Algorithm 794: Numerical Hankel Transform by the Fortran Program HANKEL

THOMAS WIEDER
Darmstadt University of Technology

The numerical evaluation of the Hankel transform poses the problems of both infinite integration and Bessel function calculation. Using the corresponding numerical program routines from the literature, a Fortran program has been written to perform the Hankel transform for real functions, given either in analytical form as subroutines or in discrete form as tabulated data.

Categories and Subject Descriptors: D.3.2 [**Programming Languages**]: Language Classifications—*FORTRAN 77*; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems—*Computation of transforms* (e.g., Fast Fourier Transfor)

General Terms: Algorithms

Additional Key Words and Phrases: Hankel transform, numerical analysis

1. PROBLEMS WITH THE HANKEL TRANSFORM

The real Hankel transform $\mathcal{H}\nu(\xi, f(x))$ of a real function $y = f(x)$ requires the evaluation of the infinite integral

$$\mathcal{H}_\nu(\xi, f(x)) = \int_0^\infty (x\xi)^{1/2} f(x) J_\nu(x\xi) dx \tag{1}$$

where $\nu \in \mathrm{R}$ is the order of the transform; $\xi \in \mathrm{R}$ is the transformation parameter with $0 < \xi$; and $J_\nu(x)$ is the Bessel function of the first kind. Precise definitions of $\mathcal{H}_\nu(\xi, f(x))$ and conditions on $f(x)$ can be found in

Author's address: FB Materialwissenschaft, FG Strukturforschung, Darmstadt University of Technology, Petersenstr. 23, Darmstadt, D-64287, Germany; email: wieder@hrzpub.tu-darmstadt.de; http://www.tu-darmstadt.de/~wieder.

Erdelyi et al. [1954], Korn and Korn [1968], and Sneddon [1955]. For $-1/2 < \nu$ the Hankel transform is self-reciprocal, i.e., with $g(\xi) = \mathcal{H}_\nu(\xi, f(x))$ given one finds $f(x)$ from $f(x) = \int_0^\infty (x\xi)^{1/2} g(\xi) J_\nu(x\xi) d\xi$. The integral (1) will exist if $f(x) \in L_1$, i.e., if the integral $\int_0^\infty \|f(x)\| dx$ exists. Analytical solutions of (1) are known for a variety of functions $f(x)$ [Erdelyi et al. 1954]. By use of a symbolic computation program further analytical solutions may be found.[1]

Numerical solutions also are of interest and unavoidable, if $f(x)$ is not known analytically. For example, measured data may be given (sampled) as a table of N data pairs $(x_1, f(x_1)), \ldots, (x_N, f(x_N))$. Several computer programs are available to solve (1) numerically. Siegman used a nonlinear change of variables to convert the one-sided Hankel transform into a two-sided cross-correlation integral [Siegman 1980]. The algorithm is particularly fast and can be applied to sampled data, but requires a sampling at exponentially increasing x-values. Piessens presented a program which solves (1) making use of numerical integration [Piessens 1982]. The approximation for the Bessel function Piessens used allowed him to write the infinite integral as the sum of an integral over a finite interval and of a Fourier-sine and Fourier-cosine transform. The restriction to integer ν-values (with $0 \leq \nu \leq 10$) prevents the application of that program to transforms with noninteger ν which in particular arises if spherical Bessel functions are involved. In the same year, Anderson [1982] introduced his algorithm for the Hankel transform by using related and lagged convolutions. Again, the algorithm is restricted to integer ν. The noninteger Hankel transform with $\nu = j + 1/2$ (where j is an integer) is known as spherical Bessel transform. Talman [1983] provided a corresponding program for sampled data, but the x-values have to be distributed uniformly in $\ln(x)$.

In this work, a direct numerical approach to solve Eq. (1) is presented which consists of two parts: (1) the calculation of the infinite integral and (2) the calculation of the Bessel function $J_\nu(x)$.

(1) Several strategies can be found in literature for the numerical evaluation of infinite integrals. None of them is strictly valid, since any numerical procedure has to map the infinite integral onto a finite one. A frequent strategy is to split the integral into a finite one from lower limit 0 to upper limit x_l (where x_l stands for the upper limit of the finite integral, i.e., the x-value at which the integral in (1) is split) and an infinite one from x_l to ∞. Furthermore, no procedure can account for the uncountable variety of the integrand form. Of particular difficulty are infinitely oscillating functions as integrand, as $J_\nu(x)$ is.

[1]The function `inttrans[hankel]` of the symbolic computation program Maple [Waterloo Maple 1996] looks up known Hankel transform in an internal table.

