

FFTLog

A. J. S. Hamilton

JILA and Dept. Astrophysical & Planetary Sciences, Box 440, U. Colorado, Boulder CO 80309, USA;
Andrew.Hamilton@Colorado.EDU; <http://casa.colorado.edu/~ajsh>

13 March 2000

ABSTRACT

FFTLog is a set of fortran subroutines that compute the fast Fourier or Hankel (= Fourier-Bessel) transform of a periodic sequence of logarithmically spaced points. FFTLog can be regarded as a natural analogue to the standard Fast Fourier Transform (FFT), in the sense that, just as the normal FFT gives the exact (to machine precision) Fourier transform of a linearly spaced periodic sequence, so also FFTLog gives the exact Fourier or Hankel transform, of arbitrary order μ , of a logarithmically spaced periodic sequence. FFTLog shares with the normal FFT the problems of ringing (response to sudden steps) and aliasing (periodic folding of frequencies), but under appropriate circumstances FFTLog may approximate the results of a continuous Fourier or Hankel transform. The FFTLog algorithm was originally proposed by Talman (1978).

1 INTRODUCTION

This is a PostScript printable version of the webpage <http://casa.colorado.edu/~ajsh/FFTLog/>, from which the FFTLog fortran code may be downloaded. That webpage was adapted in turn from Appendix B of a paper by Hamilton (2000). If you wish to refer to the present document, please refer to Hamilton (2000) and/or to the website.

The FFTLog algorithm was originally proposed by Talman (1978).

Consider the continuous Hankel (= Fourier-Bessel) transform pair

$$\begin{aligned}\tilde{a}(k) &= \int_0^\infty a(r) (kr)^q J_\mu(kr) k dr, \\ a(r) &= \int_0^\infty \tilde{a}(k) (kr)^{-q} J_\mu(kr) r dk.\end{aligned}\quad (1)$$

If the substitution

$$a(r) = A(r) r^{-q} \quad \text{and} \quad \tilde{a}(k) = \tilde{A}(k) k^q \quad (2)$$

is made, then the Hankel transform pair (1) becomes equivalent to the transform pair

$$\begin{aligned}\tilde{A}(k) &= \int_0^\infty A(r) J_\mu(kr) k dr, \\ A(r) &= \int_0^\infty \tilde{A}(k) J_\mu(kr) r dk.\end{aligned}\quad (3)$$

Although the Hankel transform (1) with a power law bias $(kr)^{\pm q}$ is thus equivalent in the continuous case to the unbiased Hankel transform (3), the transforms are different when they are discretized and made periodic; for if $a(r)$ is periodic, then $A(r) = a(r) r^q$ is not periodic. FFTLog evaluates discrete Hankel transforms (1) with arbitrary power law bias.

Fourier sine and cosine transforms can be regarded as

special cases of Hankel transforms with $\mu = \pm 1/2$, since

$$\begin{aligned}J_{1/2}(x) &= (2/\pi x)^{1/2} \sin(x), \\ J_{-1/2}(x) &= (2/\pi x)^{1/2} \cos(x).\end{aligned}\quad (4)$$

As first noted by Siegman (1977), if the product kr in the Hankel transform is written as $e^{\ln k + \ln r}$, then the transform becomes a convolution integral in the integration variable $\ln r$ or $\ln k$. Convolution is equivalent to multiplication in the corresponding Fourier transform space. Thus the Hankel transform can be computed numerically by the algorithm: FFT \rightarrow multiply by a function \rightarrow FFT back. This is the idea behind a number of Fast Hankel Transform (FHT) algorithms (Candel 1981; Anderson 1982; Hansen 1985; Fanning 1996) including FFTLog (Talman 1978).

An advantage of FFTLog, emphasized by Talman (1978), is that the order μ of the Bessel function may be any arbitrary real number. In particular, FFTLog works for 1/2-integral μ , so includes the cases of Fourier sine and cosine transforms, and spherical Hankel transforms involving the spherical Bessel functions $j_\lambda(x) \equiv (\pi/2x)^{1/2} J_{\lambda+1/2}(x)$.

