0}^{n} \alpha_r^{n+1} T_r^*(x), \tag{6}$$

where β_{n+1} is the coefficient of Q_0 in the expression for Q_{n+1}, and take the parameter τ to be

$$\tau = -\frac{1}{\beta_{n+1}}.$$

Then, we seek an exact solution to the perturbed equation

$$Dy_n = 1 + \tau T^*_{n+1}(x).$$

Taking $n = 48$, the estimated maximum error of Lanczos [16] is 1.09×10^{-16}, being of order $|\tau|$. The end-point ($x = 1$) error of Lanczos is $\tau/2(n^2-1)$ and equals 2.36×10^{-20}, which is close to the calculated value of 2.27×10^{-20} obtained by Mathematica. Due to the slow convergence of the coefficients $a_k^{48} = \tau\alpha_k^{49}$ in (6), see Table 3, it requires three successive approximations to obtain our conservative estimate, i.e.,

$$\delta_{49,49} + \delta_{49,50} + \delta_{50,51} = 9.77 \times 10^{-17}.$$

In Table 3, we have 25-D accuracy, but the canonical polynomials and the Tau parameters were computed to 100 decimals. Finally, we note the presence of only one Tau term in (6), where, in theory, there should have been two such terms. This has been followed by Lanczos in [16,Chapter VIII], perhaps because the initial condition $\lim_{x\to 0} y(x) = 1$ is inherent in the differential equation itself and is allowed to be perturbed as well. For $n = 12$, we have checked the case of two Tau terms, i.e., $Dy_{12} = 1 + \tau_0 T^*_{13} + \tau_1 T^*_{12}$, against that with one Tau and found that the results were comparable.

4.4. The Sine and Cosine Functions

The trigonometric functions have been discussed by Schonfelder in [11], where expansions of order 16 to 40-D accuracy were given for $\sin x/x$ and $\cos x$ in the intervals $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ and $\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$. However, it suffices to confine such expansions to the interval $\left[0, \frac{\pi}{2}\right]$ because of the symmetric nature of the trigonometric functions. This is achieved by considering the functions $\sin \pi x$ and $\cos \pi x$ in the interval $[0, 0.5]$, and has the advantage of dealing with a shorter interval (i.e., $\subset [0, 1]$) in which Chebyshev expansions yield better results. It should be noted here that the 100-D values for π and $\sqrt{\pi}$ are obtained from Mathematica.

The functions $\sin \pi x$ and $\cos \pi x$ satisfy the second order equation

$$Dy(x) = y''(x) + \pi^2 y(x) = 0, \tag{7}$$

with different supplementary conditions. Because of their symmetric nature, it suffices to consider expansions in the interval $[0, 0.5]$ and treat (7) as a boundary value problem where $y(0) = 0$

Table 3. The exponential integral.

k	a_k^{48}	
0	+0.7578721561413121060433513	
1	−0.1918875669402128932036812	
2	+0.3750330470645315315912 42	E-1
3	−0.907354073368710980818 84	E-2
4	+0.251098168139651362991 64	E-2
5	−0.764303904378879393037 3	E-3
6	+0.250102678959300290909 5	E-3
7	−0.866941909793140117765	E-4
8	+0.315098379826276936756	E-4
9	−0.119195886153579887516	E-4
10	+0.466657509735385890 6	E-5
11	−0.188260835860400756 8	E-5
12	+0.779890162464276155 3	E-6
13	−0.330816956694393433 1	E-6
14	+0.143352320755549628 7	E-6
15	−0.633329918169725161	E-7
16	+0.284798660593050 77	E-7
17	−0.130168413146710587	E-7
18	+0.603939225901528 67	E-8
19	−0.284137944879755 96	E-8
20	+0.135424903787632 83	E-8
21	−0.653331208951990 5	E-9
22	+0.318791030152633 3	E-9
23	−0.157225473908098 7	E-9
24	+0.783284254880498	E-10
25	−0.393963805004414	E-10
26	+0.199947698498175	E-10
27	−0.102353229308694	E-10
28	+0.52823903045445	E-11
29	−0.27474945696185	E-11
30	+0.14396848161587	E-11
31	−0.7597675933457	E-12
32	+0.403687992513	E-12
33	−0.2158941397414	E-12
34	+0.1161856023922	E-12
35	−0.629034309370	E-13
36	+0.342538396843	E-13
37	−0.187573380038	E-13
38	+0.103274988596	E-13
39	−0.57167956731	E-14
40	+0.31817332113	E-14
41	−0.17805675675	E-14
42	+0.10015158165	E-14
43	−0.5648054005	E-15
44	+0.3167705939	E-15
45	−0.1729800424	E-15
46	+0.875838171	E-16
47	−0.366547491	E-16
48	+0.88859997	E-17