(2) Many books give details on the definition (e.g., Courant and Hilbert [1968] and Whittaker and Watson [1978]) and on the numerical evaluation of $J_\nu(x)$. For large x an asymptotic expansion is available which reads in its simplest form

$$J_\nu(x) = \left(\frac{2}{\pi x}\right)^{1/2} \cos\left(x - \frac{\nu\pi}{2} - \frac{\pi}{4}\right) + O(\|x\|^{-\frac{3}{2}}). \tag{2}$$

2. AN APPROXIMATION TO THE HANKEL TRANSFORM

The approach presented here combines two strategies from (1) and (2) in an appropriate manner with respect to Eq. (1). Since $J_\nu(x)$ is oscillatory, the integration procedure should be able to integrate an oscillating integrand. A particular problem is posed by the fact that the oscillations of $J_\nu(x)$ are not of constant period p (the zeros of $J_\nu(x)$ are not equidistantly spaced). Fortunately (2) is of constant period $p = 2\pi$; therefore, we can take advantage of splitting the integral into two parts and using the asymptotic form (2). The present approach is

$$\mathcal{H}_\nu(\xi, f(x)) \approx \int_0^{x_l} (x\xi)^{1/2} f(x) J_\nu(x\xi)\,dx + \int_{x_l}^{\infty} \left(\frac{2}{\pi}\right)^{1/2} f(x) \cos\left(x\xi - \frac{\nu\pi}{2} - \frac{\pi}{4}\right) dx. \tag{3}$$

We are not able to estimate the error of the approximation (3). Of decisive importance is the choice of x_l, the "splitting value." No general rule can be given, only a few guidelines, since x_l depends on $f(x)$. In any case, x_l should not violate the condition $x_l \leq x_{as}$, where x_{as} denotes the x-value above which we apply (2). A condition on the upper limit for x_l is obviously $x_l < x_{max}$, where x_{max} is the largest real number of the computer system in use. The optimal value for x_l may be suggested by $f(x)$. In case of tabulated data, one has obviously $x_l \leq x_N$. A further favorable case will be a damped function, which may be considered equal to zero above a certain value x_d, so that one will set $x_l = x_d$. Functions containing a damping factor $\exp(-ax)$ for example will belong to this case. Functions $f(x)$ which are oscillating will cause particular problems for approach (3). The period p_f of $f(x)$ will mostly be different from 2π. Then any particular integration algorithm depending on $p = \text{const}$ above x_l will fail. *In this case the user is forced to cut any oscillating part in $f(x)$ above x_l.* Any $f(x)$ for which $\int_0^\infty \|f(x)\|dx$ does not exist will cause our approach to fail. Speaking in a loose manner, we say that favorable functions should obey $f(x) \to 0$ for $x \to \infty$.

Table I. Ranges of Some Major Variables in HANKEL

Variable	Function	Minimal Value	Maximal Value
x	independent variable in $f(x)$	2.23D-307	1.79D+308
ξ	parameter in $\mathcal{H}_\nu(\xi, f(x))$	2.23D-307	1.79D+308
ν	order of the transform	-100.0	100.0
x_{as}	onset of (2) for $J_\nu(x)$	100.0	—

3. THE IMPLEMENTATION OF THE APPROXIMATION

The actual implementation of (3) has been done in the form of a Fortran program named HANKEL. The finite integral $\int_0^{x_l} \ldots dx$ is carried out by INTHP from Sikorsky et al. [1984]. Any other reliable subroutine could be taken here, but our tests lead to the choice of INTHP, partially because it can integrate even sharply peaked functions like $\sin(x) / x$, but mainly because it is designed for infinite integrals also (and thus could be used for that purpose, too). The infinite integral $\int_{x_l}^{\infty} \ldots dx$ is left to subroutine OSCINT from Lyness and Hines [1986] which is specially designed for an oscillating integrand of constant period p. OSCINT will fail if $p \neq$ const for $x > x_l$. Any oscillating part of $f(x)$ must be canceled above x_l.

Table I lists all important parameter ranges set in HANKEL. The values for x and ξ are machine dependent. The user has to tell HANKEL the actual computer platform used. This is done in subroutine D1MACH. The values listed in Table I are valid for an IBM RS 6000 , as well as for a Pentium PC (IEEE IBM/XT standard).

4. TESTS OF PROGRAM HANKEL

Although classification is desirable, we are not able to classify the functions $f(x)$ for which HANKEL will give correct transforms. Moreover, even for a given $f(x)$ we cannot provide an error estimation in closed form. Therefore, we tested HANKEL using the test functions listed in Table II. Their implementations are provided as examples in DHFUNC of HANKEL's source code, so one can take them as templates for one's own function implementations. The following figures give $\mathcal{H}_\nu(\xi, f(x))$ according to the analytical solution $\mathcal{H}_{exa}$ [Erdelyi et al. 1954] (plotted as line) and the residual $r(\xi)$ of the numerical solution $\mathcal{H}_{num}$ with $r = \mathcal{H}_{num} - \mathcal{H}_{exa}$ (plotted as $\diamond$) for test functions 1–4. Satisfactory agreement (= small residual) between the analytical and numerical solutions was achieved for functions 1, 2 and 6, whereas for 3 and 4 some deviations were observed. In the specific case of function 3, for $x = 1$ the transform does not exist; thus some deviations around this singularity are not too surprising.

