High-accuracy approximation techniques for analytic functions

Charles Schwartz
Department of Physics, University of California, Berkeley, California 94720

(Received 13 June 1984; accepted for publication 21 September 1984)

A generalization of the familiar mesh point technique for numerical approximation of functions is presented. High accuracy and very rapid convergence may be obtained by thoughtful choice of the reference function chosen for interpolation between the mesh points. In particular, derivative operators are represented by highly nonlocal matrices; but this is no drawback when one has computing machines to perform the algebraic manipulations. Some examples are given from familiar quantum mechanical problems.

I. INTRODUCTION

The most common approach to numerical approximation of continuous functions involves the representation of the function $f(x)$ by its values on a set of mesh (net, or lattice) points x_n. Familiar formulas for the derivative, in the simple case of a uniform mesh $x_n = x_0 + nh$, are

$$f'(x_n) = [f(x_n) - f(x_{n-1})]/h + O(h),$$

or

$$f'(x_n) = [f(x_{n+1}) - f(x_{n-1})]/2h + O(h^2),$$

and for the integral [writing $f(x_n) = f_n$]

$$\int_{x_0}^{x_N} f(x)dx = (x_N - x_0)\frac{1}{N}\sum_{n=1}^{N} f_n + \frac{h(f_0 - f_N)}{2} + O(h^2).$$

These are simple to derive and simple to use but they have a very low order of accuracy in general. This is due to the fact that only *local* information about the function f is used in building the approximation.

The approach presented in this paper is based upon a *global* construction of an approximation for $f(x)$, which is still flexible and easy to use and involves only the values f_n at the selected mesh points. The purpose is to achieve very high-accuracy approximations: with a total of N mesh points it is nice to get errors which are as small as A^{-N} (or even $1/N!$), rather than the $1/N$, $1/N^2$, etc., errors which are characteristic of the usual methods. In this sense the present approach is somewhat reminiscent of Gaussian quadrature; but it is rather more general in its construction and its application. The present method may also be described as a generalization of Lagrange interpolation; and the method of "collocation" is also related.

The general method will be described, along with a formal method for error analysis; then several examples will be given, mostly concerned with solving differential equations familiar in quantum mechanics.

II. THE GENERAL METHOD

To approximate a given function $f(x)$ we start by choosing a reference function $u(x)$ that has simple zeros at the (real) points $x = x_n$. The construction of an interpolating function $\bar{f}(x)$ to approximate $f(x)$ is

$$\bar{f}(x) \equiv \sum_m f_m \frac{u(x)}{x - x_m}\frac{1}{a_m}, \quad \text{where } a_n = u'(x_n). \tag{1}$$

At the points $x = x_n$, $\bar{f}(x)$ takes on the values $f_n = f(x_n)$. We should choose the reference function $u(x)$ to have analytic properties similar to those of the desired function $f(x)$; the error analysis and examples to follow will help show what this means.

To approximate the derivatives of the function f, we take derivatives of the interpolating function (1), evaluated at the mesh points x_n. The resulting formulas are [$b_n = u''(x_n)$ and $c_n = u'''(x_n)$]

$$\left.\frac{d\bar{f}}{dx}\right|_{x_n} = \sum_m f_m \begin{cases} m = n: & \dfrac{b_n}{2a_n} \\ & = \displaystyle\sum_m D_1(n, m) f_m, \\ m \neq n: & \dfrac{1}{(x_n - x_m)}\dfrac{a_n}{a_m} \end{cases} \tag{2}$$

$$\left.\frac{d^2\bar{f}}{dx^2}\right|_{x_n} = \sum_m f_m \begin{cases} m = n: \dfrac{c_n}{3a_n} \\ m \neq n: \dfrac{1}{(x_n - x_m)}\dfrac{b_n}{a_m} - \dfrac{2}{(x_n - x_m)^2}\dfrac{a_n}{a_m} \end{cases}$$

$$= \sum_m D_2(n, m) f_m. \tag{3}$$

In case the function $u(x)$ obeys an equation of the form $u''(x) = W(x)u(x)$, then there is a simplification of the above formulas: $b_n = 0$, $c_n = W_n a_n$; and the matrices representing the derivative operators can be put into a symmetric form.

