

Browser-Native Bessel Functions: The Series That Had Already Converged to the Wrong Answer

JJ Ben-Joseph^{1,*}

¹TensorSpace, Inc, Tel Aviv, Israel

*jjbj@tensorspace.ai

September 22, 2026

Interactive calculator: <https://agentcalc.com/bessel-function-calculator>

Abstract

The power series for the Bessel function of the first kind converges for every finite argument. In IEEE-754 double precision it does not, once the terms have grown and cancelled. A browser calculator summed at most 120 terms of $J_n(x)$ and stopped when a term fell below 10^{-14} , then printed the sum whenever that sum was finite. For $J_0(40)$ the largest term is $1.8575931153394045 \times 10^{15}$. The terms do become smaller than 10^{-14} , after 68 of them, and the sum is finite: -0.09012077265738101 . The function is $+0.007366890584237339$. The stopped sum has the wrong sign and is 12.233 times too large in magnitude. For $J_0(50)$ the same test returns 655.2869784264319. Bessel's integral representation says that $|J_n(x)| \leq 1$ for real x and integer n , so 655 is not a rounding of a Bessel value. It is 11740.9 times the function. A second failure of the same test is easy to miss on a six-decimal display. At order 50 and argument 10 every term is far below 10^{-14} , the sum stops after two terms, and the value $1.4887731105862523 \times 10^{-30}$ is 16.6 percent below the series. The replacement keeps the power series only while its largest term is at most 10^4 . Past that point it uses Miller's backward recurrence, normalized by the Jacobi-Anger identity $J_0(x) + 2 \sum_{k \geq 1} J_{2k}(x) = 1$, and it reports the change in the value when the recurrence is started 24 orders higher. The default $J_0(1)$ is unchanged, 0.7651976865579666. At argument 40 the recurrence starts at order 107 and the two starting orders differ by 1.81×10^{-16} . A start only five orders past the argument still returns 0.008457.

1 Introduction

Bessel functions of the first kind are the bounded solutions of Bessel's equation. They are the radial factors for the wave equation, the heat equation, and the Helmholtz equation in cylindrical coordinates, which is why a calculator for $J_n(x)$ sits next to the drum, the waveguide, and the circular membrane [1, 2]. The function is old. Bessel's 1824 memoir on planetary perturbations already expands the quantities that became these series [3]. The numerical question on a web page is newer, and it is narrower: given an integer order and a real argument, print a number that is $J_n(x)$, and do not print a number that merely came from a loop that stopped.

The previous AgentCalc page took the power series, multiplied the next term by $-(x/2)^2/(k(k+n))$, and broke when the absolute value of that term dropped below 10^{-14} , or after 120 terms. If the sum was not a finite floating-point number it said the series had overflowed. If the sum was finite it printed six decimals, or an exponential when the absolute value was at least 10^6 . Nothing in that test asks whether the terms had been large along the way. A term of size 10^{-15} at the end of an

alternating sum whose middle terms were 10^{15} is not evidence that the sum has 14 correct decimals. It is evidence that the terms eventually got small, which the mathematical series always does.

This note documents the [Bessel function calculator](#) after that test was retired as a certificate of correctness. The series is still the method when its largest term is at most 10^4 . Otherwise the value is a backward recurrence in the order, checked by running the recurrence from two starting orders. The benchmarks are the classical value $J_0(1)$, the first positive zero of J_0 , the sign error at argument 40, the bound violation at argument 50, and the two-term truncation at order 50.

2 The series in exact arithmetic

For integer $n \geq 0$ the function is an entire function of x , given by the series

$$J_n(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{k!(k+n)!} \left(\frac{x}{2}\right)^{2k+n}, \quad (1)$$

which is the case $\nu = n$ of the standard definition [2, eq. 9.1.10]. The Digital Library of Mathematical Functions records the same expansion as (10.2.2) [4]. The ratio of successive terms is

$$\frac{t_{k+1}}{t_k} = -\frac{(x/2)^2}{(k+1)(k+1+n)}. \quad (2)$$

The factorials in the denominator win for every fixed x , so the terms tend to zero and the series converges. That fact does not say that the largest term is moderate. The index of the largest term is near the k where $|t_{k+1}/t_k| \approx 1$, that is, near $k \approx |x|/2$. For $J_0(40)$ that index is near 20, and the term there has size $1.8575931153394045 \times 10^{15}$. For $J_0(50)$ it has size $3.278755830813472 \times 10^{19}$.