Table III summarizes for test functions 1–4 the mean residuals $\bar{r} = \Sigma_{i=1}^{N}(\mathcal{H}_{num,\,i} - \mathcal{H}_{exa,\,i})/N$ and the standard deviations $\sigma(r) = (\Sigma_{i=1}^{N}((\mathcal{H}_{num,\,i} - \mathcal{H}_{exa,\,i}) - \bar{r})^2)^{1/2}$ of the residuals. Also given are the relative residuals $\bar{r}_{rel} = \Sigma_{i=1}^{N}((\mathcal{H}_{num,\,i} - \mathcal{H}_{exa,\,i})/\mathcal{H}_{exa,\,i})/N$ and the relative standard

Table II. List of Test Functions

No.	$f(x)$	Analytical Solution [Erdelyi et al. 1954] $\mathcal{H}_\nu(\xi, f(x))$
1	$x^{\nu+1/2}$	$\xi^{-1/2}\boldsymbol{J}_{\nu+1}(\xi)$
2	$x^{-1/2}\exp(-x)$	$\xi^{1/2-\nu}(1+\xi^2)^{-1/2}((1+\xi)^{1/2}-1)^\nu$
3	$x^{-1/2}\sin(x)$	$\cos(1/2\pi\nu)\xi^{\nu+1/2}(1-\xi^2)^{-1/2}(1+(1-\xi^2)^{1/2})^{-\nu}$ for $0<\xi<1$ $\xi^{1/2}(\xi^2-1)^{-1/2}\sin(\nu\arcsin(1/\xi))$ for $1<\xi<\infty$
4	$x^{-1/2}\boldsymbol{J}_{\nu-1}(x)$	$\xi^{-\nu+1/2}$ for $1<\xi<\infty$
5	$\cos^2(x)\exp(-0.1x)$	—
6	$x^{-1/2}$	$y^{-1/2}$

Table III. Residuals $\bar{r}$ for the Hankel Transforms of the Test Functions

Function No.	1	2	3	4
$\bar{r}$	0.010×10^{-9}	0.953×10^{-6}	0.188×10^{-3}	0.072×10^{-3}
$\sigma(r)$	1.854×10^{-9}	23.01×10^{-6}	545.795×10^{-3}	31.043×10^{-3}
$\bar{r}_{rel}$	-0.0006×10^{-3}	0.004×10^{-3}	0.018×10^{3}	-58.915×10^{-3}
$\sigma(r)_{rel}$	0.116×10^{-3}	0.120×10^{-3}	3.717×10^{3}	9371.676×10^{-3}
$\bar{r}/\epsilon$	0.045×10^{6}	4.292×10^{9}	849×10^{9}	3283×10^{9}
$\sigma(r)/\epsilon$	8.352×10^{6}	103.640×10^{9}	2458046.38×10^{9}	139806.914×10^{9}
$\bar{r}_{rel}/\epsilon$	-3.077×10^{9}	22.064×10^{9}	8.334×10^{16}	-0.026×10^{16}
$\sigma(r)_{rel}/\epsilon$	524.745×10^{9}	541.175×10^{9}	1674.229×10^{16}	4.220×10^{16}

deviations $\sigma(r)_{rel} = (\Sigma_{i=1}^{N}((\mathcal{H}_{num,\,i} - \mathcal{H}_{exa,\,i}) - \bar{r}_{rel})^2)^{1/2}$. The precision ϵ of the computer arithmetic is defined as the smallest number such that $1 + |\epsilon| \neq 1$. In order to take into account the computer arithmetic, residuals and relative residuals were also calculated relative to ϵ. These numbers should be comparable among different computer platforms with different ϵ. Since $\epsilon = 0.222044604925031308\times10^{-15}$ for the present examples, all ϵ-relative residuals are unusual large numbers compared to absolute residuals.

A test on a sampled function was done for $f(x) = \cos^2(x)\exp(-0.1x)$ with $x_1 = 0$, $x_N = 2\pi$, and $N = 200$. The abscissa values were equidistantly spaced in this example, but HANKEL accepts also nonequidistant spacing. Since $f(x)$ is given at discrete points only, although required at many more points x_i by the integration process, $f(x)$ has to be interpolated at these x_i. HANKEL carries out this interpolation by itself. We compared HANKEL's results with results from HANKEL(AART) from Piessens [1982]. Table IV summarizes the comparison. HANKEL arrived at correct results for all arguments. HANKEL's results deviate not more than 1.5% from those of HANKEL(AART) for which HANKEL(AART) returns no error message.

