Numerical evaluation of the confluent hypergeometric function for complex arguments of large magnitudes *

Mark Nardin

Electrical Engineering Department, Princeton University, Princeton, NJ 08544, United States

W.F. Perger and Atul Bhalla

Electrical Engineering and Physics Departments, Michigan Technological University, Houghton, MI 49931,
United States

Received 2 August 1990
Revised 2 March 1991

Abstract


A numerical evaluator for the confluent hypergeometric function for complex arguments with large magnitudes using a direct summation of Kummer’s series is described. Extended precision subroutines using large integer arrays to accumulate a single numerator and denominator are ultimately used in a single division to arrive at the final result. The accuracy has been verified through a variety of tests and they show the evaluator to be consistently accurate to thirteen significant figures, and on rare occasion accurate to only nine for magnitudes of the arguments ranging into the thousands in any quadrant in the complex plane. Because the evaluator automatically determines the number of significant figures of machine precision, and because it is written in FORTRAN 77, tests on various computers have shown the evaluator to provide consistently accurate results, making the evaluator very portable. The principal drawback is that, for certain arguments, the evaluator is slow; however, the evaluator remains valuable as a benchmark even in such cases.

Keywords: Confluent hypergeometric function, numerical evaluation, Bessel functions, Coulomb functions, Hankel functions, performance evaluation.

1. Introduction

The confluent hypergeometric function (CHF) is a solution of the differential equation

\[ zf''(z) + (\gamma - z)f'(z) - \alpha f(z) = 0, \]

(1)

* This work was supported in part by a grant from the National Science Foundation, CCR-8823031.
where $\alpha$, $\gamma$ and $z$ may all be complex. An exact solution of this equation is given by Kummer’s series [1]. Previously, summing this series on a computer to obtain a solution when the magnitude of $z$ was large and either complex or negative proved difficult due to limited computer precision. For example, if $\alpha$ and $\gamma$ are equal to each other and $z = 0 + 140i$, the series will grow to an intermediate value on the order of $10^{60}$. But, as the terms cancel each other, the final solution of the series would have a magnitude of about one. This means that at least 60 digits of precision are required just to get one digit of accuracy for the solution.

The typical solution to this problem has been to use an asymptotic expansion for large $z$, such as that given in [5]. When comparing the exact series expansion to the asymptotic expansion in the transition regions for $z$, however, we found that there were many values of the arguments for which the CHF was not converging. It was found that “large $z$” could range from 10 or less to a thousand or more depending on the values of the arguments $\alpha$ and $\gamma$.

The purpose of this paper is to describe a way of using arrays to obtain extra precision on a computer and allow the use of the exact solution of Kummer’s series for complex arguments with magnitudes ranging into the thousands. The precision and limits of the evaluator are discussed along with methods for verifying its accuracy.

2. Obtaining extended precision using arrays

After resorting to double precision and still needing more digits in order to get a final answer that is more accurate than an asymptotic expansion, an alternative is to explicitly keep track of the digits. Using arrays, we split every number up into groups of eight digit integers, keeping each eight digits in consecutive array positions and also keeping track of the sign and exponent of each number separately. Once the numbers were represented by arrays, subroutines had to be written that would perform complex addition, subtraction, division and multiplication using these arrays.

Kummer’s series, defined as

$$\, _1F_1(\alpha, \gamma, z) = 1 + \frac{\alpha z}{\gamma} + \frac{\alpha(\alpha + 1)z^2}{\gamma(\gamma + 1)2!} + \cdots + \frac{(\alpha)_n z^n}{(\gamma)_n n!} + \cdots,$$

(2)

where

$$(\alpha)_n = \alpha(\alpha + 1)(\alpha + 2) \cdots (\alpha + n - 1),$$

(3)

was used, as it will produce an exact solution as $n \to \infty$. However, it cannot be used in this standard form using conventional complex floating-point arithmetic. As given, Kummer’s formula would require division to be performed in order to get each current term to be added to the running sum, and given two arbitrary numbers, the division of the two cannot necessarily be represented as an exact number. But, multiplication of two exact numbers can always be represented exactly. Therefore, a form of Kummer’s equation which requires only one division to obtain the final answer is used.

