- Parent: 0. Numerical Evaluation of Special Functions[1]
- Previous: 0. Numerical Evaluation of Special Functions[2]
- Next: 2. Mathematical Developments[3]

Press here[4] to get the full document in PostScript format.
Press here[5] to get this subdocument in PostScript format.

NUMERICAL EVALUATION OF SPECIAL FUNCTIONS

D. W. LOZIER AND F. W. J. OLVER

ABSTRACT. This document is an excerpt from the current hypertext version of an article that appeared in Walter Gautschi (ed.), *Mathematics of Computation 1943–1993: A Half-Century of Computational Mathematics*, Proceedings of Symposia in Applied Mathematics 48, American Mathematical Society, Providence, RI 02940, 1994. The symposium was held at the University of British Columbia August 9–13, 1993, in honor of the fiftieth anniversary of the journal *Mathematics of Computation*.

The original abstract follows.

Higher transcendental functions continue to play varied and important roles in investigations by engineers, mathematicians, scientists and statisticians. The purpose of this paper is to assist in locating useful approximations and software for the numerical generation of these functions, and to offer some suggestions for future developments in this field.

1. Introduction

When automatic computers began to appear in the 1950s various confident, and often incorrect, predictions were made concerning the impact of these devices on applied mathematics, science and engineering. One of these predictions was that the need for special functions, or higher transcendental functions (as they are also known), would disappear entirely. This was based on the observation that the main use of these functions in those days was to approximate the solutions of classical partial differential equations: with automatic computers it would become possible to solve these equations by direct numerical methods. This observation is indeed correct; nevertheless, a perusal of current computational journals in the sciences reveals a persistent need for numerical algorithms to generate Airy functions, Bessel functions, Coulomb wave functions, error functions and exponential integrals—to name but a few of the classical special functions. Equally significantly, the National Bureau of Standards' Handbook of Mathematical Functions [**AS64**][6] continues to

---

[1] /nesf/Sect0/Sect0.html
[2] /nesf/Sect0/Sect0.html
[3] /nesf/Sect2/Sect2.html
[4] ftp://gams.nist.gov/pub/lozier/nesf/Paper.ps.Z
[5] ftp://gams.nist.gov/pub/lozier/nesf/Sect1.ps.Z

1991 *Mathematics Subject Classification.* Primary 65D20; Secondary 33-00.
The research of the second author has been supported by NSF Grant CCR 89-14933.

[6] An explanation of the scheme used for acronyms of references is given on p. 25.

be one of the best-selling mathematical books of all time[7].

The purpose of the present paper is to provide some assistance to those mathematicians, engineers, scientists, and statisticians who discover that they need to generate numerical values of the special functions in the course of solving their problems. "Generate" is the operative word here: we are thinking primarily of either software or numerical approximations that can be programmed fairly easily. Numerical tables are not covered in this survey. Furthermore, for the most part we shall concentrate on the functions themselves; only in certain cases do we include, for example, zeros, inverse functions or indefinite integrals. Elementary functions, also, are excluded[8]. Lastly, we believe that the majority of readers would prefer us to emphasize the more useful algorithms rather than make an attempt to be encyclopedic: algorithms or approximations that have clearly been superseded are omitted.

We identify three stages in the development of computational procedures for the special functions:

(i) Derivation of relevant mathematical properties.
(ii) Development of numerical approximations and algorithms.
(iii) Construction and testing of robust software.

Included in Stage (i) are asymptotic expansions, continued fractions, difference and differential equations, functional identities, integral representations, and Taylor-series expansions. Included in Stage (ii) are expansions in series of Chebyshev polynomials ("Chebyshev series"), minimax polynomial and rational approximations, Padé approximations, numerical quadrature, and numerical solution of difference and differential equations. **In this paper the emphasis will be on Stages (ii) and (iii), but in §2 we supply some general references for Stage (i).**

In §3 we make a general survey of software libraries and packages[9] that include collections of special functions.

In §§4 and 5 the functions are treated individually. We list software that is already available and readily programmable numerical approximations. In §6 we also include references to articles (or books) that may be useful for the testing or comparison of existing software, or in the construction of new libraries. We do not attempt what would be a herculean task of testing and comparing everything that is available. Our philosophy in this survey has to be that of *caveat emptor*: no algorithm, approximation or package that we mention should be relied upon to produce accurate output in the absence of evidence of independent and systematic checks.

As in the progress of other branches of numerical analysis, procedures used to evaluate special functions are influenced heavily by the computing equipment available at the time. In the era of desk-calculating machines the medium was a printed table of numerical values of the wanted function, or functions, generally for equispaced values of the arguments. Nontabular values were calculated by

[7] In 1988 the National Bureau of Standards became the National Institute of Standards and Technology.