To approximate the integrals of f we get the formulas

$$\int_{x_0}^{x_n} \bar{f}(x)dx = \sum_m Q_1(n, m) f_m,$$

$$\text{where } Q_1(n, m) = \int_{x_0}^{x_n} dx \frac{u(x)}{a_m(x - x_m)}, \tag{4}$$

$$\int_{x_0}^{x_n} dx \int_{x_0}^{x} dy\, \bar{f}(y) = \sum_m Q_2(n, m) f_m,$$

$$\text{where } Q_2(n, m) = \int_{x_0}^{x_n} dx \frac{u(x)(x_n - x)}{a_m(x - x_m)}. \tag{5}$$

In the case where $u(x)$ is an orthogonal polynomial times a weight function and the integral is taken over the entire domain, then (4) yields the usual Gaussian quadrature results.

The above general method is very flexible since one can choose any reference function $u(x)$. The quantities that enter into the matrices for the derivative operators (D) or integral operators (Q) may be determined by some computational procedure, if not readily expressed in closed form, depending

 0022-2488/85/030411-05$02.50

on this choice of u. A practical question is the following: Does one pay a heavy price by having the derivative operator so nonlocal, since one may be forced to invert or otherwise manipulate these matrices in order to solve differential equations? A computing machine can readily carry out such matrix operations numerically for moderate-sized matrices. Furthermore, when one gets into partial differential equations the usual mesh point methods already require working with sizable matrices for the derivative operators. Since the whole point of the present method is to construct approximate but accurate functions $\bar{f}$ in terms of a *small* number of mesh points it is anticipated that the net result should be a general increase in efficiency of computation.

Now we present a general approach for analyzing the error in approximating the function f by $\bar{f}$, once u is chosen. Assume that both $f(x)$ and $u(x)$ are analytic functions in some appreciable domain of the complex plane surrounding the set of mesh points x_n. Then, using the contour around $z = x$ [see Fig. 1(a)] we have the identity

$$f(x) = \oint \frac{dz}{2\pi i} \frac{f(z)}{z - x} \frac{u(x)}{u(z)}. \tag{6}$$

One may take the point x to be slightly off the real axis to be assured that there is no difficulty in this integral representation when x approaches one of the mesh points x_n, where u vanishes. Now move the contour of integration to the large loop C and the small circles around each of the points $z = x_n$ [see Fig. 1(b)]. Calculating the residues at each x_n we have the exact result

$$f(x) = \sum_n f(x_n) \frac{u(x)}{(x - x_n)} \frac{1}{u'(x_n)} + \epsilon. \tag{7}$$

The first term on the right-hand side of (7) is just the approximation $\bar{f}(x)$ defined in (1); the second term ϵ is the error and is given by the integral over the contour C of the expression (6). A general argument about the smallness of this error is as follows: Since $u(z)$ has many oscillations along the real axis, one expects it to grow rapidly along the imaginary directions in the z plane; and it is this factor in the denominator that should make the error ϵ decrease rapidly as the mesh points become more closely spaced. A concrete example will be studied in the next section.

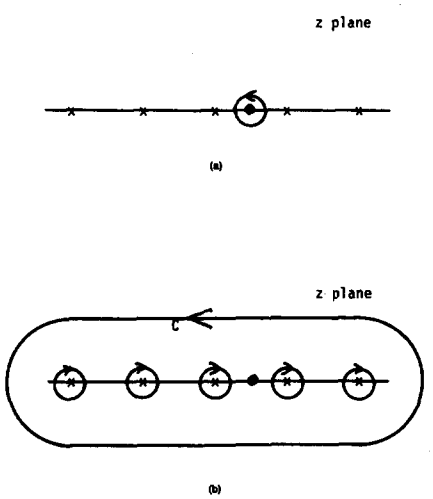

FIG. 1. Integration contours for error analysis. The several crosses × represent the mesh points x_n. The solid dots ● represent the point $z = x$.

III. EXAMPLE I

Consider the infinite line, $-\infty < x < \infty$, and the choice of a uniform sequence of mesh points: $x_n = nh$, $n = 0, \pm 1, \pm 2, \pm 3, \ldots$. Then we take the reference function $u(x) = \sin \pi x/h$; and the matrices for d/dx and d^2/dx^2 become

$$D_1(n, m) = \frac{1}{h} \begin{cases} n = m: & 0, \\ n \neq m: & (-1)^{n-m}/(n-m); \end{cases} \tag{8}$$

$$D_2(n, m) = \frac{1}{h^2} \begin{cases} n = m: & -\pi^2/3, \\ n \neq m: & -2(-1)^{n-m}/(n-m)^2. \end{cases} \tag{9}$$