We calculated Kummer’s series by keeping track of the numerator and denominator separately as each term is added, and only performing one division of the numerator by the denominator to obtain the final answer. This is done by keeping a common denominator for the total sum and the current term. Since the denominator of each term in the series is the same as
the denominator of the previous term multiplied by \((\gamma + n - 1) \times n\), a common denominator can always be obtained by multiplying the current sum of numerators, along with the total denominator, by this term. For illustration, the first, second and third terms would have numerators/denominators like (with the 0th term being 1):

\[\frac{\gamma + \alpha z}{\gamma},\]

\[\frac{(\gamma + \alpha z)(\gamma + 1) + \alpha(\alpha + 1)z^2}{2\gamma(\gamma + 1)},\]

\[\frac{[(\gamma + \alpha z)(\gamma + 1) + \alpha(\alpha + 1)z^2](\gamma + 2) + \alpha(\alpha + 1)(\alpha + 2)z^3}{(2)(3)\gamma(\gamma + 1)(\gamma + 2)}.

Since the numerator and denominator are kept track of separately, and since all of the variables (\(\alpha, \gamma, z\) and \(n\)) are numbers which are represented by exact decimals, there are no rounding errors made in any steps which can be propagated through the series. However, the numerator and denominator tend to get quite large, as they contain factorials, but since the exponent and all of the digits of the numbers are taken care of explicitly, the overflow is effectively controlled.

Typically, floating-point processors introduce an error in the least-significant digit of a number. Therefore, 5.1 could be stored internally as 5.100 000 000 000 000 009. Because of the large number of terms (and intermediate multiplications) required for the series to converge, this seemingly small error was found to contribute to an unacceptably large error as the number of terms needed from Kummer's function grew.

In order to circumvent this representation error, upon calling the program, the number of available bits is determined. Each element in the arrays then uses only half of the available bits, thus eliminating the possibility of error upon multiplication of any two elements. For example, if a machine has \(n\) available bits, when two numbers are multiplied together, the result is a number with \(2n\) bits. In order to prevent any rounding errors when multiplying, therefore, the numbers that are being multiplied together can only have at most \(\frac{n}{2}\) bits. Furthermore, since decimal numbers could not be represented exactly, all of the variables used in Kummer's series (\(\alpha, \gamma, z\) and \(n\)) are first multiplied by a constant and then made into integers before any calculations are started. The constant depends upon the number of available bits. The multiplication by this constant will just introduce a scaling factor at the end to reach the final answer.

3. Validation of results

A problem encountered previously in evaluating the confluent hypergeometric function has been for values of large imaginary \(z\). Since we could find no previously published results for this case, no data was available for comparison and other tests were used. These tests were:

1. evaluation of Bessel functions;
2. evaluation of the Coulomb wave function;
3. exponential test;
(4) comparison with Kummer's first formula, also known as the Kummer transformation;
(5) comparison with tables obtained for real values of arguments;
(6) evaluation of the Wronskian for the CHF;
(7) comparison with the asymptotic expansion for large $z$;
(8) recurrence relationship.

It is important to note that the critical testing of a mathematical function evaluator typically requires an effort equal to the writing of the evaluator and this evaluator was no exception. The above tests were specifically chosen because one must test the evaluator for its absolute accuracy over a wide range of arguments. Because of the paucity of results for the CHF with large, complex arguments, we selected two distinct types of tests: absolute and relative. The absolute tests ((1)–(3), (5) and (7)) were extremely useful in establishing the absolute accuracy of the CHF evaluator over a limited range of arguments. The relative tests ((4), (6) and (8)) serve to demonstrate the accuracy of the CHF evaluator over a very broad range of arguments. The exponential test served as a pivotal test to connect these two types because it is capable of being evaluated over a wide range of complex arguments and it is necessary for the Kummer transformation test (test (4)).