Accuracy of coeff's 25-D; estimated functional error 9.77×10^{-17}.

and $y(0.5) = 1$ for the sine function and $y(0) = 1$ and $y(0.5) = 0$ for the cosine. The fact that one equation is involved simplifies the computation of the expansions' coefficients, since the canonical polynomials are independent of the initial or boundary conditions. We apply the transformation $t = 2x \in [0, 1]$ for $x \in [0, 0.5]$ and (7) becomes

$$\tilde{D}y(t) = 4y''(t) + \pi^2 y(t) = 0. \tag{8}$$

Then, for $n \geq 0$, we find the the canonical polynomials to be

$$Q_{2n}(t) = \frac{1}{\pi^2} T_{2n}^* - \frac{64}{\pi^2} \left(n^3 Q_0 + 2n \sum_{k=1}^{n-1} \left(n^2 - k^2\right) Q_{2k} \right),$$

$$Q_{2n+1}(t) = \frac{1}{\pi^2} T_{2n+1}^* - \frac{64}{\pi^2} (2n+1) \sum_{k=0}^{n-1} (n-k)(n+1+k)\, Q_{2k+1},$$

and the expansion of order n becomes

$$y_n(x) = \tau_0 \sum_{k=0}^{n} Q_k(t) + \tau_1 \sum_{k=0}^{n-1} Q_k(t) = \sum_{k=0}^{n} a_k^n T_k^*(t), \qquad t = 2x,$$

so that

$$Dy(x) = \tilde{D}y(t) = \tau_0 T_n^*(t) + \tau_1 T_{n-1}^*(t).$$

Table 4. The sine function.

k	a_k^{50}	
0	+0.602194701255546403285976667564525859028053998022000614384408292297196756106387495392352299	
1	+0.513625166679107025112286792544191962756601843152420539525470262938713271659739860670276657	
2	−0.103546344262963753811582693025147937446135716045422099736916346651690434037518812050919897	
3	−0.1373203423435855321199970910219843449132426087979352051442506529190780164964542926233383	E-1
4	+0.1358669838090361777593700796783370019122482238724332009058046029948409094361328290617053	E-2
5	+0.107263094406002214091092231279202495880676760925399099991825411232732104081150211443131	E-3
6	−0.7046296793468567813549767176246522144012811779952407390534336463893859714046663791181	E-5
7	−0.396390250614857181242252259704951515067682284195193747432120562054056236743720091324	E-6
8	+0.19499597755881365115981355355162674040022807458858418677097356007416001881001071038	E-7
9	+0.852292892617847612927436294585157776801709168503669028002887984058133912585147178	E-9
10	−0.33516506518372362871802735509792889398121213183487254610407614481702634515607203	E-10
11	−0.1197937393463184823426153322127600672073617021574725777306353881331963895388788	E-11
12	+0.39241234062080870690195014071539443433518695820977349008591896867624885237157	E-13
13	+0.1186390346010533285905223339595900218556295759432112525612343452064262647764	E-14
14	−0.33302642767861187034308856229220974636930681997217466970818133824489346378	E-16
15	−0.872422043888120639125512443498169891589076258526889735833840355175295907	E-18
16	+0.21424611449738250271750762163212208192555470289433676599192464765297551	E-19
17	+0.495157020438836026307105582810516039450319144465528111545089598921829	E-21
18	−0.10807525644499454355046327530367071596167725195684550723626795092025	E-22
19	−0.223464688542888062987008682320198958288743289930026833328739288874	E-24
20	+0.4389331080351098043186810004654149832547591303914207739148561304	E-26
21	+0.82107714637211406741639819754262848550118944246205553676362551	E-28
22	−0.1466066213865566718349143519228522168550831200905046962946951	E-29
23	−0.25038425315697410220064102176678742381656552706847510724146	E-31
24	+0.409795627254643213051835634236163187838373278153147419584	E-33
25	+0.643858255038260750230891256941052605481276748456445353	E-35
26	−0.97268504802479173937575802831783839515112724295743702	E-37
27	−0.1415001749878073328844137724561039761785325628717131	E-38
28	+0.19849123488523983328850636988122962182708983033119	E-40
29	+0.268831541463510818253046305997777767745004936012	E-42
30	−0.3519580356263122078694061126118681980042526868	E-44
31	−0.44591964525936046661543208973546641547218088	E-46
32	+0.547305671584154238027014209240377034483582	E-48
33	+0.6513817665337852965047063125786677844928	E-50
34	−0.75244170219223641825952062837265006318	E-52
35	−0.844340967305966221515862082222735616	E-54
36	+0.9211397749030148049372410810113874	E-56
37	+0.9777579120843858945855649310794	E-58
38	−0.1010538411949671194849718671937	E-59
39	−0.10176326744126913022744921528	E-61
40	+0.99915250264872476797518274	E-64
41	+0.957076020773005095303852	E-66
42	−0.8949409722017466088174	E-68
43	−0.81737476231324015338	E-70
44	+0.729562095740565454	E-72
45	+0.6367104709873043	E-74
46	−0.54359446622295	E-76
47	−0.454220512181	E-78
48	+0.3716326558	E-80
49	+0.2978168	E-82
50	−0.23392	E-84

