Consistency tests for elementary functions

by A. C. R. NEWBERY and ANNE P. LEIGH

University of Kentucky
Lexington, Kentucky

INTRODUCTION

The possibility of using consistency tests to determine the quality of an elementary function subroutine has been considered by several authors. Although none of the authors thought highly of the idea, our investigations have led us to conclude that consistency tests do have a definite provable value in some situations. The error in a subroutine has two possible sources: (a) range-reduction and (b) the reduced-range approximation. For instance, in approximating the sine of a large angle one reduces the problem to that of approximating the sine or cosine of an angle in a reduced-range—perhaps \([0, \pi/4]\). Since the range-reduction process will then involve subtracting a large integer-multiple of an inaccurately represented \(\pi/2\), it is clear that the reduced-range argument will be in error by a quantity which varies linearly with the original argument. Since these range-reduction errors are unavoidable and well understood, we have concentrated our efforts on consistency tests which will help to evaluate the quality of a subroutine in the reduced-range approximation. We give three examples. In each case the variable \(x\) is supposed to be within the reduced range; the tests are still valid without this condition, but there is a diminished likelihood of our bounds being realistic when \(x\) is outside the reduced range.

SINE AND COSINE TEST

We can construct a consistency test for the sine and cosine routine for \(0 \leq x < \pi/4\) based on the identity \(\cos^2 x + \sin^2 x = 1\). Let \(c, s\) denote \(\cos x, \sin x\); let \(M(c), M(s)\) denote machine values for \(c, s\), respectively, and let \(c', s'\) be the “admitted errors” in \(c, s\), i.e., the manufacturer is prepared to admit that \(|M(c) - c|\) could be as big as \(c'\) but no bigger. If values for \(c', s'\) are not available to us we have a good cause for complaint, although our test can still yield meaningful results using assumed values for \(c', s'\). First we compute the residual \(r\) defined by \((M(c))^2 + (M(s))^2 - 1 = r\), and we assume that \(M(c)\) and \(M(s)\) are each wrong by a fraction \(t\) of the admitted error, and that the signs reinforced the error to build it up to the observed residual \(r\). Our assumption then is \((c + tc')^2 + (s + ts')^2 - 1 = r\), or

\[
(c^2 + s^2) + 2t(cc' + ss') - r = 0. \tag{1}
\]

If \(t_m\) is the root of (1) of smallest magnitude we can assert that either \(M(c)\) or \(M(s)\) is wrong by a fraction at least \(|t_m|\) of the admitted error. This is a rigorous assertion, because throughout we have given the manufacturer the benefit of the doubt by assuming that the observed residual \(r\) was the result of small errors reinforcing rather than large errors cancelling. The only trouble is that we cannot solve the quadratic (1) because we do not have trustworthy values for \(c, s\). To get around the difficulty we define a neighboring quadratic equation

\[
z^2(c'^2 + s'^2) + 2z(cc'M(c) + ss'M(s)) - r = 0, \tag{2}
\]

and let its root of smaller magnitude be \(z_m\); we attempt to relate \(t_m, z_m\). If we differentiate (1) with respect to the parameter \(c\) we obtain

\[
2\left(\frac{dt}{dc}\right)(c'^2 + s'^2) + 2\left(\frac{dt}{dc}\right)(cc' + ss') + 2tc' = 0,
\]

hence

\[
\frac{dt}{dc} = -\frac{tc'}{(t(c'^2 + s'^2) + cc' + ss')} \tag{3}
\]

Hence \(dt/dc\) has the sign—\(t\) for the chosen argument range. From (1) we see that if \(r > 0\) then \(t_m > 0\). Hence from (3) \(\left(\frac{dt}{dc}\right)_{t=t_m} < 0\), i.e., the positive root of (1) decreases if we increase the value of \(c\). But increasing the value of \(c\) is precisely what we are doing when we replace \(c\) by \(M(c)\) on moving from (1) to (2), because positive \(r\) implied \(M(c) > c\). Similarly, the root will also
diminish on replacement of $s$ by $M(s)$. We conclude that for $r > 0$ the root $z_m$ of (2) is a closer lower bound for the quantity $t_m$. Unfortunately the case $r < 0$ is different. In this case $t_m < 0$ and from (3) $(dt/dc)_{t=t_m} > 0$. The replacement of $c$ by $M(c)$ is like a decrease in the parameter $c$, implying an algebraic decrease in the (negative) value $t$, hence an increase in the magnitude of the smaller root. This gives us a bound on the wrong side of $t_m$. To get a bound on the right side we must replace $c$ by a computable approximation that is guaranteed $\approx c$. Such a quantity is $M(c) + c'$; similarly we replace $s$ by $M(s) + s'$. Hence, when $r < 0$ we have, in place of (2)

