JOURNAL OF COMPUTATIONAL PHYSICS 32, 270–279 (1979)

Note

An Algorithm with ALGOL 60 Program for the Computation of the Zeros of Ordinary Bessel Functions and those of their Derivatives

1. Introduction

In this note we describe an algorithm for the numerical computation of the zeros of the Bessel functions

$$J_a(x),\ Y_a(x),\ J_a'(x),\ Y_a'(x). \tag{1.1}$$

When a is real, these functions each have an infinite number of real zeros, all of which are simple with the possible exception of $x = 0$. For non-negative a the sth positive zeros of these functions are denoted by

$$j_{a,s},\ y_{a,s},\ j_{a,s}',\ y_{a,s}' \quad (s = 1, 2, 3,\ldots) \tag{1.2}$$

except that $x = 0$ is counted as the first zero of $J_0'(x)$. For properties of the zeros the reader is referred to the literature, for instance to Olver [3], or to his contribution in Abramowitz and Stegun [1, Chap. 9].

In Section 2 we describe the algorithm for computing the zeros. It is based on a higher order Newton process. In Section 3 we give methods for obtaining first approximations of the zeros. We use asymptotic expansions as given in the literature. In Section 4 we give the implementation of the algorithm as an ALGOL 60 procedure. In order to compute values of the functions of (1.1) we use the ALGOL 60 procedures published earlier in Temme [4], which procedures are not incorporated here.

2. Higher Order Newton Process

It is well-known (cf., for instance, Hofsommer [2]) that, if a function satisfies a second order differential equation, this fact may be used with advantage in the computation of its zeros. In that event it is convenient to use a process in which derivatives of the function are needed. Such a process is the Newton–Raphson method. We use a higher order version of it.

Let f be the function, the roots of which are to be computed. Let α be such a (simple!) root and let x be a first approximation. If the approximation is sufficiently accurate we have

$$\alpha = x - \rho(x)[1 + \alpha_1(x)\rho(x) + \alpha_2(x)\,\rho^2(x) + \mathcal{O}(\rho^3(x))] \tag{2.1}$$

0021-9991/79/080270-10$02.00/0

with

$$\begin{aligned} \rho(x) &= f(x)/f'(x) \\ \alpha_1(x) &= \tfrac{1}{2}f''(x)/f'(x) \\ \alpha_2(x) &= \tfrac{1}{6}\{3[f''(x)/f'(x)]^2 - f'''(x)/f'(x)\}. \end{aligned} \tag{2.2}$$

By using the differential equation of the Bessel functions

$$x^2C_a''(x) + xC_a'(x) + (x^2 - a^2)\, C_a(x) = 0 \tag{2.3}$$

the derivatives f'' and f''' can be eliminated. Furthermore we remark that derivatives of the Bessel functions can be expressed in terms of other Bessel functions by using

$$C_a'(x) = \frac{a}{x}\, C_a(x) - C_{a+1}(x). \tag{2.4}$$

In (2.3) and (2.4) C_a stands for J_a or Y_a.

The expression (2.1) will be modified in several ways. A first modification comes from replacing $\rho(x) = C_a(x)/C_a'(x)$ by $r(x) = C_a(x)/C_{a+1}(x)$ (for the case of J_a and Y_a) and from replacing $\rho(x) = C_a'(x)/C_a''(x)$ by $r(x) = C_a'(x)/C_a(x)$ (for J_a' and Y_a'). In both cases $\rho(x)/r(x) = \mathcal{O}(1)$ for $x \to \alpha$, hence the $\mathcal{O}$-term remains of the same order, if we change α_1 and α_2 in the right way.

A second modification comes from writing the quadratic polynomial in (2.1) as a Padé-fraction. An expression of the type

$$f(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2 + \mathcal{O}(x^3), \qquad x \to 0, \tag{2.5}$$

may be converted by Padé-methods into

$$f(x) = \frac{p_0 + p_1 x}{1 + q_1 x} + \mathcal{O}(x^3), \qquad x \to 0, \tag{2.6}$$

where, if $\alpha_1 \neq 0$,

$$p_0 = \alpha_0\,, p_1 = (\alpha_1{}^2 - \alpha_0\alpha_2)/\alpha_1\,, q_1 = -\alpha_2/\alpha_1\,. \tag{2.7}$$

In an asymptotic sense (2.5) and (2.6) are equivalent, but numerically the fraction in (2.6) may yield a better approximation to f than the polynomial part in (2.5). From numerical experiments we concluded that this is indeed the case for the representation in (2.1).