2 MOTIVATION AND EXAMPLE

FFTLog emerged from a problem in cosmology (Hamilton 2000). The problem required Fourier transforming a function that extended over many orders of magnitude, and was ‘smooth’ in logarithmic space. Actually, it was necessary to transform whole matrices of such functions, so a fast transform method was desirable.

In cosmology, fluctuations in the matter density of the Universe are thought to have been laid down during an inflationary epoch in the first few moments following the Big Bang (Turner 1997). Vacuum fluctuations in the field that drives inflation should produce a Gaussian distribution of

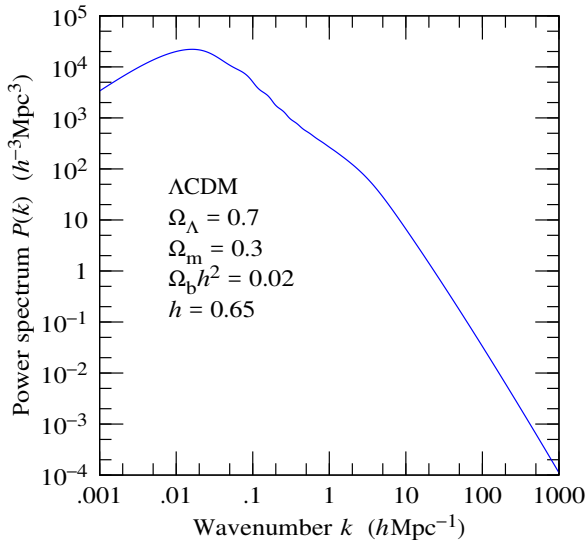


Figure 1. Cosmological power spectrum $P(k)$ of matter fluctuations predicted by the so-called Λ CDM model, a flat ($\Omega = 1$) Universe dominated by a cosmological constant ($\Omega_\Lambda = 0.7$), and Cold Dark Matter ($\Omega_m = 0.3$) including a sprinkling of baryons ($\Omega_b = 0.05$). The Λ CDM power spectrum was computed from the formulae of Eisenstein & Hu (1998), nonlinearly evolved according to the formula of Peacock & Dodds (1996). The spectrum is normalized to the amplitude of fluctuations observed by the COBE satellite.

density fluctuations with a near scale-invariant power spectrum $P(k) \propto k$. That primordial spectrum was processed prior to Recombination by the action of gravity modulated by the pressure of radiation. Following Recombination, when the Universe was about 300,000 years old, the matter power spectrum was further processed by nonlinear gravitational clustering, up to the present time.

The cosmological power spectrum $P(k)$, a function of wavenumber k , is the 3-dimensional Fourier transform of the cosmological correlation function $\xi(r)$, a function of spatial separation r . With the conventional normalization used by cosmologists,

$$P(k) = \int_0^\infty \xi(r) \frac{\sin(kr)}{kr} 4\pi r^2 dr, \quad (5)$$

$$\xi(r) = \int_0^\infty P(k) \frac{\sin(kr)}{kr} \frac{4\pi k^2 dk}{(2\pi)^3}.$$

Figures 1 and 2 show the cosmological power spectrum $P(k)$ of matter fluctuations, and the corresponding correlation function $\xi(r)$, predicted for a flat ($\Omega = 1$) Cold Dark Matter Universe dominated by a cosmological constant ($\Omega_\Lambda = 0.7$). The power spectrum here was computed from the formulae of Eisenstein & Hu (1998), nonlinearly evolved according to the formula of Peacock & Dodds (1996).

In this particular instance, FFTLog outperforms the normal FFT on all counts: it is more accurate, with fewer points, over a larger range, and it shows no signs of ringing. This does not mean that FFTLog is always better than FFT. Rather, FFTLog is well matched to the problem at hand: the cosmological power spectrum extends over many orders of magnitude in wavenumber k , and varies smoothly in $\ln k$.