$$s^2(c'^2 + s'^2) + 2s(c'(M(c) + c') + s'(M(s) + s')) - r = 0.$$  

$(2')$

The quantity $|z_m|$, computed by (2) for $r > 0$ and by (2') for $r < 0$, is a lower bound for $|t_m|$ and we can state that at least one of the quantities, $c$, $s$ is wrong by at least $|z_m c'|$, $|z_m s'|$ respectively. If no values are given for $c'$, $s'$ we can choose them at will. A particularly appropriate choice in this situation is $c' = c$, $s' = s$. Although these might seem unreasonably large 'errors,' this will be offset by the smallness of $|t_m|$. On substitution into (1) we shall obtain

$$p^2 + 2t - r = 0,$$

and hence

$$t_m = -1 + (1 + r)^{1/2} = \frac{r - \sqrt{r^2}}{2}, \quad |r| < 1.$$  

$(5)$

Ordinarily one is only interested in obtaining one significant figure of $t_m$, and it would be a grossly incorrect subroutine that yielded $r$-values so large that the above two terms of the binomial series did not suffice. Having obtained $t_m$, we can be sure that at least one of the following inequalities holds:

$$|M(c) - c| \geq |t_m| c, \quad |M(s) - s| \geq |t_m| s.$$  

$(6)$

If one prefers to have the error bound measured against the observables $M(c)$, $M(s)$ rather than against the unknowns, $c$, $s$, one can, for $z, M(z) > 0$ and $0 < T < 1$, use the following fact based on the manipulation of inequalities:

If

$$|M(z) - z| \geq Tz,$$

then

$$|M(z) - z| \geq TM(z)/(1 + T).$$  

$(7)$

By substituting $|t_m|$ for $T$ and either $c$ or $s$ for $z$, the error bounds (6) can then be written in terms of observables $M(c)$, $M(s)$. This will generally involve a slight weakening of the inequalities.

### Exponential Test

A consistency test for an exponential routine can be made by selecting number pairs $x, h$, and defining

$$\frac{M(e^x)M(e^h)}{M(e^{x+h})} - 1 = r.$$  

$(8)$

If the admitted error magnitudes for $e^{x+h}$, $e^x$, $e^h$ are $a', b', c'$ respectively then the counterpart of (1) for this problem is

$$a'b'c' + (b'c' + c'e' + a'(1+r)) - r = 0.$$  

$(9)$

In order to obtain quadratics analogous to (2), (2') we modify (9) by substituting machine values for the exponentials when $r > 0$, and when $r < 0$ we replace them by machine values plus admitted errors. The analogs of (4), (5) are

$$p^2 + (3 + r)t - r = 0, \quad t_m = (3 + r)(\rho - \rho^2), \quad \rho = r/(3 + r)^2.$$  

$(10)$

### Hyperbolic Functions

As a final example we consider a slightly more complicated identity and we simplify the problem by ignoring the admitted errors and going straight to the determination of a relative error bound. For non-negative number pairs $x, h$ we use the identity

$$\sinh(x + h) + \sinh(x - h) = 2 \cosh h \sinh x.$$  

As we assume relative errors of magnitude $t$ and with signs chosen so that there is a reinforcing effect on the composite error, we obtain

$$\frac{M(\sinh(x + h)) + M(\sinh(x - h))}{2M(\cosh h)M(\sinh x)} - 1 = \frac{1}{(1 + t)^2} - 1.$$  

$(11)$

This leads to

$$\rho - \frac{t(3 + 2r)}{(1 + r)^2} + \frac{r}{(1 + r)^2} = 0$$  

$(12)$

and

$$t_m = \left(\frac{(3 + 2r)}{(1 + r)^2}\right)(\rho + \rho^2), \quad \rho = \frac{r(1 + r)}{(3 + 2r)^2}.$$  

### Testing the Tests

In order to see whether our tests would indeed give useful lower bounds in a practical situation, we performed several test runs on an IBM 360/65 using...
standard single-precision software for the test functions and double precision for the computation and processing of residuals. A description of the tests follows:

The consistency test for the sine and cosine subprograms was run over a range of values \((0, 2^{-10}, \pi/4)\) for the argument \(x\). The residual \(r\) was evaluated from the equation

\[
(M(\cos x))^2 + (M(\sin x))^2 - 1 = r
\]

for \(r > 0\), equation (2) was solved for the root of smaller magnitude \(t_m\). For \(r < 0\), equation (2') was used. The admitted absolute error bounds for the cosine and sine subprograms were \(1.47 \times 10^{-7}\) and \(1.31 \times 10^{-7}\) respectively. The maximum value for \(|t_m|\) was found to be 0.512 at \(x = 0.700195\). Therefore, at least one of \(\cos x\) or \(\sin x\) is wrong by at least \(7.52 \times 10^{-7}\), and \(6.70 \times 10^{-7}\) respectively. The first bad binary digit provable by this technique occurs in the 23rd bit position.