The matrices for the indefinite integrals become

$$Q_1(n, m) = (h/\pi)[S_i((n-m)\pi) + \pi/2], \tag{10}$$

$$Q_2(n, m) = (h^2/\pi^2)[(n-m)\pi(S_i((n-m)\pi) + \pi/2) + (-1)^{n-m}], \tag{11}$$

where

$$S_i(x) = \int_0^x dt \frac{\sin t}{t}. \tag{12}$$

The only familiar result contained here is for the infinite integral

$$Q_1(-\infty, \infty) = h, \quad \int_{-\infty}^{\infty} f(x)dx = \sum_{n=-\infty}^{\infty} hf(nh) + \epsilon. \tag{13}$$

The high accuracy of the trapezoidal rule for the infinite integration of analytic functions has been explored elsewhere.[1] This is the "Gaussian quadrature" formula for the infinite line.

Obviously, if this approach is to be practical, we should be dealing with functions $f(x)$ which decrease very rapidly as x grows large, so that the infinite sums over the mesh points can be truncated effectively. Thus we have two sources of error to analyze: ϵ_A from (7) due to the analytic approximation and ϵ_T due to the truncation. A good strategy will be to choose a relation between the mesh spacing h and the truncation at $|n| < N$ so that ϵ_A and ϵ_T are approximately equal to each other. This will avoid wasting effort on too small a mesh (when truncation error dominates) or on too large a cutoff (when mesh error dominates).

For illustration, consider that the function $f(x)$ is known to be analytic everywhere in the finite complex plane and is dominated at large distances by the behavior

$$e^{-ax^p}. \tag{14}$$

Then we have

$$\epsilon_T \approx e^{-a(Nh)^p}. \tag{15}$$

For the mesh size error, we see that the error in (7) involves the integral over the large contour C in Fig. 1(b); and with $u(z) = \sin \pi z/h$, we see that this error is given roughly by

$$\epsilon_A \approx \int dz\, e^{\pm i\pi z/h} e^{-az^p}. \tag{16}$$

This integral may be estimated by the stationary phase method (we are interested in the dependence of ϵ_A on h for small h) and we find

$$\epsilon_A \approx e^{-bh^{-q}}, \quad \text{where } q = p/(p-1), \tag{17}$$

and

$$b=\left(\frac{\pi^p}{ap}\right)^{1/(p-1)}\left(\frac{p-1}{p}\right)\sin\left[\pi\frac{1}{2(p-1)}\right].$$

Equating the results (15) and (17) we find the optimum choice of h, given N,

$$h=(b/aN^p)^{((p-1)/p^2)}, \tag{18}$$

and along with this is the error estimate

$$\epsilon\approx e^{-CN}, \tag{19}$$

where

$$C=b(a/b)^{1/p}.$$

This result—exponential decrease of the error with increasing number of mesh points—is most exciting. Rather than trying to make this rough error analysis more respectable I shall proceed to some numerical experiments.

The one-dimensional Schrödinger equation

$$\left(-\frac{1}{2}\frac{d^2}{dx^2}+\frac{1}{k}x^k\right)\psi(x)=E\psi(x), \tag{20}$$

for $k=2,4,\ldots$, is an eigenvalue problem in which the solution $\psi(x)$ has the characteristics described above: it is an analytic function for all finite x and has the asymptotic behavior for large x given by (14) with

$$p=\frac{k+2}{2},\quad a=\frac{2}{k+2}\left(\frac{2}{k}\right)^{1/2}. \tag{21}$$

Thus we predict the optimum convergence strategy, (18) and (19):

$$h=(\pi/N)^{1/2},\quad \epsilon\approx e^{-1.57N},\quad \text{for } k=2, \tag{22}$$

and

$$h=1.78N^{-2/3},\quad \epsilon\approx e^{-1.32N},\quad \text{for } k=4. \tag{23}$$

Taking account of the symmetry, $\psi(x)=\pm\psi(-x)$, and choosing the mesh points $x_n=(n-1/2)h$, for $n=1,2,\ldots,N$, and using (9) for the second derivative operator, Eq. (20) was represented as an $N\times N$ matrix eigenvalue problem which the computer solved for the sequence $N=1,2,3,\ldots$.