Accuracy of coeff's 90-D; calculated functional error 7.5×10^{-87}.

Having the boundary conditions satisfied by y_n, we find that the Tau parameters for the sine and cosine are related in the following way:

$$\tau_0(\cos) = \tau_0(\sin) \quad \text{and} \quad \tau_1(\cos) = -\tau_1(\sin).$$

For $n = 50$ and $y(x) = \sin \pi x$, we have $\tau_0 = -2.3 \times 10^{-84}$ and $\tau_1 = 2.9 \times 10^{-82}$. In either case, the maximum absolute error is found to be 7.5×10^{-87}. The coefficients a_k^{50} for $\sin \pi x$ and $\cos \pi x$ are given to 90-D accuracy in Tables 4 and 5, respectively.

4.5. The Error Function

Schonfelder has also obtained expansions of order 27 for the error function erf (x) and of order 43 for its complement erfc (x), both to 30-D accuracy (see [10]). We shall follow Schonfelder's arguments with regard to the differential equations used in representing these two functions while applying Tau Method to obtain the required results in terms of regular Chebyshev polynomials. Comparing our results with those of Schonfelder, we find the two to be consistent up to the degrees of accuracy which he has chosen. Thus, we compute the coefficients of expansion of order 50 for the function $y(x) = \mathrm{erf}(x)/x$, where $x \in (0, 2)$. Apply the quadratic transformation

$$t = \frac{x^2}{2} - 1, \qquad t : (0,2) \to (-1,1),$$

Table 5. The cosine function.

k	a_k^{50}	
0	+0.602194701255546403285976667564525859028053998022000614384408292297196756106387495392352299	
1	−0.513625166679107025112286732544191962756601843152420539525470262938713271650739860670276657	
2	−0.103546344262963753811582693025147937446135716045422099736916346651690434037518812050919897	
3	+0.13732034234358553211999970910219843449132426087979352051442506529190780164964542926233383	E-1
4	+0.1358669838090361777593700796783370019122482238724332009058046029948409094361328290617053	E-2
5	−0.107263094406002214091092231279202495880676760925399099991825411232732104081150211448131	E-3
6	−0.7046296793468567813549767176246522144012811779952407390534836463893859714046663791181	E-5
7	+0.396390250614857181242252259704951515067682284195193747432120562054056236743720091324	E-6
8	+0.19499597755881365115981355355162674040022807458858418677097356007416001881001071038	E-7
9	−0.852292892617847612927436294585157776801709168503669028002887984058133912585147178	E-9
10	−0.33516506518372362871802735599792889398121213183487254610407614481702634515607203	E-10
11	+0.1197937393463184823426153322127600672073617021574725777306353881331963895388788	E-11
12	+0.39241234062080870690195014071539443433518635820977349008591896867624885237157	E-13
13	−0.1186390346010533285905223339595900218556295759432112525612343452064262647764	E-14
14	−0.33302642767861187034308856229220974636930681997217466970818133824489346378	E-16
15	+0.872422043888120639125512443498169891589076258526889735833840355175295907	E-18
16	+0.21424611449738250271750762163212208192555470289433676599192464765297551	E-19
17	−0.495157020438836026307105582810516039450319144465528111545089598921829	E-21
18	−0.10807525644499454355046327530367071596167725195684550723626795092025	E-22
19	+0.223464688542888062987008682320198958288743289930026833328739288874	E-24
20	+0.4389331080351098043186810004654149832547591303914207739148561304	E-26
21	−0.82107714637211406741639819754262848550118944246205553676362551	E-28
22	−0.1466066213865566718349143519228522168550831200905046962946951	E-29
23	+0.25038425315697410220064102176678742381656552706847510724146	E-31
24	+0.409795627254643213051835634236163187838373278153147419584	E-33
25	−0.643858255038260750230891256941052605481276748456445353	E-35
26	−0.97268504802479173937575802831783839515112724295743702	E-37
27	+0.1415001749878073328844137724561039761785325628717131	E-38
28	+0.19849123488523983328850636988122962182708983033119	E-40
29	−0.268831541463510818253046305997777767745004936012	E-42
30	−0.3519580356263122078694061126118681980042526868	E-44
31	+0.4459196452593664661543208973546641547218088	E-46
32	+0.547305671584154238027014209240377034483582	E-48
33	−0.6513817665337852965047063125786677844928	E-50
34	−0.75244170219223641825952062837265006318	E-52
35	+0.844340967305966221515862082222735616	E-54
36	+0.9211397749030148049372410810113874	E-56
37	−0.9777579120843858945855649310794	E-58
38	−0.1010538411949671194849718671937	E-59
39	+0.10176326744126913022744921528	E-61
40	+0.99915250264872476797518274	E-64
41	−0.957076620773005095303852	E-66
42	−0.8949409722017466088174	E-68
43	+0.81737476231324015338	E-70
44	+0.729562095740565454	E-72
45	−0.6367104709873043	E-74
46	−0.54359446622295	E-76
47	+0.454220512181	E-78
48	+0.3716326558	E-80
49	−0.2978168	E-82
50	−0.23392	E-84