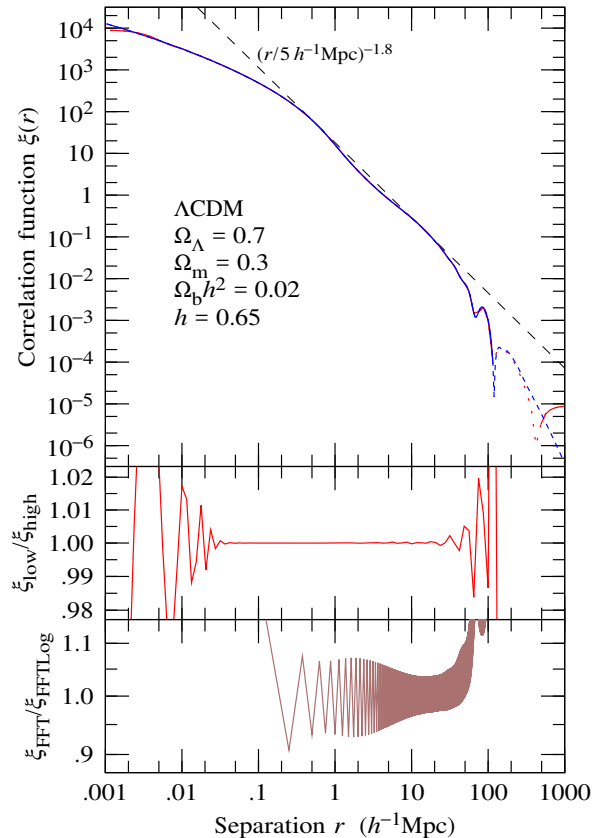


Figure 2. Cosmological correlation function $\xi(r)$ corresponding to the Λ CDM power spectrum shown in Figure 1.

The top panel shows the correlation functions computed with FFTLog at two different resolutions, plotted on top of each other: (red, low resolution) with 96 points over the range $r = 10^{-3}$ to $10^3 h^{-1}$ Mpc, and (blue, high resolution) with 768 points over the range $r = 10^{-6}$ to $10^6 h^{-1}$ Mpc. The lines are dashed where the correlation function is negative, at separations $r > 119 h^{-1}$ Mpc. The low and high resolution curves are almost indistinguishable except at $r \gtrsim 200 h^{-1}$ Mpc, where the low resolution curve goes to a constant, while the high resolution curve declines as a power law $\sim r^{-4}$. The disagreement is caused by aliasing (see §8) of small and large separations in the low resolution case. Aliasing is almost eliminated in the high resolution case because the range $r = 10^{-6}$ to $10^6 h^{-1}$ Mpc over which the transform was computed is much broader than the range plotted. The straight dashed line shows the canonical power law $(r/5 h^{-1} \text{Mpc})^{-1.8}$ for reference. Both low and high resolution cases used an unbiased ($q = 0$) transform, §7, and a low-ringing value of $k_0 r_0$, §6 (actually the choice of $k_0 r_0$ made little difference here).

The middle panel shows the ratio $\xi_{\text{low}}/\xi_{\text{high}}$ of the low to high resolution correlation functions.

The bottom panel of Figure 2 shows the ratio $\xi_{\text{FFT}}/\xi_{\text{FFTLog}}$ of the correlation function ξ_{FFT} computed with a normal FFT (sine transform) with 1023 points over the range $r = 0.125$ to $128 h^{-1}$ Mpc, to the high resolution correlation function ξ_{FFTLog} computed with FFTLog. The FFT'd correlation function ξ_{FFT} rings at the ± 5 percent level.