$f(x) = \cos^2 x\exp(-0.1x)$ and $x = 0.0$ and $x = 2\pi$ and $\Delta x = \pi/100.0$

A major problem for all implementations of numerical algorithms is the detection of errors. Any program which realizes (3) has three major sources for errors: (a) the finite integration, (b) the infinite integration, and (c) the calculation of $J_\nu(x)$. A further complication arises from the different and varying severity of errors occurring in different parts of the algorithm. So

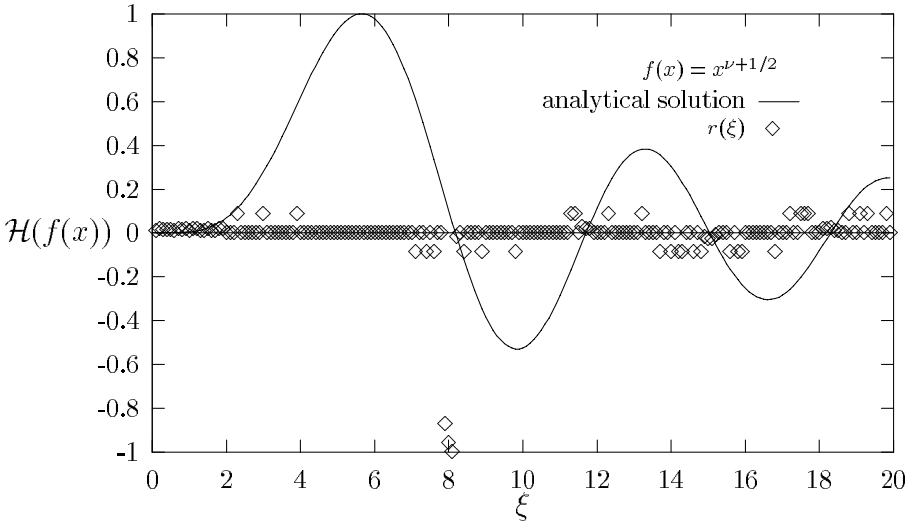

Fig. 1. Analytical solution (line) and residual $r(\xi)$ ($\diamond$) of the numerical calculation of $\mathcal{H}_\nu(\xi, f(x))$ for test function 1, $f(x) = x^{\nu+0.5}$, as a function of parameter ξ with $\nu = 3.5$. Both $\mathcal{H}_\nu(\xi, f(x))$ and $r(\xi)$ are normalized for plotting. Multiply $r(\xi)$ by a factor of 7.147×10^{-9} to have $r(\xi)$ in the same magnitude as the normalized $\mathcal{H}_\nu(\xi, f(x))$.

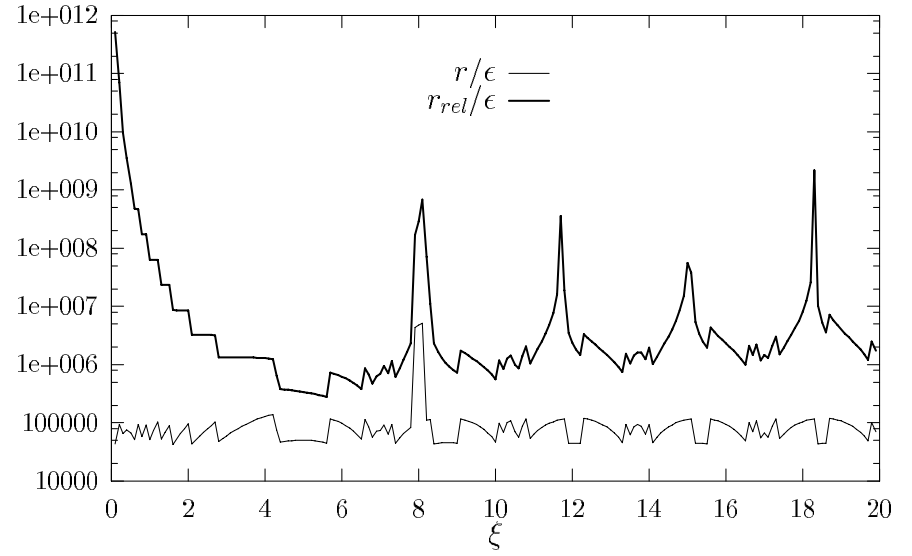

Fig. 2. Residual $r(\xi)/\epsilon$ and relative residual $r_{rel}(\xi)/\epsilon$ relative to arithmetic precision $\epsilon \approx 2.220 \times 10^{-16}$ for test function 1.

even if an error may have occurred, the final result still could be acceptable within some error bounds.[2] For several errors HANKEL will continue the

[2]For example, an integration subroutine may report that the requested accuracy has not been achieved within the preset maximal number of function evaluations, but the returned integral values may be correct.