A third modification of (2.1) comes from eliminating the derivatives of the Bessel functions appearing in α_1 and α_2 of (2.2). In the following subsections we give more details for the computation of the coefficients p and q appearing in the form obtained after the modifications, viz.

$$\alpha = x - r(x)\,\frac{1 + pr(x)}{1 + qr(x)} + \mathcal{O}(r^4(x)), \qquad x \to \alpha. \tag{2.8}$$

2.1. *The Coefficients p and q for $J_a(x)$ and $Y_a(x)$*

First we remark that

$$\rho(x) = C_a(x)/C_a'(x) = -r(x)\Big/\Big[1 - \frac{a}{x}\, r(x)\Big],$$
$$r(x) = C_a(x)/C_{a+1}(x).$$

Further we have for α_1 and α_2 of (2.2) the relation

$$\alpha_1(x) = -\frac{1}{2x}\left[1 + \frac{a^2 - x^2}{x}\, r(x) + \mathcal{O}(r^2(x))\right], \qquad x \to \alpha,$$
$$\alpha_2(x) = \frac{1 + x^2 - a^2}{6x^2} + \mathcal{O}(r(x)), \qquad x \to \alpha.$$

This gives (2.8) (by using (2.5), (2.6), (2.7)), with

$$p = \frac{1 + 4x^2 - 4a^2}{6x(2a+1)}, \qquad q = \frac{2x^2 - 1 - 6a - 8a^2}{3x(2a+1)}. \tag{2.9}$$

Formula (2.8) is then used in the sense that if x is an approximation for the sth zero of $J_a(x)$ or $Y_a(x)$, then

$$j_{a,s}\,,\, y_{a,s} = x - r(x)\,\frac{1 + pr(x)}{1 + qr(x)} \tag{2.10}$$

is a better approximation, with p and q given in (2.9) and

$$r(x) = J_a(x)/J_{a+1}(x) \qquad \text{or} \qquad r(x) = Y_a(x)/Y_{a+1}(x).$$

2.2. *The Coefficients p and q for $J_a'(x)$ and $Y_a'(x)$*

In this case we write in (2.1)

$$\rho(x) = \frac{C_a'(x)}{C_a''(x)} = r(x)\,\frac{x^2}{a^2 - x^2 - xr(x)}, \qquad r(x) = C_a'(x)/C_a(x)$$

and we change α_1 and α_2 as before. The result is

$$j_{a,s}'\,,\, y_{a,s}' = x - \frac{x^2}{a^2 - x^2}\, r(x)\,\frac{1 + pr(x)}{1 + qr(x)}, \tag{2.11}$$

where x is an approximation of the sth zero of $J_a'(x)$ or $Y_a'(x)$,

$$r(x) = J_a'(x)/J_a(x) \qquad \text{or} \qquad r(x) = Y_a'(x)/Y_a(x)$$

and

$$p = \frac{4x(a^2 - x^2)}{6(a^2 + x^2)} \left[1 + \frac{10x^2a^2 + 3x^4 - a^4}{4(a^2 - x^2)^3}\right],$$

$$q = \frac{4x(a^2 - x^2)}{6(a^2 + x^2)} \left[1 + \frac{8x^2a^2 + 3x^4 + a^4}{2(a^2 - x^2)^3}\right].$$

Formulas (2.10) and (2.11) will be repeatedly used until the approximation is sufficiently accurate. In the next section we describe methods for obtaining first (i.e., initial) approximations to the zeros.