The consistency test for the exponential subprogram was run using Equation (8) to evaluate the residual \(r\). Assuming relative errors of magnitude \(t\), and with signs chosen to have a reinforcing effect on the composite error, we obtain

\[
M(e^x)M(e^h) - 1 = r = \frac{(1+t)^2}{1-t} - 1 \quad \text{for } r > 0
\]

and

\[
r = \frac{(1-t)^2}{1+t} - 1 \quad \text{for } r < 0.
\]

These lead to

\[
e^t + t(r + 3) - r = 0 \quad \text{for } r > 0
\]

and

\[
e^t - t(r + 3) - r = 0 \quad \text{for } r < 0,
\]

and the root of smaller magnitude in either case is given by

\[
|t_m| \approx (3 + r)(\rho - r^2), \quad \rho = r/(3 + r)^3.
\]

The test was run over a range of values \((h, 2^{-10}, 0.5)\) for \(x\) and values of \(2^{-10}, 2^{-1},\) and \(2^{-4}\) for \(h\). The table below shows the results obtained:

| \(h\)  | \(\text{Maximum } |t_m|\) | \(z\) |
|-------|------------------|-----|
| \(2^{-4}\) | 0.159 \times 10^{-6} | .09180 |
| \(2^{-5}\) | 0.212 \times 10^{-6} | .0625 |
| \(2^{-4}\) | 0.370 \times 10^{-6} | .0625 |

The admitted relative errors are \(1.2 \times 10^{-6}\) and \(1.31 \times 10^{-6}\) for sinh and cosh respectively.

The above test was rerun using the expression \(.5(e^x - e^{-x})\) with the library exponential routine to synthesize \(\sinh x\); hence low accuracy can be expected for small \(x\)-values owing to the subtraction of nearly equal quantities. The following table shows the results obtained:

| \(h\)  | \(\text{Maximum } |t_m|\) | \(z\) |
|-------|------------------|-----|
| \(2^{-4}\) | 4.42 \times 10^{-4} | .02246 |
| \(2^{-5}\) | 2.36 \times 10^{-4} | .03223 |
| \(2^{-4}\) | 2.10 \times 10^{-4} | .08105 |

The errors are seen to be much larger than in the previous run. We can conclude that this consistency test is thus able to detect a 'bad' sinh subprogram.

CONCLUSION

Although we had (and still have) no reason to doubt the manufacturer's word concerning the accuracy of the subroutines supplied, we note that our tests were able to prove errors of comparable magnitude to those that the manufacturer admitted. In one case the provable error was 89 percent of the admitted error. In view of this we feel it is safe to conclude that any manufacturer whose claims are at all extravagant can almost certainly be proved wrong on the basis of carefully constructed consistency tests alone. It is also worth noting that our tests were able to show up a badly coded hyperbolic function routine, since badly coded routines of this sort are said to be in circulation.

In summary we would like to state our position on consistency tests: Firstly we do not advocate them as a substitute for an elaborate full-scale validation process;
we merely note that there are many routines in circulation that have evidently not passed any stringent tests at all. Since few institutions possess the funds and manpower to do their own full-scale validation, there is a legitimate market for consistency tests which can quickly and cheaply give a rough indication of quality. Secondly we believe that in the literature, particularly, there has been a tendency to underrate the potentialities of the consistency test. Too much importance is sometimes attached to the fact that a subroutine can be grossly in error while exactly satisfying a mathematical identity. One can guard against being misled by this situation, partly by use of redundancy (using several different identities) and partly by common sense (i.e., by avoiding identities which the subroutine programmer is likely to have used in the given range). Cody\(^2\) gives additional advice on the choice of identity. Thirdly, although we believe that consistency tests deserve to be held in higher esteem than is commonly the case, we are anxious not to over-correct the situation. A consistency test can only provide lower bounds for errors. Such a test cannot possibly tell “the whole truth” about a subroutine; on the other hand, it will tell “nothing but the truth,” and we think we have demonstrated that just this much information, delivered promptly and cheaply, can be very helpful.

REFERENCES

1 C HAMMER
Statistical validation of mathematical computer routines
Proc SJCC 1967 331–333
2 W J CODY
Performance testing of function subroutines
Proc SJCC 1969 759–763
3 J F HART et al.
Computer approximations
Wiley New York 1968
4 IBM Publication form C 28-6596-4

From the collection of the Computer History Museum (www.computerhistory.org)