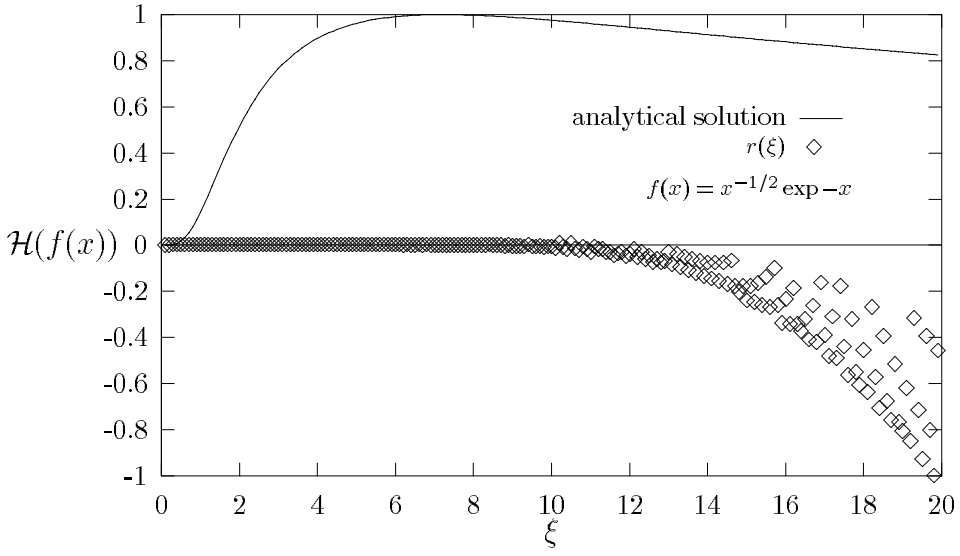

Fig. 3. Analytical solution (line) and residual $r(\xi)$ ($\diamond$) of the numerical calculation of $\mathcal{H}_\nu(\xi, f(x))$ for test function 2, $f(x) = x^{-0.5}\exp(-x)$, as a function of parameter ξ with $\nu = 3.5$. Both $\mathcal{H}_\nu(\xi, f(x))$ and $r(\xi)$ are normalized for plotting. Multiply $r(\xi)$ by a factor of 3.194×10^{-6} to have $r(\xi)$ in the same magnitude as the normalized $\mathcal{H}_\nu(\xi, f(x))$.

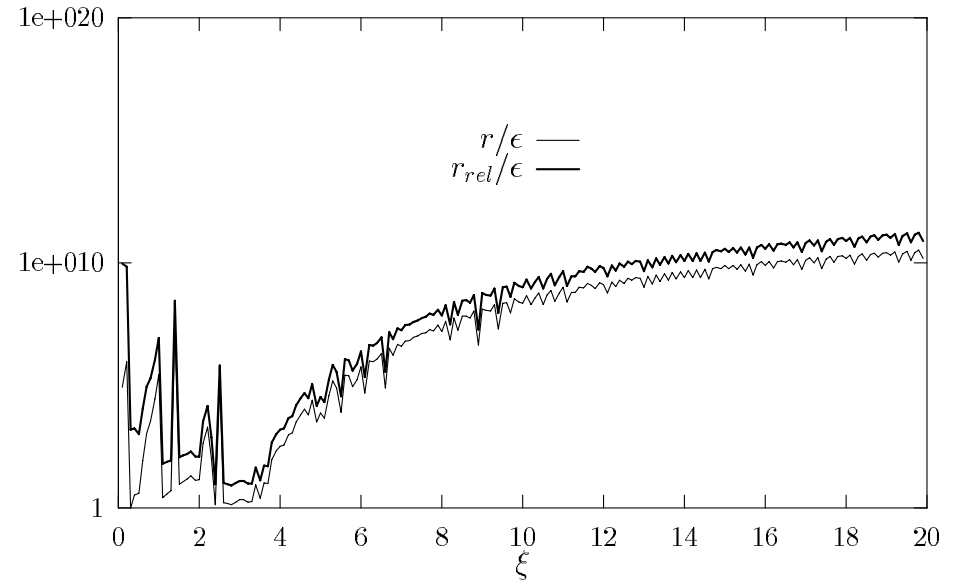

Fig. 4. Residual $r(\xi)/\epsilon$ and relative residual $r_{rel}(\xi)/\epsilon$ relative to arithmetic precision $\epsilon \approx 2.220 \times 10^{-16}$ for test function 2.

calculation and output an estimate for $\mathcal{H}_\nu(\xi, f(x))$, but in theses cases HANKEL cannot decide on the validity of the result. The most serious