Accuracy of coeff's 90-D; calculated functional error 7.5×10^{-87}.

and seek a Tau method solution to the differential equation

$$Dy(t) := (4+4t)y''(t) + (14+8t)y'(t) + 4y(t) = 0, \tag{9}$$

with initial conditions

$$y(-1) = \frac{2}{\sqrt{\pi}}, \qquad y'(-1) = -\frac{4}{3\sqrt{\pi}}. \tag{10}$$

The canonical polynomials $Q_j(t)$ become

$$Q_{2n} = \frac{2n}{4n+1}\left\{\frac{T_{2n}}{8n} - 2(n^2+1)Q_0 - 4\sum_{k=1}^{n-1}(n^2-k^2+1)Q_{2k} \right.$$
$$\left. -4\sum_{k=0}^{n-1}\left(n^2-k^2-k+\frac{5}{4}\right)Q_{2k+1}\right\}$$

$$Q_{2n+1} = \frac{2n+1}{4n+3}\left\{\frac{T_{2n+1}}{4(2n+1)} - \left[2n(n+1)+\frac{7}{2}\right]Q_0 - 4\sum_{k=0}^{n-1}\left[1+(n-k)(n+1+k)\right]Q_{2k+1} \right.$$
$$\left. -4\sum_{k=1}^{n}\left(n^2+n-k^2+\frac{7}{4}\right)Q_{2k}\right\}.$$

Table 6. The error function.

k	a_k^{50}	
0	+0.7415552820424017909447240395285827483926705618861171012079305285624247	
1	−0.3010710733865949424707310463108174388564346693890941434587183401901566	
2	+0.6899483068983156624660318071758496436932146091190716241888400598154060	E-1
3	−0.1391627126472218768254652568686419527456265441758770470772872078317220	E-1
4	+0.2420799522433463662891678239445241555458322346623289745723433216411600	E-2
5	−0.3658639685848086446493825765085529089145478492667196464410430167023	E-3
6	+0.4862098443231904828288756822152504591731506555392930767255853734830	E-4
7	−0.57492565580356848350542147952991131279697792278788485772375631238	E-5
8	+0.6113243578434764697067584304007906663818982320716324562500040268	E-6
9	−0.5899101531295843439084612502741835180818792539870840188850577960	E-7
10	+0.520700909206864824045508362663881280562579478768586984441924460	E-8
11	−0.42329758799655432680967975613367760223516881975395800722832910	E-9
12	+0.3188113506649174974754397084909665227022896560429407365414730	E-10
13	−0.223615501883268427274975927905757353682267442240385485891570	E-11
14	+0.146732984799108491850625366445419367386490774779140762014000	E-12
15	−0.90440019853817471414651879387249230681392873842754422803	E-14
16	+0.5254813715470918677368415666992690643106562349014381451	E-15
17	−0.28874261222849453543211098613287041863410105114926099400	E-16
18	+0.1504785187557632499601007415237720809038261412402599500	E-17
19	−0.745728928209442366024889414167882105664958963880551	E-19
20	+0.352256380990399687811188238353622487974390431178210	E-20
21	−0.158944644175636731276259209725098371838087123819800	E-21
22	+0.68643646352508711191432270744918317362603458545	E-23
23	−0.284256574469111777651159152046854758717720424200	E-24
24	+0.113058407111076004304848862173398476232482636	E-25
25	−0.432562503464726722509275688810050383133321600	E-27
26	+0.1594303550727540868696531707310462180990420	E-28
27	−0.566823521101793312322931179802386181485200	E-30
28	+0.194633920333296518189270656581092142968	E-31
29	−0.646229124983247735048330594076632304800	E-33
30	+0.20769361952496008134226200742564158700	E-34
31	−0.6468006833888916595363813806154635	E-36
32	+0.1953637154621609774410411946081270	E-37
33	−0.5728416548260786714221274260440	E-39
34	+0.1631967116412317817205131049760	E-40
35	−0.452087825190862406471430386500	E-42
36	+0.121870475741041745577919010	E-43
37	−0.3199269862798691799230335	E-45
38	+0.8184201602630949520562700	E-47
39	−0.2041533193530081420053	E-48
40	+0.4968887264260650874700	E-50
41	−0.118069998098071821800	E-51
42	+0.27405704303147504	E-53
43	−0.621721098490656	E-55
44	+0.13791940586607	E-56
45	−0.299324688845	E-58
46	+0.6358436106	E-60
47	−0.132264543	E-61
48	+0.2695332	E-63
49	−0.53854	E-65
50	+0.1035	E-66