The same recurrence (2) is what the page evaluates. It does not compute $k!$ and $(k+n)!$ as separate products and then divide two enormous numbers. It starts from $t_0 = (x/2)^n/n!$, built by successive multiplication by $(x/2)/i$, and then applies (2). In exact arithmetic the two procedures are the same series. In floating point they both fail when a term overflows the format, and the successive-product form fails more quietly when the terms remain finite and cancel.

Two evaluations of (1) are easy, and the previous page got both of them right. At the origin, $J_0(0) = 1$ and $J_n(0) = 0$ for $n \geq 1$. At argument 1, the terms decrease from the first one, the largest term of J_0 is 1, and eleven terms produce

$$J_0(1) = 0.7651976865579666, \quad (3)$$

which agrees with a 60-digit evaluation of (1) through the last bit of this double. The companion value is $J_1(1) = 0.44005058574493355$. These are the numbers a six-decimal table is rounding. They are not evidence about argument 40.

The derivative identity $J'_0(x) = -J_1(x)$ [2, eq. 9.1.28] turns a root of J_0 into a Newton iteration that uses only the same evaluator. Started at 2.4, that iteration reaches 2.404825557695773 and then stops. At that argument the series returns $-9.515724379496709 \times 10^{-17}$. The first positive zero of J_0 has been tabulated to 19 significant figures [5]. The calculator does not need the table to see that its own series is still honest there: the largest term is 1.446, and the returned value is a unit roundoff. The series has not yet been asked to cancel fifteen digits.

3 What double precision does with the series

A double-precision number has a 53-bit significand. Adding a number of size 10^{15} to a running sum, and later subtracting a nearly equal number, leaves an absolute error on the order of $10^{15} \times 2^{-53}$,

n	x	stopped sum		$J_n(x)$	terms	largest $ t_k $
0	1	0.7651976865579666	0.7651976865579666	9	1	
0	20	0.16702466426290605	0.16702466434058313	40	7.594×10^6	
0	30	-0.08638030775692472	-0.08636798358104028	54	1.121×10^{11}	
0	40	-0.09012077265738101	0.007366890584237339	68	1.858×10^{15}	
1	40	0.01028773273447872	0.12603831803758503	67	1.858×10^{15}	
0	50	655.2869784264319	0.05581232766925194	81	3.279×10^{19}	
20	50	86.21406388287797	-0.1167043527595798	71	6.335×10^{17}	
50	10	$1.4887731105862523 \times 10^{-30}$	$1.7845136078715964 \times 10^{-30}$	2	2.920×10^{-30}	

Table 1: The power series (1) summed in double precision until a term fell below 10^{-14} . “Terms” counts the terms the stopped sum actually used. The last column is the largest absolute term seen by that sum. The reference values are 60-digit evaluations of the same series, and they are the values the replacement calculator returns.

which is about 0.2. The function $J_0(40)$ is about 0.007. An absolute error of 0.2 is not a small relative error in that function. It is a different number, and it can have the other sign.

Table 1 is that comparison. The column headed “stopped sum” is the previous algorithm: at most 120 terms, and a break when the current term is smaller than 10^{-14} in absolute value. The column headed $J_n(x)$ is a 60-digit evaluation of (1), printed to the digits that the double-precision replacement actually returns. At argument 20 the stopped sum is still close. The largest term is $7.594058428126623 \times 10^6$, the stopped sum is 0.16702466426290605, and the function is 0.16702466434058313. The absolute error is $7.767708698480646 \times 10^{-11}$. Eight significant figures still agree. The loss is already visible in the next digit, and the largest term is already a million times the function.

At argument 30 the six-decimal display the old page used is already the wrong display. The stopped sum -0.08638030775692472 prints as -0.086380 . The function -0.08636798358104028 prints as -0.086368 . The absolute error is $1.2324175884440014 \times 10^{-5}$. A reader comparing two calculators to six decimals would conclude that they disagree, and the series calculator would be the one that is wrong. The page had no way to say so. The sum was finite, the last term was small, and the overflow branch did not run.