3 NORMAL DISCRETE FOURIER TRANSFORM

First, recall the essential properties of the standard discrete Fourier transform of a periodic sequence of linearly spaced points. Suppose that $a(r)$ is a continuous, in general complex-valued, function that is periodic with period R ,

$$a(r + R) = a(r) . \quad (6)$$

Without loss of generality, take the fundamental interval to be $[-R/2, R/2]$, centred at zero. Since $a(r)$ is periodic, its continuous Fourier transform contains only discrete Fourier modes $e^{2\pi imr/R}$ with integral wavenumbers m . Suppose further that the function $a(r)$ is 'smooth' in the specific sense that it is some linear combination only of the N lowest frequency Fourier modes, $m = 0, \pm 1, \dots, \pm[N/2]$, where $[N/2]$ denotes the largest integer greater than or equal to $N/2$,

$$a(r) = \sum'_m c_m e^{2\pi imr/R} \quad (7)$$

the outermost Fourier coefficients being equal, $c_{-N/2} = c_{N/2}$, in the case of even N . The primed sum in equation (7) signifies a sum over integral m from $-[N/2]$ to $[N/2]$, with the proviso that for even N the outermost elements of the sum receive only half weight:

$$\sum'_n x_n \equiv \sum_{n=-[N/2]}^{[N/2]} w_n x_n \quad (8)$$

with $w_n = 1$ except that $w_{-N/2} = w_{N/2} = 1/2$ if N is even.

The sampling theorem (e.g. Press et al. 1986 §12.1) asserts that, given a function $a(r)$ satisfying equation (7), the Fourier coefficients c_m can be expressed in terms of the values $a_n \equiv a(r_n)$ of the function $a(r)$ at the N discrete points $r_n = nR/N$ for $n = 0, \pm 1, \dots, \pm[N/2]$. For even N , the periodicity of $a(r)$ ensures that $a_{-N/2} = a_{N/2}$. Specifically, the sampling theorem asserts that the Fourier coefficients in the expansion (7) satisfy

$$c_m = \frac{1}{N} \sum'_n a_n e^{-2\pi imn/N} \quad (9)$$

the discrete points a_n themselves satisfying

$$a_n = \sum'_m c_m e^{2\pi imn/N} \quad (10)$$

in accordance with equation (7).

Equations (9) and (10) constitute a discrete Fourier transform pair relating two periodic, linearly spaced sequences a_n and c_m of length N . The standard FFT evaluates the discrete Fourier transform exactly (that is, to machine precision).

4 DISCRETE HANKEL TRANSFORM

Now suppose that the function $a(r)$, instead of being periodic in ordinary space r , is periodic in logarithmic space $\ln r$, with logarithmic period L ,

$$a(re^L) = a(r) . \quad (11)$$

Take the fundamental interval to be $[\ln r_0 - L/2, \ln r_0 + L/2]$, centred at $\ln r_0$. As in §3, the periodicity of $a(r)$ implies that

its Fourier transform with respect to $\ln r$ contains only discrete Fourier modes $e^{2\pi im \ln(r/r_0)/L}$ with integral wavenumbers m . Suppose further, as in §3 eq. (7), that $a(r)$ contains only the N lowest frequency Fourier modes

$$a(r) = \sum'_m c_m e^{2\pi im \ln(r/r_0)/L} \quad (12)$$

with $c_{-N/2} = c_{N/2}$ for even N . The sampling theorem asserts that the Fourier coefficients c_m are given by

$$c_m = \frac{1}{N} \sum'_n a_n e^{-2\pi imn/N} \quad (13)$$

where $a_n \equiv a(r_n)$ are the values of the function $a(r)$ at the N discrete points $r_n = r_0 e^{nL/N}$ for $n = 0, \pm 1, \dots, \pm[N/2]$,

$$a_n = \sum'_m c_m e^{2\pi imn/N} . \quad (14)$$

The continuous Hankel transform $\tilde{a}(k)$, equation (1), of a function $a(r)$ of the form (12) is

$$\tilde{a}(k) = \sum'_m c_m \int_0^\infty e^{2\pi im \ln(r/r_0)/L} (kr)^q J_\mu(kr) k dr . \quad (15)$$

The integrals on the right hand side of equation (15) can be done analytically, in terms of

$$U_\mu(x) \equiv \int_0^\infty t^x J_\mu(t) dt = 2^x \frac{\Gamma[(\mu + 1 + x)/2]}{\Gamma[(\mu + 1 - x)/2]} \quad (16)$$

where $\Gamma(z)$ is the usual Gamma-function. Thus equation (15) reduces to

$$\tilde{a}(k) = \sum'_m c_m u_m e^{-2\pi im \ln(k/k_0)/L} \quad (17)$$

where u_m is

$$u_m(\mu, q) \equiv (k_0 r_0)^{-2\pi im/L} U_\mu\left(q + \frac{2\pi im}{L}\right) . \quad (18)$$