In originally testing the program, the results were compared against tables given for the CHF [1,5]. Unfortunately, these tables are currently only published for real values of the arguments, but from all comparisons with given tables, accuracy was observed to eight significant figures, the number of significant figures provided by the tables.

The Coulomb wave function and the Bessel function were also evaluated using tables, along with the Hankel function [1,3,7]. Using the table of zeros for the Bessel function, the arguments of the CHF were tested with $\alpha$ and $\gamma$ ranging from $\frac{1}{2}$ to 60, and $z$ ranging from 0i to 340i, $z$ being pure imaginary, and the CHF evaluator agreed in all cases to all significant figures tabulated. Using the Coulomb function, the arguments were: $\alpha$ from 1 to 1 – 20i, $\gamma$ equal to 2 and $z$ from 0i to 40i, and the agreement was again to all significant figures tabulated. By far the most dramatic results, however, were achieved through evaluating the Hankel function, since this was the first table [7] which supplied complex arguments along with arguments of magnitudes in the thousands. This test pushed the limits of our arguments out to 2000 + 2000i for $\alpha$, 4000 + 4000i for $\gamma$ and 1000 + 3200i for $z$. The results compared to four significant figures, the number of significant digits of the table.

Three methods of testing that allowed us to confirm a much greater range were the exponential function
\[ \exp(z) = _1F_1(\alpha, \alpha, z), \]  

(4)

Kummer's first formula
\[ _1F_1(\alpha, \gamma, z) = \exp(z) _1F_1(\gamma - \alpha, \gamma, -z), \]

(5)

and the recurrence relation
\[ (\gamma - \alpha) _1F_1(\alpha - 1, \gamma, z) + (2\alpha - \gamma + z) _1F_1(\alpha, \gamma, z) - \alpha _1F_1(\alpha + 1, \gamma, z) = 0. \]

(6)

Ten thousand random numbers were chosen using an IMSL® subroutine, GGUBFS (seed = 2), for that purpose. These random numbers, uniform over the interval [0, 1], were then rescaled to the interval [0, 1000] for the magnitude and [0, 2\pi] for the phase of the three arguments $\alpha$, $\gamma$ and $z$, yielding 1665 different sets of arguments. These sets were then used in (4)–(6) and a
Table 1
Number of cases for which agreement was observed to the indicated number of significant figures for the exponential test, Kummer's transformation and recurrence test

<table>
<thead>
<tr>
<th>Number of significant figures</th>
<th>Exponential test</th>
<th>Kummer's transformation</th>
<th>Recurrence test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Real part</td>
<td>Imaginary part</td>
<td>Real part</td>
</tr>
<tr>
<td>10 or less</td>
<td>0</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>72</td>
<td>9</td>
</tr>
<tr>
<td>12</td>
<td>14</td>
<td>375</td>
<td>58</td>
</tr>
<tr>
<td>13</td>
<td>182</td>
<td>643</td>
<td>367</td>
</tr>
<tr>
<td>14</td>
<td>1026</td>
<td>328</td>
<td>785</td>
</tr>
<tr>
<td>15</td>
<td>357</td>
<td>134</td>
<td>290</td>
</tr>
<tr>
<td>16 or more</td>
<td>86</td>
<td>107</td>
<td>154</td>
</tr>
<tr>
<td>Average significant figures</td>
<td>14.20</td>
<td>13.20</td>
<td>14.00</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>0.73</td>
<td>1.22</td>
<td>0.99</td>
</tr>
</tbody>
</table>

program was written that would count the number of significant figures of comparison. Thirteen significant figures were typically observed, with the worst-case performance being two instances of nine significant figures. Table 1 lists the distribution of number of cases (out of the 1665) satisfying equations (4)–(6), respectively, for the specified number of significant digits.