3. First Approximations to the Zeros

From Abramowitz and Stegun [1, p. 371] we take the following expansions of the sth zero $j_{a,s}$ (or $y_{a,s}$) of $J_a(x)$ (or $Y_a(x)$)

$$j_{a,s}\,,\, y_{a,s} = \beta - \frac{\mu - 1}{8\beta}\left[1 + \frac{4(7\mu - 31)}{3(8\beta)^2} + \frac{32(83\mu^2 - 982\mu + 3779)}{15(8\beta)^4}\right] + \mathcal{O}(\beta^{-7}), \tag{3.1}$$

for $s \to \infty$, where $\mu = 4a^2$ and

$$\begin{aligned} \beta &= (s + \tfrac{1}{2}a - \tfrac{1}{4})\,\pi \qquad \text{for} \quad j_{a,s} \\ \beta &= (s + \tfrac{1}{2}a - \tfrac{3}{4})\,\pi \qquad \text{for} \quad y_{a,s}\,. \end{aligned} \tag{3.2}$$

This approximation can be written as (following the method described by (2.5), (2.6), (2.7))

$$j_{a,s}\,,\, y_{a,s} = \beta - \frac{\mu - 1}{8\beta}\left[\frac{1 - p/(8\beta)^2}{1 - q/(8\beta)^2}\right] + \mathcal{O}(\beta^{-7}), \tag{3.3}$$

for $s \to \infty$, with β given in (3.2) and

$$p = \frac{4(253\mu^2 - 3722\mu + 17869)}{15(7\mu - 31)}, \qquad q = \frac{8(83\mu^2 - 982\mu + 3779)}{5(7\mu - 31)}.$$

The relations in (3.1) and (3.3) are valid for $s \to \infty$. However, for small values of a they give good approximations for small values of s, even for $s = 1$. For $a = 0$, $s = 1$, (3.3) gives $j_{0,1} = 2.4052...$ while in 7 significant digits it is 2.404825. Hence, the absolute error is 0.00043. (Formula (3.1) gives a result with absolute error 0.0016.) For large values of a (3.3) (and (3.1)) is useless for the smaller s-values. Other approximations will be given for this case. First we give the results for the zeros of $J'_a(x)$ and $Y'_a(x)$ for small a.

In this case we have for large s

$$j'_{a,s}\,,\ y'_{a,s} = \beta' - \frac{\mu+3}{8\beta'} - \frac{4(7\mu^2+82\mu-9)}{3(8\beta')^3} - \frac{32(83\mu^3+2075\mu^2-3039\mu+3537)}{15(8\beta')^5} + \mathcal{O}(\beta'^{-7}), \tag{3.4}$$

where $\mu = 4a^2$ and

$$\begin{aligned} \beta' &= (s + \tfrac{1}{2}a - \tfrac{3}{4})\,\pi \qquad \text{for } j'_{a,s} \\ \beta' &= (s + \tfrac{1}{2}a - \tfrac{1}{4})\,\pi \qquad \text{for } y'_{a,s}\,. \end{aligned} \tag{3.5}$$

The Padé-version of (3.4) reads as follows:

$$j'_{a,s}\,,\ y'_{a,s} = \beta' - \frac{1}{8\beta'}\left[\frac{p_0 - p_1/(8\beta')^2}{1 - q_1/(8\beta')^2}\right] + \mathcal{O}(\beta'^{-7}), \tag{3.6}$$

where

$$\begin{aligned} p_0 &= \mu + 3, \\ p_1 &= \frac{4(253\mu^4 + 8204\mu^3 - 13874\mu^2 - 26100\mu + 63261)}{15(7\mu^2 + 82\mu - 9)} \\ q_1 &= \frac{8(83\mu^3 + 2075\mu^2 - 3039\mu + 3537)}{5(7\mu^2 + 82\mu - 9)}\,. \end{aligned}$$

As mentioned earlier, the approximations (3.3) and (3.6) are valid for small values of a. If a is large (this "large" will be specified in more detail further on) we need other approximations for the early zeros. These can be obtained from Olver's result on uniform asymptotic expansions for the Bessel functions. We use the formulas in Abramowitz and Stegun [1, p. 371, 9.5.22–9.5.26]. The zeros are expressed in terms of the zeros of Airy functions. The formulas are valid for all $s \geqslant 1$.