Notice that $u_m^* = u_{-m}$, which ensures that $\tilde{a}(k)$ is real if $a(r)$ is real. Equation (17) gives the (exact) continuous Hankel transform $\tilde{a}(k)$ of a function $a(r)$ of the form (7). Like $a(r)$, the Hankel transform $\tilde{a}(k)$ is periodic in logarithmic space $\ln k$, with period L . The fundamental interval is $[\ln k_0 - L/2, \ln k_0 + L/2]$, centred at $\ln k_0$, which may be chosen arbitrarily (but see §6 below).

The sampling theorem requires that $u_{-N/2} = u_{N/2}$ for even N , which is not necessarily satisfied by equation (18). However, at the discrete points $k_n = k_0 e^{nL/N}$ considered by the sampling theorem, the contributions at $m = \pm N/2$ to the sum on the right hand side of equation (17) are $(-)^n c_{N/2} (u_{N/2} + u_{N/2}^*)/2 = (-)^n c_{N/2} \text{Re } u_{N/2}$. Thus the equality (17) remains true at the discrete points k_n if $u_{\pm N/2}$ are replaced by their real parts,

$$u_{\pm N/2} \rightarrow \text{Re } u_{N/2} . \quad (19)$$

With the replacement (19), the sampling theorem asserts that the coefficients $c_m u_m$ in the sum (17) are determined by the values $\tilde{a}_n \equiv \tilde{a}(k_n)$ of the Hankel transform at the N discrete points $k_n = k_0 e^{nL/N}$ for $n = 0, \pm 1, \dots, \pm[N/2]$

$$c_m u_m = \frac{1}{N} \sum'_n \tilde{a}_n e^{2\pi imn/N} \quad (20)$$

$$\tilde{a}_n = \sum'_m c_m u_m e^{-2\pi i m n / N} . \quad (21)$$

Putting together equations (13), (14), (20) and (21) yields the discrete Hankel transform pair

$$\tilde{a}_n = \sum'_m a_m v_{m+n}^+(\mu, q) \quad (22)$$

$$a_m = \sum'_n \tilde{a}_n v_{m+n}^-(\mu, q) \quad (23)$$

in which the forward discrete Hankel mode $v_n^+(\mu, q)$ is the discrete Fourier transform of $u_m(\mu, q)$ given by equations (18) and (19),

$$v_n^+(\mu, q) = \frac{1}{N} \sum'_m u_m(\mu, q) e^{-2\pi i m n / N} \quad (24)$$

while the inverse discrete Hankel mode $v_n^-(\mu, q)$ is the discrete Fourier transform of the reciprocal $1/u_{-m}(\mu, q)$,

$$v_n^-(\mu, q) = \frac{1}{N} \sum'_m \frac{1}{u_{-m}(\mu, q)} e^{-2\pi i m n / N} . \quad (25)$$

The Hankel transform matrices $v_{m+n}^+(\mu, q)$ and $v_{m+n}^-(\mu, q)$ are mutually inverse

$$\sum'_l v_{m+l}^+(\mu, q) v_{l+n}^-(\mu, q) = \delta_{mn} \quad (26)$$

where δ_{mn} denotes the Kronecker delta. The forward and inverse Hankel modes have the interesting property of being self-similar; that is, Hankel modes $v_{m+n}^+(\mu, q)$ [or $v_{m+n}^-(\mu, q)$] with different indices m consist of the same periodic sequence $v_n^+(\mu, q)$ [or $v_n^-(\mu, q)$] cyclically shifted by m notches.

FFTLog evaluates the forward and inverse discrete Hankel transforms given by equations (22), (23), exactly (to machine precision).