The values of h were chosen according to (22) and (23) with $(N+1/2)$ replacing N. The numerical results for the ground state eigenvalue showed very rapid convergence:

$k=2$, $E=0.5$:

$N=1$ error 7×10^{-2}, $N=2$ error 3×10^{-3},

$N=3$ error 2×10^{-4}, $N=4$ error 1×10^{-5},

with a good fit to the formula $\epsilon\approx e^{-2.9N}$; (24)

$k=4$, $E=0.420\,804\,974\,475$:

errors of -7×10^{-2}, 6×10^{-3}, 2×10^{-4},

7×10^{-6} for $N=1,2,3,4$,

with a good fit to the formula $\epsilon\approx e^{-2.8N}$. (25)

These are very gratifying results: high accuracy at low-order approximation with very rapid improvement as the order of approximation is increased. Indeed, these numerical results for the x^4 potential converge even more rapidly than the results of a Rayleigh–Ritz variational calculation that used a harmonic oscillator basis.[2] The predicted exponential form of convergence (19) was well borne out by the numerical results; however, there is some discrepancy between the predicted and observed values of the decay constant C. The decay constants in (22) and (23) describe the error in the wave function and perhaps one ought to square these errors for the eigenvalue: the values $2C=3.14$ $(k=2)$ and $2C=2.64$ $(k=4)$ are not so far from the observed results 2.9 and 2.8, respectively.

In the computations described above the matrix eigenvalue was determined by a direct numerical method (which requires computing time proportional to the cube of the dimension of the matrix). For these one-dimensional problems the size of the matrix is so small that this is no problem. However, when one envisions going to multidimensional problems with a much larger dimension for the matrix of the partial differential operators involved then some alternative method of manipulating the matrix may be necessary. There are a variety of iterative techniques commonly used for large matrix manipulations (inversion, diagonalization, etc.) and the critical question is how fast such iterative methods converge. As an experiment I tried solving the above-mentioned Schrödinger equation iteratively by a few different strategies and found convergence that varied from fair (about 1/2 decimal accuracy gained per iteration) to very good (several decimals gained per iteration.) As with all iterative schemes it is valuable to have a good starting guess for the solution; and the attempts I made worked best when I used the resulting eigenvector for the solved $N-1$ problem to get a starting estimate for the N-problem eigenvector through use of the basic interpolation formula (1).

IV. EXAMPLE II

For a problem on the semi-infinite line $0\leqslant r<\infty$ consider the Schrödinger equation for the hydrogen atom:

$$\left[-\frac{1}{2}\frac{d^2}{dr^2}+\frac{1}{2}\frac{l(l+1)}{r^2}-\frac{1}{r}\right]\phi(r)=E\phi(r). \tag{26}$$

At the origin ϕ goes to zero as r^{l+1} and at infinity it goes exponentially to zero for bound states (E negative eigenvalues).

To choose a good reference function $u(r)$ we would like a function which has analytic properties similar to ϕ for finite r and also has many zeroes. It is known that the solution of (26) for $E=0$ is given in terms of a Bessel function:

$$\phi_{E=0}(r)=r^{1/2}J_{2l+1}((8r)^{1/2}). \tag{27}$$

This leads to the choice

$$u(r)=r^{1/2}J_{2l+1}\left[\left(\frac{8r}{h}\right)^{1/2}\right], \tag{28}$$

with the mesh points

$$r_n=(h/8)y_n^2,\quad J_{2l+1}(y_n)=0,\quad n=1,2,3,\ldots. \tag{29}$$

With the change of variables

$$\phi_m=\chi_m J'_{2l+1}(y_m), \tag{30}$$

we reduce the differential equation (26) to the algebraic form

$$\sum_{m\neq n}\frac{64}{h^2}\frac{1}{(y_n^2-y_m^2)^2}\chi_m+\frac{8}{3h^2}\left[\frac{8l(l+1)}{y_n^4}+\frac{1}{y_n^2}\right]\chi_n-\frac{8}{hy_n^2}\chi_n=E\chi_n. \tag{31}$$

Numerical computations of the ground state eigenvalue ($l = 0$) were carried out for a sequence of mesh scales ($h = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}$) and a sequence of truncations ($m, n = 1,2,3,...,N$, for $N = 1,2,3,4,...$). At each h value the error would decrease rapidly with increasing N (about one decimal place improvement per unit step in N) until it reached a saturation value; then one would need to decrease h to gain further improvement. Taking the results from these saturation points one could deduce an overall convergence rate which went approximately as

$$\epsilon \approx 10^{-N}. \tag{32}$$

This is an experimental result; I have not tried (as in the previous section) to carry out an analytical estimate of the expected error. This is a very rapid convergence rate, indicating that this is an extremely powerful approximation technique for atomic wave functions.