[8] Methods for constructing and testing algorithms for generating elementary functions are surveyed in [**CW80**]. See also [**Bai93**,**Bre76**,**Bre78a**,**Bre78b**,**Cod93a**,**LCY65**,**MY91**,**Smi89**,**Smi91**,**Ziv91**].

[9] Certain company products are identified in the text. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose.

means of Lagrange's interpolation formula or central-difference interpolation formulas [**Fox56**]. In other words, local polynomial approximations were employed. These interpolation procedures were reasonably successful for functions of a single variable, but two-dimensional interpolation on desk-calculating machines was often a laborious computation that was prone to error. The daunting task that faced a (human) computer is summed up in the following quotation from the Introduction to Karl Pearson's tables of the incomplete gamma function [**Pea22**]:

"As a matter of fact, supposing the use of a machine, which every modern computer has at his command, no interpolation suggested ought to take more than an hour's work and many much less. If the user of these tables groans under that hour, let him compute *de novo* a function value, say $I(6.86877, 47.1813)$ —including of course $\Gamma(48.1813)$—to seven-figure accuracy, and when he has completed the task, we believe his feelings towards those who have provided him with these tables will be very sensibly modified."

With electronic computers, the number of arithmetic operations that could be contemplated for the generation of a single function value increased considerably. In consequence, high-degree global approximations appeared for functions of a single variable in the form of minimax polynomial or rational approximations, or truncated Chebyshev-series expansions; see, for example, [**Cle62,HCL+68**] and [**Luk77b**].

Chebyshev series in two dimensions also became feasible [**CP66,Luk71a,Luk71b,Luk72a**]. However, because of their more complicated asymptotic behavior, special functions of two variables cannot be covered comprehensively simply by use of polynomial or rational approximations or Chebyshev series. The situation is cloudier still when the variables or parameters are complex, or of course when they are more than two in number. For this reason, to achieve maximum speed, a comprehensive software package for generating a function of two (or more) variables typically employs several different algorithms in addition to, or quite commonly in place of, polynomial or rational approximations or Chebyshev series. The construction and testing of such a package invariably entails prodigious effort.

Computers continue to increase in sophistication and power; in consequence we should expect further changes in the algorithms used to generate the special functions. So far, the potential offered by the introduction of vector and parallel computing machines has not been exploited to any great extent. It might well lead to simplifications in the algorithms needed for many functions, as well as to an increase in execution speeds. We refer again to this possibility in the concluding section (§7).

In assembling the bibliography of this paper we have been assisted by the references collected and classified by the late Dr. L. W. Fullerton in his 1980 Bell Laboratories report [**Ful80**], by access to Dr. N. M. Temme's private collection of references, and by GAMS, the Guide to Available Mathematical Software prepared by the National Institute of Science and Technology [**BHKS90**] and also available online[10].

We have searched through issues of the following journals of the past twenty-five years for relevant references:

[10]To get started with the online GAMS system, `telnet gams.nist.gov` and login as `gams` or `xgams`. No password is required. Information on commercial and noncommercial software is provided, and electronic retrieval of noncommercial software is possible.

Applied Statistics (Appl. Statist.), *Association for Computing Machinery Transactions on Mathematical Software* (ACM Trans. Math. Software), *BIT*, *Collected Algorithms from the Association for Computing Machinery* (CALGO), *Communications of the Association for Computing Machinery* (Comm. ACM), *Computer Physics Communications* (Comput. Phys. Comm.), *Computing*, *Journal of Computational and Applied Mathematics* (J. Comput. Appl. Math.), *Journal of Computational Physics* (J. Comput. Phys.), *Mathematical Reviews* (Math. Rev.), *Mathematics of Computation* (Math. Comp.), *Numerische Mathematik* (Numer. Math.), *U.S.S.R. Computational Mathematics and Mathematical Physics* (U.S.S.R. Comput. Math. and Math. Phys.)[11], *Zeitschrift für Angewandte Mathematik und Physik* (Z. Angew. Math. Phys.), *Zentralblatt für Mathematik und ihre Grenzgebiete* (Zbl.).

However, because of the sheer volume and diversity of publications on special functions it is almost inevitable that we have overlooked some useful algorithms and important articles. In this event we tender, in advance, our apologies to the authors.

References

[AS64] M. Abramowitz and I. A. Stegun (eds.), *Handbook of mathematical functions with formulas, graphs and mathematical tables*, National Bureau of Standards Applied Mathematics Series, vol. 55, U. S. Government Printing Office, Washington, D. C., 1964.