At argument 40 the disagreement is no longer a last-digit argument. The stopped sum is negative and the function is positive. Their ratio of absolute values is 12.23321720702749. Sixty-eight terms were enough for the stopping test. The sum never overflowed. The previous page would have printed -0.090121 . The same pattern at order 1 is less theatrical and just as wrong: the stopped sum 0.01028773273447872 against $J_1(40) = 0.12603831803758503$.

At argument 50 the stopped sum leaves the range of the function entirely. Section 4 records why $|J_n(x)|$ cannot exceed 1. The number 655.2869784264319 exceeds it by a factor of 11740.900367204033. Eighty-one terms were summed. The result was finite. The overflow message, which was the page’s only alarm, did not fire. Order 20 at the same argument is the same kind of failure: a stopped sum of 86.214 for a function whose value is -0.1167043527595798 .

The last row of Table 1 is the other face of an absolute tolerance. For $J_{50}(10)$ the first term is $(5)^{50}/50! = 2.9202857169191873 \times 10^{-30}$. The next term is smaller than 10^{-14} , so the old loop adds it and stops. Two terms give $1.4887731105862523 \times 10^{-30}$. The series, carried until the terms are small relative to the largest term rather than small relative to 1, is $1.7845136078715964 \times 10^{-30}$. The relative error is -0.16572610933355453 . The old page would have printed 0.000000 for both, because it used a fixed-point format for every finite sum below 10^6 . The stopping test and the display agreed to hide a relative error of one part in six.