Accuracy of coeff's 70-D; calculated functional error 9.25×10^{-40}.

For an expansion of order 50, we take

$$y_{50}(t) = \tau_1 Q_{50}(t) + \tau_2 Q_{49}(t) = \sum_{k=0}^{50} a_k^{50}\, T_k(t),$$

where the coefficients a_k^{50} are shown in Table 6 with 70-D accuracy. We have the relation

$$\operatorname{erf}(x) \simeq x \sum_{k=0}^{50} a_k^{50}\, T_k(t),$$

with a maximum absolute error of 9.25×10^{-40}.

4.6. The Complement of the Error Function

Following Schonfelder, the function under consideration here is

$$y(x) = xe^{x^2}(1 - \operatorname{erf}(x)) = xe^{x^2} \operatorname{erfc}(x), \qquad x \in (2, \infty).$$

Via the transformation

$$t = \frac{h - x^2}{h - 8 + x^2},$$

Table 7. Error function complement.

k	a_k^{50}	
0	+0.538988926036191575584167955173962869073545805	
1	−0.265598904091486733721465009038074979517077 11	E-1
2	−0.148707314669809950960504633348546909171 2592	E-2
3	−0.138040145414143859607708919699500678213751	E-3
4	−0.112803033322874914985073655627905335 22094	E-4
5	−0.117286984274372522405373897533975 9471431	E-5
6	−0.103476150393304615537381582464021001111	E-6
7	−0.118991140858924382544471338299376773 56	E-7
8	−0.101622254498949864047616642337969681 5	E-8
9	−0.137895716146965692169375099343944566	E-9
10	−0.936961303373730333520622231358513 4	E-11
11	−0.191880958395952534862859938324009 6	E-11
12	−0.375730172019937071037113966423 91	E-13
13	−0.370537260269833570044046417256 82	E-13
14	+0.262756542349037115306385394 54	E-14
15	−0.112132287643793254876147530086 5	E-14
16	+0.184136028922538037416853102622	E-15
17	−0.491302565748859684212355147 93	E-16
18	+0.107044551673729820258376055 56	E-16
19	−0.267189366240487206924837405 9	E-17
20	+0.649326867975758153967179783	E-18
21	−0.165399353182612030788224501	E-18
22	+0.426056266038381580660911 81	E-19
23	−0.112558407645489379761730 22	E-19
24	+0.302561744801491717426167 6	E-20
25	−0.829042146324786013284882	E-21
26	+0.231049558214612757451532	E-21
27	−0.654695114652185501531 64	E-22
28	+0.188423143309708302364 49	E-22
29	−0.550434070456013520684 7	E-23
30	+0.163094950142285043561 8	E-23
31	−0.489859977513249207109	E-24
32	+0.149054152484754741081	E-24
33	−0.459222242340945718 48	E-25
34	+0.143182466768833729 18	E-25
35	−0.451587213894408612 9	E-26
36	+0.144008602744040319 9	E-26
37	−0.464143968159607703	E-27
38	+0.151136054267363163	E-27
39	−0.497025153877879 68	E-28
40	+0.165020065395606 98	E-28
41	−0.552974533575428 1	E-29
42	+0.186962043680280 6	E-29
43	−0.637611507467724	E-30
44	+0.219267261772557	E-30
45	−0.760007884698 35	E-31
46	+0.265493808793 52	E-31
47	−0.939112276496 9	E-32
48	+0.344409637304 1	E-32
49	−0.135503084463 1	E-32
50	+0.39351631139	E-33