The reciprocal $1/u_{-m}(\mu, q)$ in equation (25) is equal to $u_m(\mu, -q)$, according to equations (16) and (18),

$$\frac{1}{u_{-m}(\mu, q)} = u_m(\mu, -q) \quad (m \neq N/2) \quad (27)$$

except in the case $m = \pm N/2$ for even N , when the replacement (19) generally invalidates equation (27). However, in the special case where $u_{\pm N/2}$ are already real, then equation (19) leaves $u_{\pm N/2}$ unchanged, and equation (27) remains valid also at $m = \pm N/2$. This special case is of particular interest, and is discussed further in §6 below.

In the continuous case, the inverse Hankel transform is equal to the forward transform with $q \rightarrow -q$, equations (1). In the discrete case this remains true for odd N , but it is not generally true for even N (the usual choice) except in the important special case discussed in §6.

In the general discrete case (i.e. if the condition [28] in §6 is not satisfied), the inverse discrete Hankel mode $v_n^-(\mu, q)$, equation (25), differs from the forward Hankel mode $v_n^+(\mu, -q)$, equation (24), only for even N and only in the coefficient of the highest frequency Fourier component, $1/u_{-m}(\mu, q)$ versus $u_m(\mu, -q)$ for $m = \pm N/2$. To the extent that the highest frequency Fourier coefficient $c_{\pm N/2}$ of a sequence a_n is small, the difference between its inverse discrete Hankel transform and its forward transform with $q \rightarrow -q$ should be small.

It is possible for the inverse discrete Hankel transform to be singular, if $u_{\pm N/2}$ is purely imaginary, so that its real part vanishes, making $v_n^-(\mu, q)$ singular. As discussed in §6, this singularity can be avoided by choosing a low-ringing value of $k_0 r_0$, equation (30).

The forward (inverse) discrete Hankel transforms are also singular at special values of μ and q , namely where $\mu + 1 + q$ (or $\mu + 1 - q$ in the inverse case) vanishes, because $u_0(\mu, q) = U_\mu(q)$ is singular at these points. This singularity reflects a real singularity in the corresponding continuous Hankel transform (unlike the singularity of the previous paragraph, which is an avoidable artefact of discreteness). The singularity in u_0 leads to an additive infinite constant in the discrete Hankel transform. In physical problems this additive infinite constant may somehow cancel out (for example, in the difference between two Hankel transforms). FFTLog's strategy in these singular cases is to evaluate the discrete Hankel transform with the infinite constant set to zero, and to issue a warning.

5 FFTLOG ALGORITHM

The FFTLog algorithm for taking the discrete Hankel transform, equation (22), of a sequence a_n of N logarithmically spaced points is:

- FFT a_n to obtain the Fourier coefficients c_m , equation (13);
- multiply by u_m given by equations (18) and (19) to obtain $c_m u_m$;
- FFT $c_m u_m$ back to obtain the discrete Hankel transform \tilde{a}_n , equation (21).

A variant of the algorithm is to sandwich the above operations with power law biasing and unbiasing operations. For example, one way to take the unbiased continuous Hankel transform $\tilde{A}(k)$ of a function $A(r)$, equation (3), is to bias $A(r)$ and $\tilde{A}(k)$ with power laws, equation (2), and take a biased Hankel transform, equation (1). The discrete equivalent of this is:

- Bias A_n with a power law to obtain $a_n = A_n r_n^{-q}$, equation (2);
- FFT a_n to obtain the Fourier coefficients c_m , equation (13);
- multiply by u_m given by equations (18) and (19) to obtain $c_m u_m$;
- FFT $c_m u_m$ back to obtain the discrete Hankel transform \tilde{a}_n , equation (21);
- Unbias \tilde{a}_n with a power law to obtain $\tilde{A}_n = \tilde{a}_n k_n^{-q}$, equation (2).

Although in the continuous limit the result would be identical to an unbiased Hankel transform, in the discrete case the result differs. With a simple unbiased discrete Hankel transform, it is the sequence A_n that is taken to be periodic, whereas in the algorithm above it is not A_n but rather a_n that is periodic.