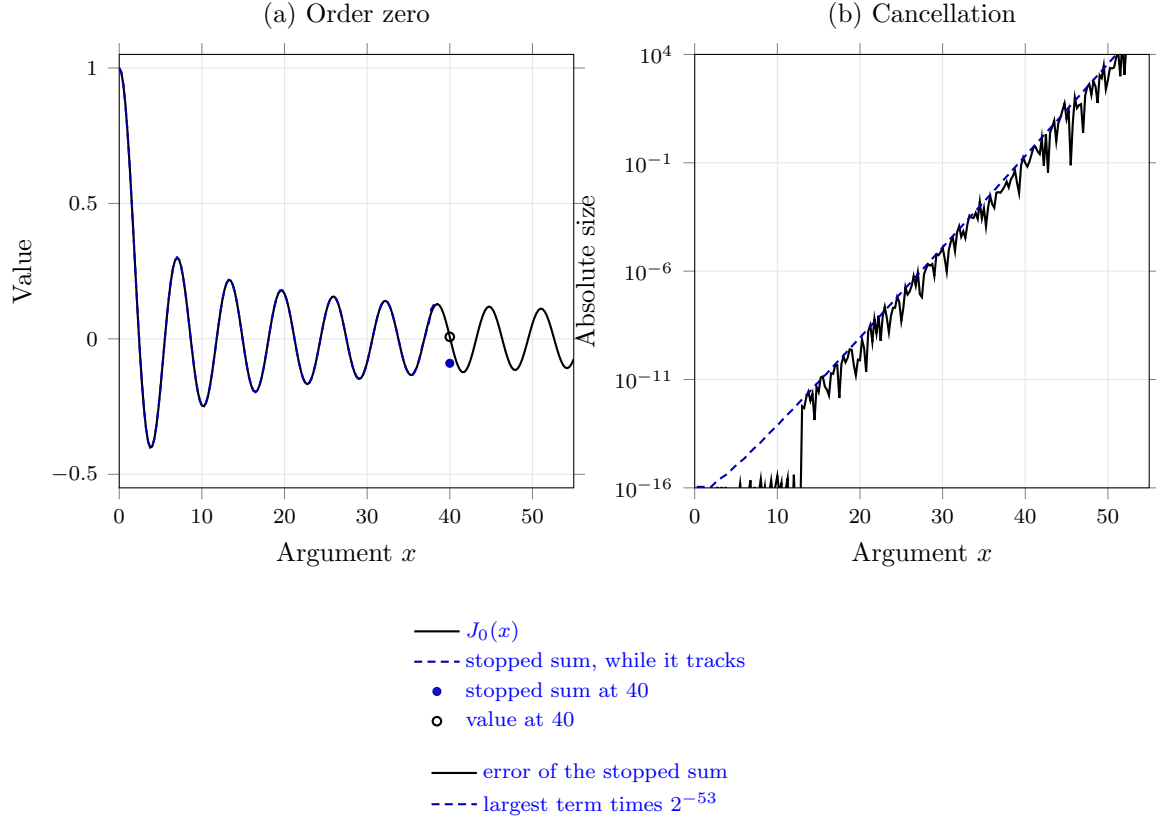


Figure 1: Order $n = 0$. Panel (a): the function, and the double-precision series that stops when a term falls below 10^{-14} . The dashed curve is omitted once it differs from J_0 by more than 0.02, which happens before argument 40. The filled mark is the stopped sum at 40, and the open mark is $J_0(40)$. Panel (b): the absolute error of that stopped sum, and the product of its largest term with 2^{-53} . Differences below 10^{-16} are drawn at 10^{-18} so the logarithm has a value. The error follows the cancellation scale once the terms exceed the function.

Figure 1(a) draws $J_0(x)$ and the stopped sum on the same axes. The stopped sum is drawn only while it remains within 0.02 of the function. It tracks J_0 through the first several lobes and leaves the curve between 38 and 39. The mark at argument 40 is the stopped sum -0.09012077265738101 , plotted against the function 0.007366890584237339 . Past that argument the stopped sum does not return to the curve and stay there. By argument 42 its absolute value exceeds 1. Figure 1(b) is the absolute error of the stopped sum together with the product of the largest term and 2^{-53} . The two curves rise together once the largest term exceeds the scale of the function. The error is not a mysterious property of Bessel functions. It is the size of the number the sum was asked to cancel.

4 A bound the sum violated

For every integer n and every real x ,

$$J_n(x) = \frac{1}{\pi} \int_0^\pi \cos(n\theta - x \sin \theta) d\theta. \quad (4)$$

This is Bessel’s integral, recorded as (10.9.2) in the DLMF and as equation 9.1.21 in Abramowitz and Stegun [2, 4]. The integrand is at most 1 in absolute value and the interval has length π , so

$$|J_n(x)| \leq 1. \quad (5)$$

The inequality is elementary and it is sharp at the origin for $n = 0$. It does not require a library, a table, or a second algorithm. Any procedure that returns 655 for a real argument and an integer order has not returned $J_n(x)$, whatever its last term looked like.

The integral also explains the oscillation in Figure 1(a). The phase $n\theta - x \sin \theta$ is stationary, and the value is a cancellation inside an integral of length π , not a quantity that grows with x . The power series represents the same function by a cancellation among terms that do grow with x . Those are different cancellations. The integral’s cancellation is the function. The series’ cancellation is an artifact of the representation, and double precision does not have the digits to perform it once the largest term exceeds about 10^{16} .

One further identity fixes the scale of the recurrence in the next section. The Jacobi–Anger expansion at angle zero reduces to

$$1 = J_0(x) + 2 \sum_{k=1}^{\infty} J_{2k}(x), \quad (6)$$

which is (10.12.4) in the DLMF and equation 9.1.46 in Abramowitz and Stegun [2, 4]. It is an identity for the functions, not a normalization one is free to choose. A set of trial values that satisfies the recurrence can be divided by the left-hand side of (6) computed from those trial values, and the quotients are then on the scale of J_n .

5 Backward recurrence

The functions of integer order satisfy the three-term recurrence

$$J_{n-1}(x) + J_{n+1}(x) = \frac{2n}{x} J_n(x), \quad (7)$$

for $x \neq 0$ [2, eq. 9.1.27]. The DLMF states it for cylinder functions as (10.6.1) [4]. Rearranged, (7) computes a lower order from two higher ones,

$$J_{n-1}(x) = \frac{2n}{x} J_n(x) - J_{n+1}(x). \quad (8)$$

The same relation run toward higher orders is unstable for J_n once the order exceeds the argument. $J_n(x)$ decreases rapidly in n for $n \gg |x|$, while the second solution of the difference equation grows. A rounding error introduced in a forward step has a component along that growing solution, and a few steps later the component is the answer [6, 7]. Running (8) downward does the opposite. The wanted solution is the one that was small at high order, and the unwanted solution is the one that shrinks as the order falls.