It might be noticed that (4) and (5) were very special cases of the confluent hypergeometric function, with the exponential test appearing to be an extremely special case. From studying the originally defined Kummer's series, one can further notice that if $\alpha$ and $\gamma$ are equal, they will cancel each other in each of the terms, leaving the series which describes the exponential function. But, since our application of this series never calculates the ratio until the last step, no cancellation of terms occurs in any intermediate steps to make the solving any simpler. Consequently, allowing $\alpha$ and $\gamma$ to be equal provides no advantage that any general choice for $\alpha$ and $\gamma$ would provide, thereby making the exponential test a more stringent test of the CHF evaluator.

Another test of the CHF evaluator was made by calculating its Wronskian, defined as

$$W = \frac{1}{\gamma} \int F_1(\alpha, \gamma, z),$$

where

$$\phi = z^{1-\gamma} 1 F_1(\alpha - \gamma + 1, 2 - \gamma, z),$$

$$\phi' = \frac{1}{\gamma} \int F_1(\alpha + 1, \gamma + 1, z),$$

$$\phi' = \frac{(1 - \gamma)}{z^{1-\gamma}} 1 F_1(\alpha - \gamma + 1, 2 - \gamma, z) + z^{1-\gamma} \frac{\alpha - \gamma + 1}{2 - \gamma} 1 F_1(\alpha - \gamma + 2, 3 - \gamma, z).$$

This is a useful test of the relative type described earlier for showing that the evaluator is returning accurate answers for values of the arguments where no tables have been published.
### Table 2
Values of the \( \text{I}_F(\alpha, \gamma, z) \) evaluator described here and an asymptotic expansion for \( \alpha = -15 + 55i \), \( \gamma = 20 + 25i \) and \( z \) as given

<table>
<thead>
<tr>
<th>( z )</th>
<th>This work (^a)</th>
<th>( N ) (^b)</th>
<th>Asymptotic expansion (^c)</th>
<th>( N ) (^d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-100 + 0i)</td>
<td>(-1.43854938 \cdot 10^{-6})</td>
<td>269</td>
<td>(-1.46574500 \cdot 10^{-6})</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>(-6.90078120 \cdot 10^{-5})</td>
<td></td>
<td>(-6.80243508 \cdot 10^{-5})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 25i)</td>
<td>(6.41049217 \cdot 10^{-1})</td>
<td>284</td>
<td>(6.41128500 \cdot 10^{-2})</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>(-1.28406883 \cdot 10^{-2})</td>
<td></td>
<td>(-1.28466475 \cdot 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 50i)</td>
<td>(3.16110920 \cdot 10^{-2})</td>
<td>313</td>
<td>(3.16108058 \cdot 10^{-2})</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>(-3.86769676 \cdot 10^{-2})</td>
<td></td>
<td>(-3.86778368 \cdot 10^{-2})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 75i)</td>
<td>(1.68928954 \cdot 10^{-5})</td>
<td>354</td>
<td>(1.68927489 \cdot 10^{-5})</td>
<td>58</td>
</tr>
<tr>
<td></td>
<td>(-8.23211903 \cdot 10^{-5})</td>
<td></td>
<td>(-8.23211231 \cdot 10^{-5})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 100i)</td>
<td>(-2.52328037 \cdot 10^{-7})</td>
<td>395</td>
<td>(-2.52327858 \cdot 10^{-7})</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>(-5.51419146 \cdot 10^{-7})</td>
<td></td>
<td>(-5.51418964 \cdot 10^{-7})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 125i)</td>
<td>(-1.09699230 \cdot 10^{-8})</td>
<td>449</td>
<td>(-1.09699215 \cdot 10^{-8})</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>(-9.27940664 \cdot 10^{-9})</td>
<td></td>
<td>(-9.27940824 \cdot 10^{-9})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 150i)</td>
<td>(-6.92773735 \cdot 10^{-10})</td>
<td>500</td>
<td>(-6.92773755 \cdot 10^{-10})</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>(-4.92600433 \cdot 10^{-10})</td>
<td></td>
<td>(-4.92600497 \cdot 10^{-10})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 175i)</td>
<td>(-5.85414031 \cdot 10^{-11})</td>
<td>559</td>
<td>(-5.85414066 \cdot 10^{-11})</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>(-8.12117429 \cdot 10^{-11})</td>
<td></td>
<td>(-8.12117463 \cdot 10^{-11})</td>
<td></td>
</tr>
<tr>
<td>(-100 + 200i)</td>
<td>(2.31145634 \cdot 10^{-12})</td>
<td>617</td>
<td>(2.31145590 \cdot 10^{-12})</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>(-1.96169650 \cdot 10^{-11})</td>
<td></td>
<td>(-1.96169655 \cdot 10^{-11})</td>
<td></td>
</tr>
</tbody>
</table>