[Bai93] D. H. Bailey, *Algorithm 719. Multiprecision translation and execution of Fortran programs*, ACM Trans. Math. Software **19** (1993), 288–319.

[BHKS90] R. F. Boisvert, S. E. Howe, D. K. Kahaner, and J. L. Springmann, *Guide to available mathematical software*, Tech. Report NISTIR 90-4237, National Institute of Standards and Technology, Center for Computing and Applied Mathematics, Gaithersburg, Maryland 20899, 1990.

[Bre76] R. P. Brent, *Fast multiple-precision evaluation of elementary functions*, J. Assoc. Comput. Mach. **23** (1976), 242–251.

[Bre78a] R. P. Brent, *Algorithm 524. MP, A Fortran multiple-precision arithmetic package*, ACM Trans. Math. Software **4** (1978), 71–81, for remark see same journal v. 5 (1979), pp. 518–519.

[Bre78b] R. P. Brent, *A Fortran multiple-precision arithmetic package*, ACM Trans. Math. Software **4** (1978), 57–70.

[Cle62] C. W. Clenshaw, *Chebyshev series for mathematical functions*, National Physical Laboratory Mathematical Tables, vol. 5, Her Majesty's Stationery Office, London, 1962.

[Cod93] W. J. Cody, *Algorithm 714. CELEFUNT: A portable test package for complex elementary functions*, ACM Trans. Math. Software **19** (1993), 1–21.

[CP66] C. W. Clenshaw and S. M. Picken, *Chebyshev series for Bessel functions of fractional order*, National Physical Laboratory Mathematical Tables, vol. 8, Her Majesty's Stationery Office, London, 1966.

[CW80] W. J. Cody and W. Waite, *Software manual for the elementary functions*, Prentice Hall, Englewood Cliffs, New Jersey, 1980.

[Fox56] L. Fox, *The use and construction of mathematical tables*, National Physical Laboratory Mathematical Tables, vol. 1, Her Majesty's Stationery Office, London, 1956.

[Ful80] L. W. Fullerton, *A bibliography on the evaluation of mathematical functions*, Computing Science Tech. Report No. 86, Bell Laboratories, Murray Hill, New Jersey 07974, 1980.

[HCL+68] J. F. Hart, E. W. Cheney, C. L. Lawson, H. J. Maehly, C. K. Mesztenyi, J. R. Rice, H. C. Thacher, Jr., and C. Witzgall, *Computer approximations*, John Wiley and Sons, Inc., New York, 1968.

[11] In 1991 this journal was retitled *Computational Mathematics and Mathematical Physics.*

[LCY65] L. A. Lyusternik, O. A. Chervonenkis, and A. R. Yanpol'skii, *Handbook for computing elementary functions*, International Series of Monographs in Pure and Applied Mathematics, vol. 76, Pergamon Press, Oxford, 1965.

[Luk71a] Y. L. Luke, *Miniaturized tables of Bessel functions*, Math. Comp. **25** (1971), 323–330.

[Luk71b] Y. L. Luke, *Miniaturized tables of Bessel functions. II*, Math. Comp. **25** (1971), 789–795, for corrigendum see same journal v. 26 (1972), Microfiche Supplement A1–A7.

[Luk72] Y. L. Luke, *Miniaturized tables of Bessel functions. III*, Math. Comp. **26** (1972), 237–240, with Microfiche Supplement.

[Luk77] Y. L. Luke, *Algorithms for the computation of mathematical functions*, Academic Press, New York, 1977.

[MY91] P. Midy and Y. Yakovlev, *Computing some elementary functions of a complex variable*, Math. Comput. Simulation **33** (1991), 33–49.

[Pea22] K. Pearson, *Tables of the incomplete* Γ*-function*, H. M. Stationery Office, London, 1922, reissued by Biometrika, 1934.

[Smi89] D. M. Smith, *Efficient multiple-precision evaluation of elementary functions*, Math. Comp. **52** (1989), 131–134.

[Smi91] D. M. Smith, *Algorithm 693. A Fortran package for floating-point multiple-precision arithmetic*, ACM Trans. Math. Software **17** (1991), 273–283.

[Ziv91] A. Ziv, *Fast evaluation of elementary mathematical functions with correctly rounded last bit*, ACM Trans. Math. Software **17** (1991), 410–423.

Applied and Computational Mathematics Division, National Institute of Standards and Technology, Gaithersburg, Md 20899

E-mail address: dlozier@nist.gov

Institute for Physical Science and Technology, University of Maryland, College Park, MD 20742

E-mail address: olver@bessel.umd.edu