Miller’s algorithm is this downward sweep, followed by a normalization that does not require a known value of J_0 [8]. Olver’s error analysis gives strict bounds for the truncation and for the rounding committed at each step, under the assumption that the sweep starts where the wanted solution is already decreasing and the unwanted solution is already increasing [9]. For Bessel functions that regime begins past the turning point $n \approx |x|$, and the width of the turning region scales as $|x|^{1/3}$. Gautschi’s Algorithm 236 is the same idea in Algol, for the first kind [10]. The

survey of three-term recurrence that followed is the reason a minimal solution is computed backward rather than forward [7].

The calculator uses a concrete start, not an adaptive search. For $x \neq 0$ it sets

$$N = \max(n, \lceil |x| \rceil) + 32 + \left\lceil 10 |x|^{1/3} \right\rceil \quad (9)$$

and applies (8) from order N down to order 0, with trial values $y_N = 1$ and $y_{N+1} = 0$. Whenever a trial value exceeds 10^{150} in absolute value, the stored tail is scaled by 10^{-150} . The scale factor is then (6) summed over the even orders that were computed. The value at order n is divided by that factor, and multiplied by $(-1)^n$ when x is negative, because $J_n(-x) = (-1)^n J_n(x)$ for integer n . For argument 40 and order 0, (9) is $40 + 32 + 35 = 107$.

The three-term relation (7) is the wrong residual to print. The trial values satisfy it by construction, up to roundoff, even when N is too small and the answer is wrong. The identity (6) is also the wrong residual: it is enforced when the trial values are scaled, so the truncated sum equals 1 whether or not the omitted tail was negligible. The check that can fail is a second sweep from a higher order. The page runs (9) and the same formula with 24 extra orders, and reports the difference of the two results. At $J_0(40)$ that difference is $1.8127860323957634 \times 10^{-16}$. If the absolute difference exceeds 10^{-10} , the page does not print a value.

The start in (9) is not decorative. The same downward recurrence for $J_0(40)$, begun only five orders past the argument, at order 45, returns 0.008457012649243444. The function is 0.007366890584237339. The recurrence relation is satisfied and the normalizing sum equals 1, and the number is still wrong in the third significant figure. Olver's hypothesis, that the sweep must begin where the decreasing character has set in, is visible in that gap of 62 orders between a bad start and the start (9) uses.

The series is not abandoned for every argument. It is returned when every term stayed finite and the largest absolute term is at most 10^4 . On that range a term-by-term cancellation cannot move the result by more than about $10^4 \times 2^{-53}$, which is a few times 10^{-12} . The stopping test inside this retained series is relative to the largest term, and it requires the terms to have started decreasing. An absolute floor of 10^{-14} is what cut $J_{50}(10)$ off after two terms; that floor is not used. On the grid of even orders from 0 to 50 and arguments $0, 0.5, \dots, 40$, the value returned by this rule differs from the 60-digit series by less than 5×10^{-12} , and every returned absolute value is at most 1 up to a roundoff of 10^{-8} . Past the peak limit, or when a term overflows, the recurrence is used for $|x| \leq 20000$. A larger absolute argument is refused. It is not replaced by the last finite term, and it is not replaced by the endpoint of a loop.

A result whose absolute value exceeds $1 + 10^{-8}$ is also refused. The bound (5) is cheap to apply after either method. It would have rejected the stopped sum at argument 50 without any reference implementation. It would not have rejected the stopped sum at argument 40, which is inside $(-1, 1)$ and still has the wrong sign. The bound is a filter, not a certificate. The two-start difference is the certificate the page prints.

6 Benchmarks

The default case is the one the form opens on. Order 0, argument 1, series, eleven terms, largest term 1, last term $7.242258480192778 \times 10^{-20}$, value (3). The previous page agreed here, as Table 1 shows. A calculator that is only checked at $J_0(1)$ will keep a series that fails at $J_0(40)$.

The sign error is order 0, argument 40. The replacement starts at order 107 and returns 0.007366890584237339. The difference from a start 24 orders higher is $1.8127860323957634 \times 10^{-16}$. The 60-digit series agrees to 5×10^{-17} . The stopped sum is -0.09012077265738101 . The short

recurrence from order 45 is 0.008457012649243444, which is neither of those numbers. Three algorithms, three answers, and only one of them matches the series it claims to evaluate.