\(^a\) The values given are accurate to all digits shown.
\(^b\) Number of terms required for convergence to 10 significant figures using Kummer’s series (equation (2)).
\(^c\) Values given are not necessarily correct. They are provided to show areas of agreement and disagreement between methods.
\(^d\) Number of terms required in the series expansion of equation (12) to achieve “convergence” (equation (12) is not absolutely convergent) to 10 significant figures using quad precision (COMPLEX*32).

Although the Wronskian of a function is, in general, considered to be of great use in checking tables of functions \([2]\), we were unable to use the Wronskian test over the same 1665 cases used on the previously mentioned test. The problem encountered was due to limited precision. The Wronskian is the difference of two products. If each of these products is accurate to at least \( n \) digits, but if the difference of them is more than \( n \) orders of magnitude less than the individual terms, then the final answer will not necessarily be accurate to even one digit. So, evaluating the Wronskian was useful, but could not verify the CHF everywhere.

Another important test was made by comparing our results with those obtained through evaluating the asymptotic expansions for large \( z \) \([5]\). How large \( z \) has to be before the asymptotic expansions are valid seems greatly dependent upon the values of \( \alpha \) and \( \gamma \). But, for all values of \( \alpha \) and \( \gamma \) tested, as the magnitude of \( z \) increased, the agreement between our evaluation of the CHF and the asymptotic expansion always improved. For large enough \( z \), agreement could always reach to at least eight significant digits, providing an excellent verification of our evaluator for large values of complex \( z \). Table 2 provides a comparison of
the CHF evaluator described here and the large $z$ asymptotic expansion [5]
\[
\begin{align*}
_1F_1(\alpha, \gamma, z) &\approx \frac{\Gamma(\gamma)}{\Gamma(\gamma - \alpha)} e^{\pi i \alpha} z^{-\alpha} \left( \sum_{n=0}^{R} \frac{(\alpha)_n (\alpha - \gamma + 1)_n}{n!} (-z)^{-n} + \mathcal{O}(|z|^{-(R+1)}) \right) \\
&+ \frac{\Gamma(\gamma)}{\Gamma(\alpha)} z^{\alpha - \gamma} \left( \sum_{n=0}^{S} \frac{(\gamma - \alpha)_n (1 - \alpha)_n}{n!} z^{-n} + \mathcal{O}(|z|^{-(S+1)}) \right),
\end{align*}
\]
(12)
where $\epsilon = +1$ if $0 < \arg z < \pi$, $\epsilon = -1$ if $-\pi < \arg z \leq 0$, $\text{real}(z) < 0$, $z \to \infty$, and the terms in the numerator of the sum are as defined in (3). For this test, $\alpha = -15 + 55i$, $\gamma = 20 + 25i$ and $z = -100 + 0i$ to $z = -100 + 200i$. As can be seen from this table, the magnitude of $z$ had to be $180$ ($-100 + 150i$) before agreement was observed to seven significant figures and note that quad precision (COMPLEX * 32) was used on the IBM 4381 to sum the asymptotic series. Of course, by itself, this test does not prove that for $|z| < 150$, the CHF evaluator was correct and the asymptotic expansion incorrect. However, given the tests made on the CHF evaluator described earlier and the fact that for $|z| > 175$ the CHF evaluator shows excellent agreement with the asymptotic expansion, it is one more highly suggestive piece of evidence that the CHF evaluator is uniformly accurate.