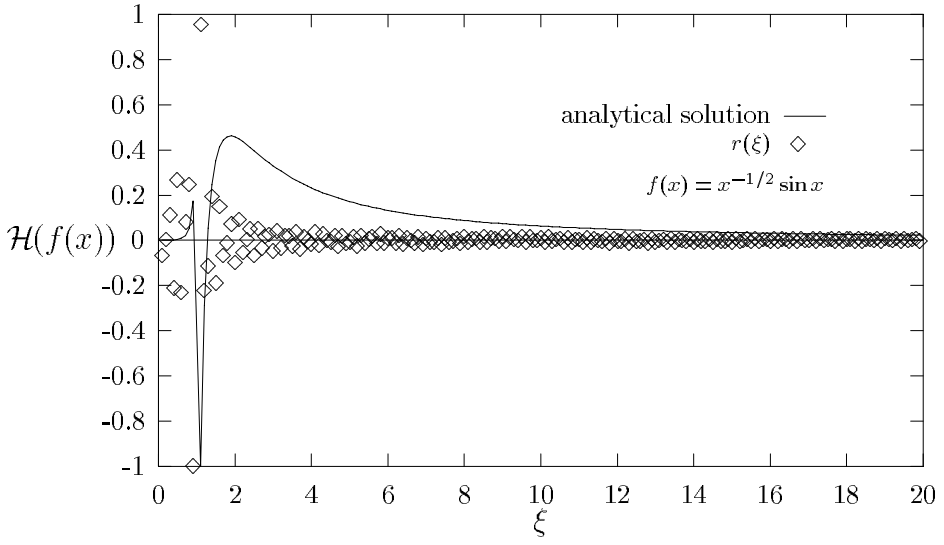

Fig. 5. Analytical solution (line) and residual $r(\xi)$ ($\diamond$) of the numerical calculation of $\mathcal{H}_\nu(\xi, f(x))$ for test function 3, $f(x) = x^{-0.5}\sin(x)$, as a function of parameter ξ with $\nu = 3.5$. Both $\mathcal{H}_\nu(\xi, f(x))$ and $r(\xi)$ are normalized for plotting. Multiply $r(\xi)$ by a factor of 0.443 to have $r(\xi)$ in the same magnitude as the normalized $\mathcal{H}_\nu(\xi, f(x))$.

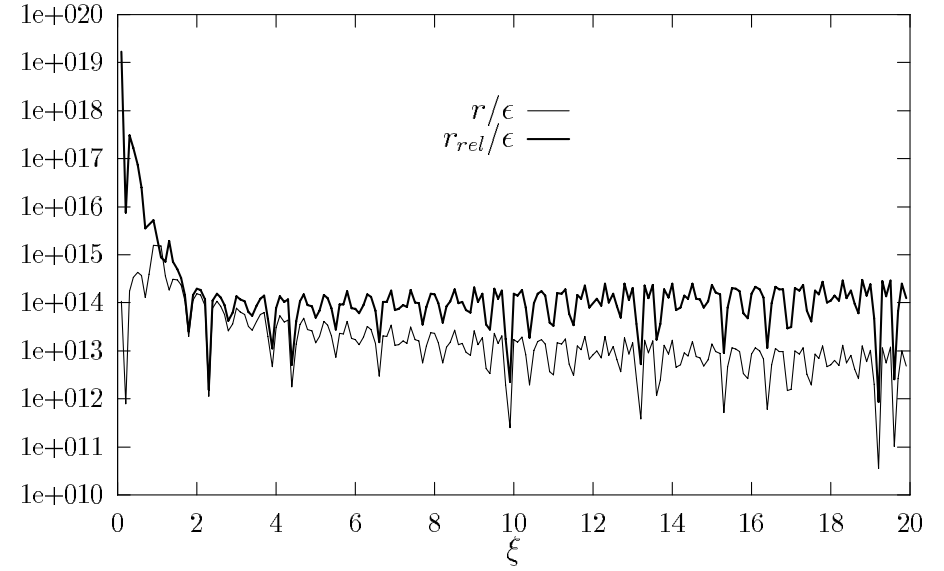

Fig. 6. Residual $r(\xi)/\epsilon$ and relative residual $r_{rel}(\xi)/\epsilon$ relative to arithmetic precision $\epsilon \approx 2.220 \times 10^{-16}$ for test function 3.

errors are those which are not detected by the program.[3] HANKEL can fail

[3]The integration subroutine will have difficulties at a singularity of the integrand. For example, if the integrand has a discontinuity, then the integration subroutine still may return a result, but an erroneous one.