Accuracy of coeff's 45-D; calculated functional error 2.05×10^{-38}.

our domain of definition becomes $(-1, 1)$ and the problem is reduced to that of solving the linear differential equation

$$Dy(t) := p_1(t)\, y'(t) + p_2(t)\, y(t) = f(t), \qquad y(-1) = \frac{1}{\sqrt{\pi}}, \tag{11}$$

where the coefficients and the right hand side are defined by

$$\begin{aligned} p_1(t) &= \frac{1}{h-4}\left[h + (8+h)t + (16-h)t^2 + (8-h)t^3\right], \\ p_2(t) &= (1+2h) + (17-2h)\,t, \\ f(t) &= \frac{1}{\sqrt{\pi}}\left[2h + (16-2h)\,t\right]. \end{aligned}$$

The canonical polynomials $Q_0(t)$ and $Q_2(t)$ are undefined and we have

$$Q_1 = \frac{1}{2h-17}\left\{(2h+1)Q_0 - T_0\right\},$$

$$Q_3 = \frac{4}{h-8}\left\{-(h^2-13h+26)\,Q_0 + \left(2h^2 - \frac{27h}{4} + 10\right)Q_1 - (h^2-12h+26)\,Q_2 - (h-4)\,T_1\right\},$$

and for $n \geq 2$,

$$Q_{n+2} = \frac{4}{n(h-8)} \left\{ 16nQ_0 + 32n \sum_{k=1}^{n-3} Q_k + \frac{n}{4}(h+120)Q_{n-2} \right.$$
$$+ \left[\frac{n}{2}(h+48) - h^2 + \frac{25}{2}h - 34 \right] Q_{n-1} + \left[16n + 2h^2 - 7h - 4 \right] Q_n$$
$$\left. - \left[\frac{n}{2}(h-16) + h^2 - \frac{25}{2}h + 34 \right] Q_{n+1} - (h-4)T_n \right\}.$$

We choose $h = 10.5$ as in Schonfelder's paper [10]. Since there are two undefined canonical polynomials and one initial condition, we need to take 3 Tau parameters for the approximate expansion (see [2]); thus,

$$y_{50}(t) = \tau_1 Q_{50}(t) + \tau_2 Q_{49}(t) + \tau_3 Q_{48}(t) = \sum_{k=0}^{50} a_k^{50}\, T_k(t),$$

so that for $x \in (2, \infty)$, the following relation holds:

$$\operatorname{erfc}(x) \simeq \frac{e^{-x^2}}{x} \sum_{k=0}^{50} a_k^{50}\, T_k(t).$$

The coefficients $\{a_k^{50}\}$ are given with 45-D accuracy in Table 7, and we have a maximum error of 2.05×10^{-38} at $x = 2$.

4.7. Remarks

Figures 1–4 exhibit the behaviour of the logarithms of approximation coefficients $\log|a_k^n|$ for each function under consideration with respect to the indices k. We are not unaware of the fact that the graphs of the significant coefficients of Tables 3 and 7 are of some interest—see Figures 2 and 4. Since exponential convergence has been shown to hold for the Tau Method (see [17,20,21]), it follows that the approximations given in these tables, although correct to the specified accuracies, are being slowly corrupted by errors. The following table contains a summary of our results:

tbl.	$y(x)$	intrvl.	basis	ord.	transf.	order of the err.	acc.
1	e^x	$[0,1]$	$T^*(x)$	48		10^{-92}	95
2	e^{-x^2}	$[0,1]$	$T^*(x)$	48		10^{-54}	60
3	$-ze^z\mathrm{Ei}(-z)$	$(1,\infty)$	$T^*(x)$	48	$x = z^{-1}$	10^{-16}	25
4	$\sin \pi x$	$[0,0.5]$	$T^*(t)$	50	$t = 2x$	10^{-86}	90
5	$\cos \pi x$	$[0,0.5]$	$T^*(t)$	50	$t = 2x$	10^{-86}	90
6	$\operatorname{erf}(x)/x$	$(0,2)$	$T(t)$	50	$t = x^2/2 - 1$	10^{-39}	70
7	$xe^{x^2}\operatorname{erfc}(x)$	$(2,\infty)$	$T(t)$	50	$t = \dfrac{10.5 - x^2}{2.5 + x^2}$	10^{-38}	45

5. CONCLUSION

In this paper, we have developed 'user-controlled' precision subroutines for the four arithmetic operations which we have used to compute the coefficients of Chebyshev expansions for some basic mathematical functions. Our subroutines are written in Fortran, thus making them widely applicable. In producing the expansion coefficients, we have adopted the recursive formulation of the Tau Method. For a particular function, this formulation allows for the recursive generation of the canonical polynomials associated with it, which, in turn, are independent of the degree of expansion and the initial/boundary conditions.