The bound violation is order 0, argument 50. The replacement starts at order 119 and returns 0.05581232766925194, with a two-start difference of $4.163336342344337 \times 10^{-17}$. The stopped sum is 655.2869784264319 after 81 terms, and the largest term was $3.278755830813472 \times 10^{19}$. The same argument at order 1 is in the table. Nothing about these cases depends on a special choice of γ or a unit conversion. The inputs are the order and the argument.

The hidden relative error is order 50, argument 10. The largest term is 2.92×10^{-30} , which is below 10^4 , so the replacement uses the series. It uses 17 terms and returns $1.7845136078715964 \times 10^{-30}$. The absolute stopping test used two terms. The new display prints this magnitude in scientific notation rather than as 0.000000. The value is small because n is large compared with x , which is the right asymptotic regime of (1), not because the calculation underflowed.

Negative arguments use the parity relation rather than a second series. $J_0(-10) = J_0(10)$ and $J_1(-3) = -J_1(3)$, and both are series evaluations because the largest term is modest. A negative order is rejected. For an integer it would be $(-1)^n J_n(x)$, and the page says that instead of rounding the sign away. A non-integer order is rounded to the nearest integer, and the form is rewritten with that integer. The previous handler applied `parseInt` before rounding, so an order of 1.6 was evaluated as order 1.

7 Discussion

The numerical claim is narrower than “backward recurrence is the right way to compute Bessel functions.” The series (1) is the right way while its terms fit in the format with room to spare, and it is the definition one should teach. The defect was a stopping test that inspected only the last term. That test coincides with convergence in exact arithmetic, where a small last term and a small tail are the same statement. In double precision the tail can be small while the sum of the earlier terms has already lost every digit that the tail was supposed to correct. Table 1 is the concrete form of that distinction. At argument 1 the last term and the error are both negligible. At argument 40 the last term is negligible and the error is the whole value, including its sign.

Printing the two-start difference is the corresponding distinction for the recurrence. A single downward sweep always produces a number. The number from order 45 and the number from order 107 are both outputs of Miller’s algorithm, and they differ by 1.09×10^{-3} . The page’s residual is that difference, not a rearrangement of (7) that the sweep satisfies automatically. Olver already noted that comparing two starting orders is the practical check, and that it is not a substitute for an a priori bound [9]. The calculator uses the comparison as a refusal condition and uses (9) so that, on the cases in Table 1, the comparison comes back at a unit roundoff rather than as the thing that saves a bad start.

The peak limit 10^4 is a deliberate early exit from the series, not an estimate of the radius of convergence. There is no radius. Leaving the series in place until the largest term is 10^8 lets a cancellation of about $10^8 \times 2^{-53}$ move a later digit of a small value. Cutting the series off at 10^4 keeps that contribution near 10^{-12} and hands the rest of the axis to a method whose error does not grow with the intermediate terms. $J_0(20)$ is on the recurrence side of the cut even though the stopped sum there is still correct to eight figures. The page gives up the series before the displayed digits go, not after.

None of this extends the mathematical domain of the page. Fractional order, complex argument, Y_n , and the modified functions I_n and K_n are still other functions. The recurrence for I_n is Miller’s original example, with a different normalizing identity [8, 9]. The page does not compute it. The

bound (5) is special to J_n on the real line. A user who needs a value outside $0 \leq n \leq 50$ and $|x| \leq 20000$ is told that the value is outside the range, which is the sentence the overflow message was attempting and did not manage to say when the sum was a finite 655.

8 Limitations

The implementation is $J_n(x)$ for an integer n from 0 to 50 and a real x with absolute value at most 20000. The two-start residual certifies agreement between two runs of (8) at the starts (9) and (9) plus 24. It does not certify a physical model that happens to contain a Bessel function. A radial wave number, a membrane tension, and a boundary condition are the user's problem. The calculator receives the dimensionless argument those choices have already produced.

The additive term $32 + 10|x|^{1/3}$ in (9) is a fixed multiple of the turning width, checked by the 60-digit series on the even-order grid above and by the two-start difference. It is not claimed to be the smallest safe start at every argument. The order-45 calculation shows that some smaller starts are not safe. An argument within a rounding error of a zero can return a value at the level of the residual, which for the series near 2.404825557695773 is about 10^{-16} and is the right size for a double-precision root. It should not be read as a 16-digit determination of a higher zero.