The major nuisance in this calculation was the need to generate zeroes of a Bessel function. As an alternative, I tried using the reference function

$$u(r) = r^{l+1/2} \sin[\pi(r/h)^{1/2}], \tag{33}$$

with the mesh points given by

$$r_n = hn^2, \quad n = 1,2,\ldots. \tag{34}$$

I will not give details but merely state the results of this approach. The matrix turns out to be unsymmetrical but this poses no serious problem. The ground state eigenvalue computation converges quite well, only slightly slower than the first approach:

$$\epsilon \approx 10^{-(2/3)N}. \tag{35}$$

V. EXAMPLE III

For problems on a finite interval one usually works with either polynomials or Fourier series as a basis for approximations. I will give a couple of illustrations based upon the latter.

Suppose we want to approximate the function $f(x)$ on the interval $[0, 1]$ with the boundary conditions $f(0) = f(1) = 0$. One choice of the reference function is

$$u(x) = \sin(N+1)\pi x, \tag{36}$$

which satisfies the same boundary conditions as $f(x)$ and has the interior mesh points

$$x_n = n/(N+1), \quad n = 1,2,3,\ldots,N. \tag{37}$$

(Here life is simpler since we do not have to deal with two variables, h and N, but only one, N.)

If we follow the original prescription for building the approximation (1), then we will have functions that are not simply a finite set of trigonometric functions. An alternative is to divide $u(x)$ by something like $\sin\alpha(x - x_n)$, rather than just $(x - x_n)$. After some experimentation I was able to find the following representation, which is equivalent to a truncated Fourier series:

$$\bar{f}(x) = \sin(N+1)\pi x \sum_{n=1}^{N} f_n \frac{(-1)^n}{2(N+1)} \times [\cot(\pi/2)(x - x_n) - \cot(\pi/2)(x + x_n)]. \tag{38}$$

From this the second derivative was calculated to be

$$\bar{f}''|_{x_n} = \sum_{m=1}^{N} f_m \pi^2 \begin{cases} m = n: & -\frac{1}{3}(N+1)^2 - \frac{1}{6} + \frac{1}{2}\csc^2\pi\frac{n}{N+1}, \\ m \neq n: & \frac{(-1)^{n+m}}{2}\left[-\csc^2\left(\frac{\pi}{2}\frac{n-m}{N+1}\right)\right] + \csc^2\left(\frac{\pi}{2}\frac{n+m}{N+1}\right). \end{cases} \tag{39}$$

The eigenvalues of this matrix (39), in units of $-\pi^2$, are

$$1 \text{ (for } N = 1); \quad 1,4 \text{ (for } N = 2); \quad 1,4,9 \text{ (for } N = 3); \ldots. \tag{40}$$

An alternative problem is one with periodic boundary conditions:

$$f(\varphi + 2\pi) = f(\varphi). \tag{41}$$

For N *odd* we construct the approximate function

$$\bar{f}(\varphi) = \sin\frac{N}{2}\varphi \sum_{n=1}^{N} \frac{f_n}{\sin\frac{1}{2}(\varphi - \varphi_n)} \frac{(-1)^n}{N}, \tag{42}$$

with mesh points

$$\varphi_n = 2\pi n/N; \tag{43}$$

and the second derivative operator is represented by

$$\bar{f}''|_{\varphi_n} = \sum_{m=1}^{N} f_m \begin{cases} m = n: & -\frac{1}{12}(N^2 - 1), \\ m \neq n: & \frac{(-1)^{n-m+1}}{2} \frac{\cos\phi_{nm}}{\sin^2\phi_{nm}}, \end{cases} \tag{44}$$

where

$$\phi_{nm} = (n - m)\pi/N.$$

This matrix has the expected eigenvalues: 0, -1(twice), -4(twice), etc.

For a numerical application I considered the problem of the Schrödinger pendulum:

$$\left[-\frac{1}{2}\frac{d^2}{d\theta^2} + g^2(1 - \cos\theta)\right]\psi(\theta) = E\psi(\theta). \tag{45}$$

Using (44), the two lowest eigenvalues were computed for a sequence of values of N, for two different values of g. No account was taken of the reflection symmetry. Results, shown in Table I, exhibit the fastest convergence yet seen. The calculation was repeated shifting the coordinate in (42) by 90° [actually, by changing $\cos\theta$ to $\sin\theta$ in (45)] and these results were even better, by up to two decimal places accuracy at each N. For comparison, a variational calculation of (45) using a truncated Fourier series with corresponding number of terms gave results which were in between those of the two computations just described.