For $j_{a,s}\,, j'_{a,s}$ the formulas read as follows:

$$\begin{aligned} j_{a,s} &= az(\zeta) + \frac{f_1(\zeta)}{a} + \mathcal{O}(a^{-2}) \\ j'_{a,s} &= az(\zeta) + \frac{g_1(\zeta)}{a} + \mathcal{O}(a^{-2}) \end{aligned} \qquad a \to \infty \tag{3.7}$$

with

$$\begin{aligned} \zeta &= a^{-2/3}a_s \qquad \text{for } j_{a,s} \\ \zeta &= a^{-2/3}a'_s \qquad \text{for } j'_{a,s} \end{aligned} \tag{3.8}$$

(where a_s, a_s' is the sth negative zero of $A_i(z)$, $A_i'(z)$) and with $z(\zeta)$ defined implicitly by

$$\tfrac{2}{3}(-\zeta)^{3/2} = (z^2 - 1)^{1/2} - \text{arc}\cos(1/z), \qquad \zeta \leqslant 0, \qquad z \geqslant 1. \tag{3.9}$$

The functions f_1 and g_1 appearing in (3.7) are given by

$$f_1(\zeta) = -z(\zeta)\,\zeta^{-1}h(\zeta)\left[5\zeta^{-1}/48 + h(\zeta)\left(\frac{5}{z^2-1} + 3\right)\Big/24\right],$$

$$g_1(\zeta) = z(\zeta)\,\zeta^{-1}h(\zeta)\left[7\zeta^{-1}/48 + h(\zeta)\left(\frac{7}{z^2-1} + 9\right)\Big/24\right],$$

$$h(\zeta) = [\zeta(1 - z^2)]^{1/2}.$$

The expansions for $y_{a,s}$, $y_{a,s}'$ are given by (3.7) if in (3.8) the zeros b_s, b_s' of $B_i(z)$, $B_i'(z)$ are used instead of a_s, a_s'.

From numerical experiments it follows that (3.7) gives good initial approximations, even if a is rather small. For $a = 3$ and $s = 1$ the approximations based on (3.7) are of the same order of accuracy as those based on (3.3) or (3.6). However, the numerical process based on (3.7) is more intricate than the other one, since we need the inversion of (3.9), which must be carried out numerically. At the end of this section we pay attention to this problem, first, however, we give an indication for which values of a and s the approximations in (3.3), (3.6) and (3.7) are to be used.

A safe bound for using (3.3) and (3.6) is $s > 3a$, even when a is large. In order to take into account the smaller values of s and a we propose the following criterion: if

$$s \geqslant 3a - 8, \qquad a \geqslant 0$$

then (3.3) and (3.6) are to be used, otherwise (3.7). Consequently, for $0 \leqslant a \leqslant 3$, we use (3.3) and (3.6) for all s; if a becomes larger than 3 it is better to use (3.7) for the early zeros.

Finally we describe a method for the inversion of (3.9). By substituting $z = 1/\cos(x)$, $y = \tfrac{2}{3}(-\zeta)^{3/2}$, it follows that we need the inversion of

$$\text{tg}\, x - x = y, \qquad y \geqslant 0, \qquad 0 \leqslant x < \frac{\pi}{2}. \tag{3.10}$$

Since we need the solution of this equation for a first approximation in (3.7), it is not necessary to solve (3.10) with high accuracy precision. Four significant digits in x is enough.

For small y we can expand

$$x = \sum_{i=0}^{\infty} c_{2i+1} p^{2i+1}, \qquad p = (3y)^{1/3}, \tag{3.11}$$

where the first few coefficients are

$$c_1 = 1,\ c_3 = -2/15,\ c_5 = 3/175,\ c_7 = -2/1575.$$

For large y we write $x = \pi/2 - \xi$, and (3.10) becomes

$$\operatorname{tg} \xi = \frac{p}{1 - p\xi}, \qquad p = 1\Big/\left(y + \frac{\pi}{2}\right),$$

and again we can expand

$$\xi = \sum_{i=0}^{\infty} d_{2i+1} p^{2i+1}. \tag{3.12}$$

In this case the first coefficients are

$$d_1 = 1,\ d_3 = 2/3,\ d_5 = 13/15,\ d_7 = 146/105,\ d_9 = 781/315,\ d_{11} = 16328/3465.$$

If we use the above given coefficients c_i and d_i, both (3.11) and (3.12) give about four correct significant digits for $y = 1$.