Orders above 50 are refused even though the recurrence would accept them. The cap is the range the page offers, not a stability limit of (8). Decimal orders are rounded to the nearest integer and the rounded value is written back into the form, so the number that was evaluated is the number that remains visible. Negative orders are not silently folded. The identity that would fold them is stated in the error message.

The figure compares the stopped sum with the replacement. It is not a plot of every algorithm that has been used for J_n . Hankel's expansion, continued fractions, and uniform asymptotic expansions in the turning region are the methods a production library combines [2, 4]. They are unnecessary on $|x| \leq 20000$ at integer order up to 50, where (9) is a few hundred steps, and they are not what this page runs.

9 Conclusion

A Bessel value is auditable when the method and a check that can fail are part of the result. The series (1) converges for every finite argument, and its double-precision partial sums do not, once the largest term has cancelled away the digits of the answer. Stopping at a term below 10^{-14} reports $J_0(40) = -0.09012077265738101$ for a function equal to 0.007366890584237339, and $J_0(50) = 655.2869784264319$ for a function equal to 0.05581232766925194. The second of those numbers is incompatible with $|J_n(x)| \leq 1$. The same absolute test, at order 50 and argument 10, stops after two terms and is low by 16.6 percent. The replacement uses the series when the largest term is at most 10^4 , including the default $J_0(1) = 0.7651976865579666$ and the first positive zero. Otherwise it recurs backward from the order (9), normalizes by (6), and reports the change from a start 24 orders higher. At argument 40 that change is 1.81×10^{-16} . The series had already converged. It had converged to the wrong answer.

Data and code availability

The calculator is the AgentCalc page [bessel-function-calculator](#), and the released copy is in the [agentcalc.science](#) repository. The benchmark script `generate_bessel_figure.py` and the series plotted in Figure 1 are archived with this note. No laboratory measurement was analyzed.

References

- [1] G. N. Watson. *A Treatise on the Theory of Bessel Functions*. Cambridge University Press, Cambridge, 2 edition, 1944.
- [2] Milton Abramowitz and Irene A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Number 55 in National Bureau of Standards Applied Mathematics Series. U.S. Government Printing Office, Washington, DC, 1964.
- [3] Friedrich Wilhelm Bessel. Untersuchung des theils der planetarischen störungen, welcher aus der bewegung der sonne entsteht. *Abhandlungen der Königlichen Akademie der Wissenschaften zu Berlin, Mathematische Klasse*, pages 1–52, 1824. doi: 10.3931/e-rara-74729. Presented 29 January 1824. Digitized by the Berlin-Brandenburg Academy of Sciences and Humanities, urn:nbn:de:kobv:b4360-1006612.
- [4] F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain. NIST digital library of mathematical functions. dlmf.nist.gov/10, 2024. Chapter 10, Bessel Functions, by F. W. J. Olver and L. C. Maximon.
- [5] Henry Gerber. First one hundred zeros of $J_0(x)$ accurate to 19 significant figures. *Mathematics of Computation*, 18:319–322, 1964. doi: 10.1090/S0025-5718-1964-0164065-3.
- [6] M. Goldstein and R. M. Thaler. Recurrence techniques for the calculation of Bessel functions. *Mathematical Tables and Other Aids to Computation*, 13:102–108, 1959. doi: 10.1090/S0025-5718-1959-0105794-5.
- [7] Walter Gautschi. Computational aspects of three-term recurrence relations. *SIAM Review*, 9(1):24–82, 1967. doi: 10.1137/1009002.
- [8] British Association for the Advancement of Science. *Bessel Functions, Part II: Functions of Positive Integer Order*, volume X of *Mathematical Tables*. Cambridge University Press, Cambridge, 1952. The backward recurrence is set out in the introduction, p. xvii.
- [9] F. W. J. Olver. Error analysis of Miller’s recurrence algorithm. *Mathematics of Computation*, 18(85):65–74, 1964. doi: 10.1090/S0025-5718-1964-0169406-9.
- [10] Walter Gautschi. Algorithm 236: Bessel functions of the first kind. *Communications of the ACM*, 7(8):479–480, 1964.