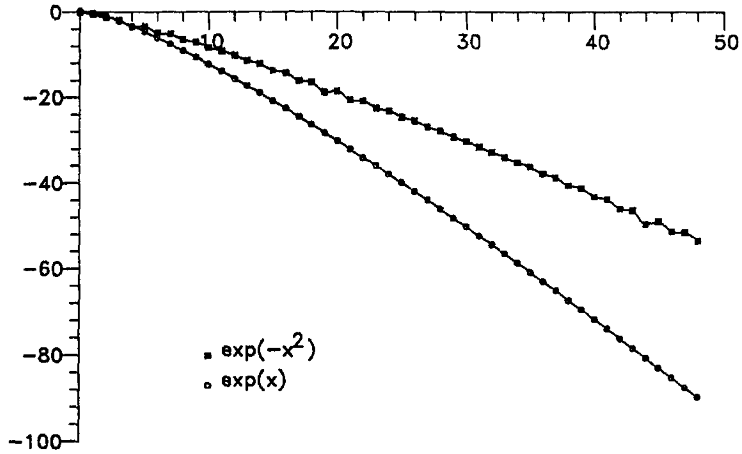

Figure 1. $\log|a_k^{48}|$ vs. k (see Tables 1 and 2).

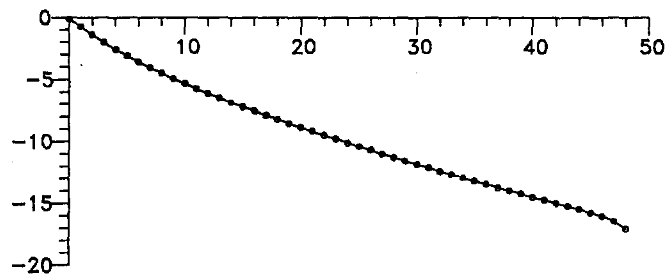

Figure 2. $\log|a_k^{48}|$ vs. k (see Table 3).

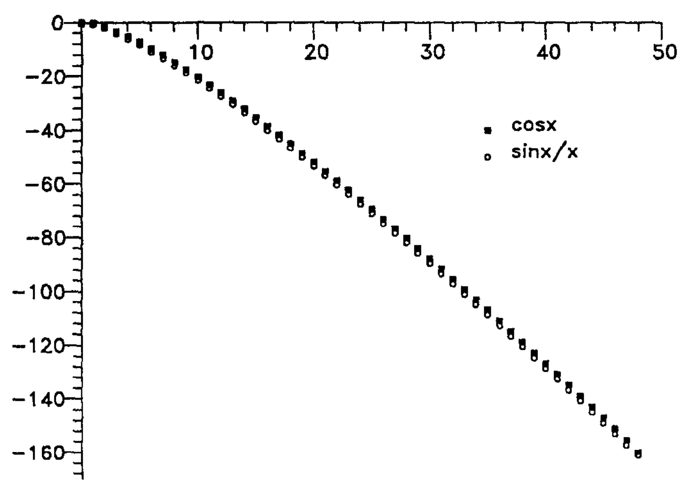

Figure 3. $\log|a_k^{50}|$ vs. k (see Tables 4 and 5).

Our choice of examples in Section 4 has been dictated by the availability of other error estimates (Tables 1–3) to compare with our own, or by the presence of other high precision tabulations to allow for comparison of the results (Tables 4–7). The orders of expansions (48 and 50) and the background accuracy (100-D) have been arbitrarily chosen, since our main objective was to test our algorithms and, for the first three cases, to illustrate the use of our error estimation technique developed in [1].

The maximum error in the determination of the coefficients of our Chebyshev expansions is controlled by the background accuracy we have selected (100-D); as we have shown, for the degrees n considered, the latter is far smaller than the approximation error of the relevant expansions. Consequently, in our tables we have truncated the coefficients to near the orders of magnitude compatible with their respective maximum errors.