4. Time requirements

The most outstanding drawback of using this evaluator for computing the CHF is its time requirement. Because we are doing calculations with each element of large arrays, and we are looping through enough terms for the CHF to converge, possibly tens of thousands of terms, the time required to obtain an answer can sometimes be quite large. Timings for the evaluator ranged from taking several milliseconds to thirty minutes. These tests were made on a 32-bit IBM 4381 and the Sun SPARC Station 4/65. Using computers with faster clock speeds could obviously reduce the run time. For example, preliminary results have shown the IBM RS6000 Model 520 to yield execution times roughly four times smaller than those mentioned above.

5. Conclusions

Our solution to the CHF utilizes Kummer’s series solution in a manner not previously done. Since this series is an exact solution, more precision can be obtained than with asymptotic expansions.

From all tests performed so far, at least nine significant figures of accuracy have been obtainable in the final answer. Also, this accuracy could be further improved. If a 64-bit machine is being used, about twice as much precision could be obtained.

The main strengths of this evaluator are its portability, range and accuracy. Regarding portability, we have successfully ported and tested the CHF evaluator on the IBM 4381, IBM 3090, IBM RS6000 Model 520, Sun SPARC Station 4/65, IBM PC, and the Sun 3/60. Previously, the most comprehensive program we could find on this subject still had regions of difficulty where an answer could not be obtained [6]. This evaluator has no such drawbacks. We
have shown that the answers obtained are correct for the magnitude of the arguments ranging into the thousands at any angle in the complex plane.

Of course, this evaluator is not optimal for all uses. Because of its run time, its use would sometimes be prohibitive in programs which would have to make multiple calls to it. For such cases, a straightforward calculation of Kummer's series or an asymptotic expansion should be used, with this evaluator as a check to make sure that $z$ is large enough to use the asymptotic expansion. As such, this evaluator is an excellent benchmarking program.

There are two times when this evaluator may be the only alternative. The first is when large complex values of $z$ are being used, but more precision is needed than can be obtained from an asymptotic expansion. (The magnitude of $z$ at which a series expansion will fail depends upon the precision of the computer being used and the values of $\alpha$ and $\gamma$.) Second, when using a large range of arguments is required, and the accuracy of the asymptotic expansion is in question, our routine should be used. From our experience, there appears to be no general statement as to how large $z$ must be before the asymptotic expansion returns acceptable results. The value of $z$ can range from 10 or less to 1000 or more.

6. Further work

We are currently planning on developing an evaluator for the generalized hypergeometric function $\,_{p}F_{q}$, using the subroutines developed for $\,_{1}F_{1}$. This will be a more challenging problem due to the slow rate of convergence of, for example, $\,_{2}F_{1}$. Nevertheless, owing to the manner in which we have attacked $\,_{1}F_{1}$, it appears likely that a multi-processing approach, while not addressing the convergence issue, holds promise for improving the speed of execution. The $\,_{1}F_{1}$ program is available from ACM-TOMS.

Acknowledgements

The authors gratefully acknowledge the support of IBM Corporation, grant RSP1057, for access to the IBM 3090 vector facility at the Los Angeles Scientific Center and the generous allocation of computing time on the IBM 4381 at Michigan Technological University. Also, the discussions that we had with Dr. W. Cody and Dr. D. Amos contributed directly to the testing of the evaluator.

References