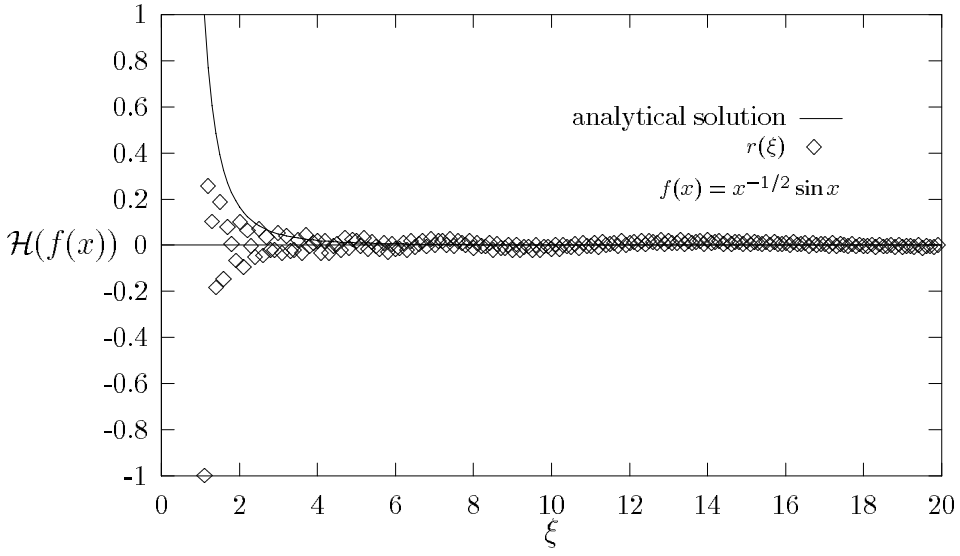

Fig. 7. Analytical solution (line) and residual $r(\xi)$ ($\diamond$) of the numerical calculation of $\mathcal{H}_\nu(\xi, f(x))$ for test function 4, $f(x) = x^{-0.5}J_{\nu-1}(x)$, as a function of parameter ξ with $\nu = 3.5$. Both $\mathcal{H}_\nu(\xi, f(x))$ and $r(\xi)$ are normalized for plotting. Multiply $r(\xi)$ by a factor of 0.037 to have $r(\xi)$ in the same magnitude as the normalized $\mathcal{H}_\nu(\xi, f(x))$.

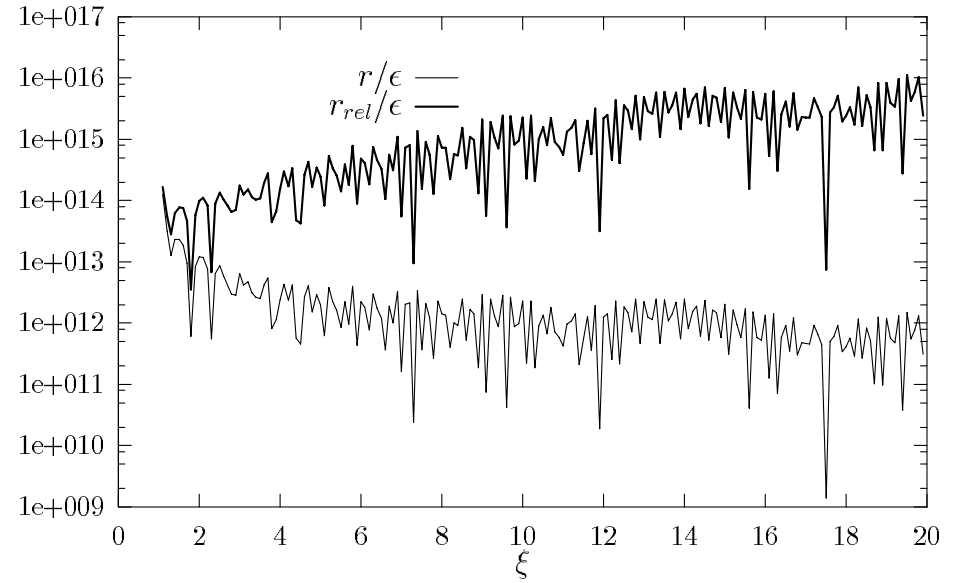

Fig. 8. Residual $r(\xi)/\epsilon$ and relative residual $r_{rel}(\xi)/\epsilon$ relative to arithmetic precision $\epsilon \approx 2.220 \times 10^{-16}$ for test function 4.

without any error message.[4]

[4]An example is provided by test function 2 for $x_l = x_{as} = 1000$.