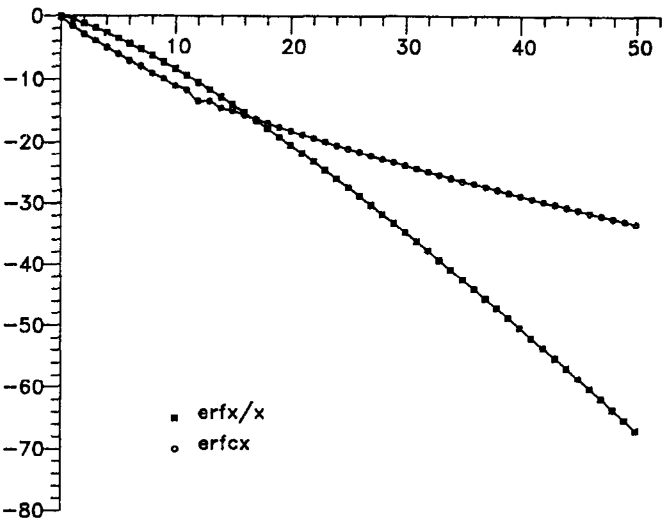

Figure 4. $\log|a_k^{50}|$ vs. k (see Tables 6 and 7).

REFERENCES

1. H.G. Khajah and E.L. Ortiz, Upper and lower error estimation for the Tau Method and related polynomial techniques, *Computers Math. Applic.* **22** (3), 81–87 (1991).
2. E.L. Ortiz, The Tau Method, *SIAM J. Numer. Anal.* **6**, 480–492 (1969).
3. E.L. Ortiz, Canonical polynomials in the Lanczos' Tau Method, In *Studies in Numerical Analysis*, (Edited by B.K.P. Scaife), pp. 73–93, Academic Press, New York, (1974).
4. R.P. Brent, A Fortran multiple-precision arithmetic package, *ACM Trans. Math. Softw.* **4** (1), 57–70 (1978).
5. D.M. Smith, A Fortran package for floating-point multiple-precision arithmetic, *ACM Trans. Math. Softw.* **17** (2), 273–283 (1991).
6. W.T. Wyatt, D.W. Lozier and D.J. Orser, A portable extended precision arithmetic package and library with Fortran precompiler, *ACM Trans. Math. Softw.* **2** (3), 209–231 (1976).
7. J.L. Schonfelder and J.T. Thomason, Arbitrary precision arithmetic in Algol 68, *Software-Practice and Experience* **9**, 173–182 (1979).
8. T.E. Hull and A. Abrham, Variable precision exponential function, *ACM Trans. Math. Softw.* **12** (2), 79–91 (1986).
9. T.E. Hull, A. Abrham, M.S. Cohen, A.F.X. Curley, C.B. Hall, D.A. Penny and J.T.M. Sawchuk, Numerical Turing, *ACM SIGNUM Newsl.* **20** (3), 26–34 (1985).
10. J.L. Schonfelder, Chebyshev expansions for the error and related functions, *Math. Comp.* **32**, 1232–1240 (1978).
11. J.L. Schonfelder, Very high accuracy Chebyshev expansions for the basic trigonometric functions, *Math. Comp.* **34**, 237–244 (1980).
12. C.W. Clenshaw, Chebyshev series for mathematical functions, In *National Physical Laboratory Mathematical Tables*, Volume 5, Her Majesty's Stationary Office, London, (1962).
13. E.L. Ortiz, W.F.C. Purser and F.J. Rodriguez Cañizarez, Automation of the Tau Method, Research report, Imperial College, (1972); Presented at the *Conference on Numerical Analysis*, Royal Irish Academy, Dublin.
14. A.A. Tsonis, The effect of truncation and round-off on computer simulated chaotic trajectories, *Computers Math. Applic.* **21** (8), 93–94 (1991).
15. Y.L. Luke, *The Special Functions and Their Approximations*, Volume I, Academic Press, (1969).
16. C. Lanczos, *Applied Analysis*, Prentice-Hall, New Jersey, (1956).
17. S. Namasivayam and E.L. Ortiz, A hierarchy of truncation error estimates for the numerical solution of a system of ordinary differential equations with techniques based on the Tau Method, In *Numerical Treatment of Differential Equations*, (Edited by K. Strehmel), pp. 113–121, Teubner, Leipzig, (1987).
18. S.L. Moshier, *Methods and Programs for Mathematical Functions*, Ellis Horwood Ltd, Chichester, England, (1989).
19. National Bureau of Standards, Tables of the exponential function e^x, Applied Math Series 14, Dept. of Commerce, U.S. Government, Washington, DC, (1951).
20. M.R. Crisci and E.L. Ortiz, Existence and convergence results for the numerical solution of differential equations with the Tau Method, Research Report NAS 23.04.81, Imperial College, (1981).
21. Ch. Cabos and E.L. Ortiz, On different formulations of the Tau Method and their convergence (to appear).