4. An ALGOL 60 Procedure

The heading of the procedure given in this section reads as follows:

procedure *bess zeros* (*a*,*n*,*z*,*d*,*e*); **value** *a*,*n*,*d*,*e*; **real** *a*, *e*; **integer** *n*,*d*; **array** *z*;

The meaning of the formal parameters is:

a: ⟨arithmetic expression⟩;
the order of the Bessel function, $a \geqslant 0$.

n: ⟨arithmetic expression⟩;
the number of zeros to be computed, $n \geqslant 1$.

z: ⟨array identifier⟩;
array $z[1{:}\,n]$;
exit: $z[j]$ is the jth zero of the selected Bessel function.

d: ⟨arithmetic expression⟩;
the thoice of d determines the type of the Bessel function of which the zeros are to be computed:

if $d = 1$ then J_a,
if $d = 2$ then Y_a,
if $d = 3$ then J'_a,
if $d = 4$ then Y'_a.

e: ⟨arithmetic expression⟩;
the desired relative accuracy in the zeros;
e should be larger than the machine accuracy.

The procedure calls for the nonlocal procedure *besspqa*, which is published in Temme [4]. It computes the functions P_a, Q_a appearing in the relations

$$J_a(x) = [2/(\pi x)]^{1/2}[P_a(x) \cos \chi - Q_a(x) \sin \chi],$$
$$Y_a(x) = [2/(\pi x)]^{1/2}[P_a(x) \sin \chi + Q_a(x) \cos \chi],$$

with $\chi = x - \pi(2a + 1)/4$, and P_{a+1}, Q_{a+1} which are defined equivalently.