Table IV. Hankel-Transform for $f(x) = \cos^2 x \exp(-0.1x)$ which Was Tabulated from $x = 0.0$ to $x = 2\pi$ with $\Delta x = \pi/100.0$. Column 3 gives the results obtained by HANKEL(AART) of R. Piessens and column 4 gives the error flag from HANKEL(AART) (0 = no error). Column 5 gives the results obtained by HANKEL presented in this work and column 6 gives the error flag from HANKEL (0 = no error)

Order ν	Parameter ξ	$\mathcal{H}(\xi, f(x))$ HANKEL(AART)	Error Flag	$\mathcal{H}(\xi, f(x))$ HANKEL	Error Flag
0	1	2.243887	1	0.175742	0
0	10	0.049471	2	0.049678	0
0	50	0.009589	0	0.010119	0
1	1	0.263275	3	0.368490	0
1	10	0.105199	2	0.107863	0
1	50	0.020902	0	0.028024	0
2	1	-1.769859	3	0.560231	0
2	10	0.138152	0	0.137769	0
2	50	0.028554	0	0.028007	0
3	1	0.365853	3	0.698227	0
3	10	0.157627	2	0.155844	0
3	50	0.034574	0	0.034760	0
4	1	3.150033	3	0.732722	0
4	10	0.166565	2	0.167138	0
4	50	0.039618	0	0.040218	0
5	1	1.966790	3	0.660152	0
5	10	0.166555	0	0.166589	0
5	50	0.043987	1	0.043631	0
6	1	-1.506114	3	0.603661	0
6	10	0.158907	2	0.158044	0
6	50	0.047840	0	0.047134	0
7	1	-2.145300	3	0.636334	0
7	10	0.144967	0	0.147134	0
7	50	0.051277	0	0.051873	0
8	1	1.290727	3	0.691614	0
8	10	0.126204	0	0.128319	0
8	50	0.054361	0	0.055248	0
9	1	4.615391	3	0.674457	0
9	10	0.104213	2	0.098799	0
9	50	0.057137	0	0.056248	0
10	1	3.411504	3	0.580114	0
10	10	0.080675	2	0.076618	0
10	50	0.059639	0	0.058462	0

5. CONCLUSIONS

HANKEL proved to give correct results for quite difficult (in particular oscillating) test functions $f(x)$, composed of algebraic, exponential, trigonometric, and special functions. HANKEL is expected to arrive at correct results for functions $f(x)$ which (a) fulfill $\int_0^\infty \|f(x)\| dx$, (b) are smooth on $[0, \infty]$, and (c) are nonoscillating for $x > x_l$.

ACKNOWLEDGMENTS

Many thanks to Professor A.E. Siegman, Stanford University, for valuable discussions, to Dr. W.-D. Kunze, Medizinische Universität Lübeck for

technical support, and to Dr. T.R. Hopkins, The University of Kent, for suggestions improving the Fortran code. I am indebted to Dr. Richard J. Hanson, ACM-TOMS, for his suggestions for improvement of the text and his patience as ACM-TOMS editor. I would like to express my gratitude to the reviewer for several suggestions and his labor. Since more than 10 years I am using the computing facilities of the computer center of the Universität Gesamthochschule Kassel, and therefore many thanks to P. Mann and R. Schwarz and all other members of the computer center. I would like to thank Professor H. Fuess, Technische Universität Darmstadt, for continuous support.

REFERENCES

ANDERSON, W. 1982. Fast Hankel transforms using related and lagged convolutions. *ACM Trans. Math. Softw. 8*, 4, 344–368.

COURANT, R. AND HILBERT, D. 1968. *Methoden der Mathematischen Physik*. Springer-Verlag, Berlin, Germany.

ERDELYI, A., MAGNUS, W., OBERHETTINGER, F., AND TRICOMI, F. 1954. *Tables of Integral Transforms*. McGraw-Hill, Inc., New York, NY.

KORN, A. AND KORN, T. 1968. *Mathematical Handbook for Scientists and Engineers*. McGraw-Hill, Inc., New York, NY.

LYNESS, J. AND HINES, G. 1986. ALGORITHM 639: To integrate some infinite oscillating tails. *ACM Trans. Math. Softw. 12*, 1 (Mar. 1986), 24–25.

PIESSENS, R. 1982. Automatic computation of Bessel function integrals. *Comput. Phys. Commun. 25*, 289.

SIEGMAN, A. 1980. Quasi fast Hankel transform. *Opt. Lett. 1*, 13–15.

SIKORSKY, K., STENGER, F., AND SCHWING, J. 1984. Algorithm 614: A Fortran subroutine for numerical integration in h^p. *ACM Trans. Math. Softw. 10*, 2 (June 1984), 152–160.

SNEDDON, I. 1955. Functional Analysis. In *Handbuch der Physik*, S. Flugge, Ed. Springer-Verlag, Berlin, Germany.

WATERLOO MAPLE. 1996. Maple. (Software). Waterloo Maple, Inc., Waterloo, Ontario, Canada. http://www.maplesoft.com

TALMAN, J. 1983. Lsfbtr: A subroutine for calculating spherical bessel transforms. *Comput. Phys. Commun. 30*, 93.

WHITTAKER, E. AND WATSON, G. 1978. *Modern Analysis*. Cambridge University Press, New York, NY.

Received: April 1998; revised: December 1998; accepted: February 1999