Some previous work on trigonometric interpolation of periodic functions[3] bears resemblance to what has been presented here; but the formula (44) appears to be new. I will

TABLE I. Eigenvalues of the Schrödinger pendulum, Eq. (45).

	$g=1$		$g=3$	
N	E_0	E_1	E_0	E_1
3	0.29	1.71	0.33	13.7
5	0.457	1.382	0.89	7.09
7	0.464 86	1.343 98	1.33	4.97
9	0.464 934 9	1.343 362 9	1.455	4.43
11	0.464 935 147 34	1.343 360 133	1.467 3	4.345
13	0.464 935 147 7119	1.343 360 128 403	1.468 031	4.337 52
15	0.464 935 147 7122[a]	1.343 360 128 3991[a]	1.468 053 5	4.337 179 2
17			1.468 054 007	4.337 170 39
19			1.468 054 013 55	4.337 170 257 1
21			1.458 054 013 609	4.337 170 255 64

[a] Machine accuracy not reliable after this point.

confess, however, that the formulas (44) and (39) were first obtained by Fourier transform calculation.

VI. SUMMARY

The general approach presented here should be very powerful in obtaining efficient and accurate numerical computational results in the form of systematic approximations to functions that are very smooth. The high accuracy and rapid convergence usually associated with variational techniques is obtained along with the simplicity of mesh techniques. The key link between these two methods is the judicious choice of the reference function; here is where the human being contributes analytical insights in setting up the problem, while leaving the later computational tedium to the machine.

The numerical examples shown here were restricted to the solution of one-dimensional differential equations (eigenvalue problems); and the results were excellent. There should be many other areas of application for this general method of approximating analytic functions.

ACKNOWLEDGMENTS

I have benefitted from conversations with Professor K. Miller and Professor A. Chorin.

APPENDIX: THE SEMI-INFINITE INTERVAL

For the infinite interval $[-\infty, \infty]$ a general interpolation scheme was given, with uniform intervals, based upon trigonometric functions as used in Fourier integrals. For the finite interval problem, alternative schemes were again based upon trigonometric functions, this time as they are used in discrete Fourier series. What follows here is a generalization of the study for the semi-infinite interval $[0, \infty]$ based upon Bessel functions.

Choose the reference function, for unspecified value of ν,

$$u(x) = x^{-\nu/2} J_\nu((x/h)^{1/2}), \tag{A1}$$

which has the mesh points $x_n = hy_n^2$, $n = 1,2,...$, where y_n is the nth zero of the Bessel function $J_\nu(y)$ on the positive real axis. Next, construct the identity integral representation, as in (6):

$$f(x) = \oint \frac{dz}{2\pi i} \frac{u(x)}{u(z)} \frac{f(z)}{z-x}, \tag{A2}$$

with the contour a small circle around $z = x$. Now, move the contour following the same general procedure illustrated in Fig. 1. The form of (A1) was chosen so that $u(z)$ is analytic in the domain $\mathrm{Re}(z) \leqslant 0$ as well as > 0. Assuming $f(z)$ is analytic in some sizable region around the positive real axis, we expect exponentially small errors to the approximate interpolation function $\bar{f}(x)$ that results from the residues at each of the zeroes of $u(z)$:

$$\bar{f}(x) = \sum_n \frac{f(hy_n^2)}{x - hy_n^2} \frac{u(x)}{J'_\nu(y_n)} 2h^{1+\nu/2} y_n^{1+\nu}. \tag{A3}$$

From this one can calculate the definite integral

$$\int_0^\infty dx\, x^\nu \bar{f}(x) = \sum_n f(hy_n^2) 4h^{\nu+1} \left(\frac{y_n^\nu}{J'_\nu(y_n)}\right)^2. \tag{A4}$$

This is a new "Gaussian quadrature" formula, or rather a family of such for any value of ν. In the special cases $\nu = \pm\frac{1}{2}$ this formula reduces to the trapezoidal rule (13). What is interesting about this formula is the fact that the points $x_n = hy_n^2$ at which one evaluates the function $f(x)$ are spaced farther and farther apart as n increases.

[1] C. Schwartz, J. Comput. Phys. 4, 19 (1969).

[2] C. Schwartz, Ann. Phys. (NY) 32, 277 (1965).

[3] H. Kreiss and J. Olinger, "Methods for the Approximate Solution of Time Dependent Problems," GARP Publications Series No. 10, February 1973, p. 42 ff.