If for the computation of a zero more than 5 iterations are needed in the Newton process then the last computed value of the zero is accepted. Otherwise the computation is stopped if consecutive approximations agree within a relative precision e (see the heading of the procedure) with each other. From various tests it followed that not more than 3 iterations are needed if $e = {}_{10}\text{-}13$. In many cases only 1 iteration is needed for obtaining this precision.

```
procedure bess zeros (a,n,z,d,e); value a,n,d,e; real a,e; integer n,d; array z;
comment computes z[1],..., z[n], the first n zeros of a bessel function.
  the choice of d determines the type of the bessel function:
  if d = 1 then ja else
  if d = 2 then ya else
  if d = 3 then ja-prime else
  if d = 4 then ya-prime.
  a is the order of the bessel function, it must be non-negative.
  e is a measure for the relative accuracy;
begin real aa, a1, a2, b, bb, c, chi, co, mu, mu2, mu3, mu4, p, pi, pa, pa1, p0, p1, pp1,
  psi, q, qa, qa1, q1, qq1, ro, si, t, tt, u, v, w, x, xx, x4, y; integer j, s;

  real procecure fi( y); value y; real y;
  comment computes fi from the equation
                          tan(fi) − fi = y,      where   y ≥ 0.
  the relative accuracy is at least 5 digits;
  if y = 0 then fi: = 0 else
  if y > 100000 then fi: = 1.570796 else
  begin real r, p, pp;
    if y < 1 then
    begin p := (3 × y) ↑ (1/3); pp := p × p;
      p := p × (1 + pp × (−210 + pp × (27 − 2 × pp))/1575)
    end     else
    begin p := 1/(y + 1.570796); pp := p × p;
     p := 1.570796 − p × (1 + pp × (2310 + pp × (3003 + pp × (4818 + pp
                          × (8591 + pp × 16328))))/3465)
    end;
    pp := (y + p) × (y + p); r := (p − arctan(p + y))/pp;
    fi := p − (1 + pp) × r × (1 + r/(p + y))
  end fi;
```

```
real procedure r;
begin besspqa(a,x,e,pa,qa,pa1,qa1);
   chi := x − psi;
   si := sin(chi); co := cos(chi);
   r := if d = 1 then (pa × co − qa × si)/(pa1 × si + qa1 × co) else
         if d = 2 then (pa × si + qa × co)/(qa1 × si − pa1 × co) else
         if d = 3 then a/x − (pa1 × si + qa1 × co)/(pa × co − qa × si) else
                    a/x − (qa1 × si − pa1 × co)/(pa × si + qa × co)
end r;
pi := 4 × arctan(1); aa := a × a; mu := 4 × aa; mu2 := mu × mu;
mu3 := mu × mu2; mu4 := mu2 × mu2;
if d < 3 then
begin p := 7 × mu − 31; p0 := mu − 1; if 1 + p = p then p1 := q1 := 0 else
   begin p1 := 4 × (253 × mu2 − 3722 × mu + 17869) × p0/(p × 15);
         q1 := 1.6 × (83 × mu2 − 982 × mu + 3779)/p
   end
end    else
begin p := 7 × mu2 + 82 × mu − 9; p0 := mu + 3;
   if p + 1 = 1 then p1 := q1 := 0   else
   begin p1 := (4048 × mu4 + 131264 × mu3 − 221984 × mu2 − 417600 × mu +
         1012176)/(p × 60);
         q1 := 1.6 × (83 × mu3 + 2075 × mu2 − 3039 × mu + 3537)/p
   end
end;
t := if d = 1 ∨ d = 4 then 0.25 else 0.75; tt := 4 × t;
if d < 3 then
begin pp1 := 5/48; qq1 := −5/36 end else
begin pp1 := −7/48; qq1 := 35/288 end;
y := .375 × pi; bb := if a ⩾ 3 then a ↑ (−2/3) else 1;
a1 := 3 × a − 8; psi := pi × (.5 × a + .25);
for s := 1 step 1 until n do
begin if a = 0 ∧ s = 1 ∧ d = 3 then
   begin x := 0; j := 0 end else
   begin if s ⩾ a1 then
      begin b := (s + .5 × a − t) × pi; c := .015625/(b × b);
               x := b − .125 × (p0 − p1 × c)/(b × (1 − q1 × c))
      end    else
      begin if s = 1 then
           begin x := if d = 1 then −2.33811 else
                      if d = 2 then −1.17371 else
                      if d = 3 then −1.01879 else −2.29444
           end    else
```

```
        begin x := y × (4 × s − tt); v := 1/(x × x);
          x := − x ↑ (2/3) × (1 + v × (pp1 + qq1 × v))
        end;
        u := x × bb; v := fi(2/3 × (−u) ↑ 1.5);
        w := 1/cos(v); xx := 1 − w × w; c := sqrt(u/xx);
        x := w × (a + c/(48 × a × u) ×
        (if d < 3 then −5/u − c × (−10/xx + 6) else 7/u + c × (−14/xx + 18)))
      end; j := 0;
      l1: xx := x × x; x4 := xx × xx; a2 := aa − xx; ro := r; j := j + 1;
        if d < 3 then
        begin u := ro; w := 6 × x × (2 × a + 1); p := (1 − 4 × a2)/w;
          q := (4 × (xx − mu) − 2 − 12 × a)/w
        end    else
        begin u := −xx × ro/a2; v := 2 × x × a2/(3 × (aa + xx));
          w := 64 × a2 × a2 × a2;
          q := 2 × v × (1 + mu2 + 32 × mu × xx + 48 × x4)/w);
          p := v × (1 + (−mu2 + 40 × mu × xx + 48 × x4)/w)
        end;
      w := u × (1 + p × ro)/(1 + q × ro); x := x + w;
      if abs(w/x) > e ∧ j < 5 then goto l1
    end; z[s] := x
  end
end bess zeros;
```


ACKNOWLEDGMENTS

I wish to thank Mr. Rob Montijn of the Mathematical Centre for his help in programming and testing the ALGOL 60 program.

REFERENCES

1. M. ABRAMOWITZ AND I. A. STEGUN (Eds.), "Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables," Appl. Math. Ser. 55, U. S. Govt. Printing Office, Washington, D. C., 1965.
2. D. J. HOFSOMMER, *in* "Math. Tables Aid Comput. XII," pp. 58–60, 1958.
3. F. W. J. OLVER (Ed.), "Royal Society Mathematical Tables, Bessel Functions. Part III. *Zeros and Associated Values*," Vol. 7, Cambridge Univ. Press, London/New York, 1960.
4. N. M. TEMME, *J. Computational Phys.* **21** (1976), 343–350.

RECEIVED: July 18, 1978; REVISED: October 16, 1978

N. M. TEMME

Mathematisch Centrum
2e Boerhaavestraat 49
1091 AL Amsterdam,
Netherlands