The inverse discrete Hankel transform is accomplished by the same series of steps, except that c_m is divided instead of multiplied by u_m .

The FFTLog code is built on top of the NCAR suite of FFT routines (Swarztrauber 1979), and a modified version

of an implementation of the complex Gamma-function from the gamfer package by Oura (1996).

FFTLog includes driver routines for the specific cases of the Fourier sine and cosine transforms.

6 LOW-RINGING CONDITION ON $k_0 r_0$

The central values $\ln r_0$ and $\ln k_0$ of the periodic intervals in $\ln r$ and $\ln k$ may be chosen arbitrarily. However, ringing of the discrete Hankel transform may be reduced, for either even or odd N , if the product $k_0 r_0$ is chosen in such a way that the boundary points of the sequence u_m , equation (18), are equal

$$u_{-N/2} = u_{N/2} . \quad (28)$$

Recall that the general procedure, for even N , was to replace $u_{\pm N/2}$ by their real part, equation (19). The condition (28) requires that $u_{\pm N/2}$ are already real. The condition (28) reduces ringing because it makes the periodic sequence u_m fold smoothly across the period boundary at $m = \pm N/2$.

In addition to reducing ringing, the condition (28) means that equation (27) remains true also at $m = \pm N/2$, so is true for all m . In this case the inverse Hankel mode $v_n^-(\mu, q)$, equation (25), is equal to the forward Hankel mode $v_n^+(\mu, -q)$ with q of the opposite sign

$$v_n^-(\mu, q) = v_n^+(\mu, -q) = \frac{1}{N} \sum'_m u_m(\mu, -q) e^{-2\pi i m n / N} . \quad (29)$$

In other words, if condition (28) is satisfied, then the inverse discrete Hankel transform equals the forward discrete Hankel transform with $q \rightarrow -q$. This is like the continuous Hankel transform, equations (1), where the inverse transform equals the forward transform with $q \rightarrow -q$.

The periodicity condition (28) on $u_{\pm N/2}$ translates, for real μ and q , into a condition on $k_0 r_0$

$$\ln(k_0 r_0) = \frac{L}{N} \left\{ \frac{1}{\pi} \text{Arg} \left[U_\mu \left(q + \frac{\pi i N}{L} \right) \right] + \text{integer} \right\} \quad (30)$$

where $\text{Arg} z \equiv \text{Im} \ln z$ denotes the argument of a complex number, and integer is any integer. In other words, to reduce ringing, it may help to choose $k_0 r_0$ so as to satisfy the condition (30). This is not too much of a restriction, since L/N is the logarithmic spacing between points (= one notch), so the low-ringing condition (30) allows $k_0 r_0$ to be chosen to lie within half a notch [= $L/(2N)$] of whatever number one chooses, for example within half a notch of $k_0 r_0 = 1$.

The low-ringing condition (30) is a condition on the phasing of the discrete points rn and kn at which the discrete Hankel transform is specified. The condition is analogous to, albeit more complicated than, the condition on the usual FFT that discrete frequencies be phased so that their wavenumbers are integers, equation (7).

FFTLog can be set to use automatically the low-ringing value of $k_0 r_0$ nearest to any input value of $k_0 r_0$.

Note that the low-ringing value of $k_0 r_0$ from condition (30) differs for different μ , q , and L/N . For example, the sine transform ($\mu = 1/2$) and cosine transform ($\mu = -1/2$) have different low-ringing values of $k_0 r_0$.

How else does the choice of $k_0 r_0$ affect the Hankel transform? Increasing the value of $\ln(k_0 r_0)$ by one notch L/N

cyclically shifts the discrete Hankel transform \tilde{a}_n , equation (21), by one notch to the left, $\tilde{a}_n \rightarrow \tilde{a}_{n-1}$. In other words, changing $\ln(k_0 r_0)$ by an integral number of notches shifts the origin of the transform, but leaves the transform otherwise unchanged, as might have been expected.

In practice, since in most cases one is probably using the discrete Hankel transform as an approximation to the continuous transform, one would probably want to use $k_0 r_0 \approx 1$ (or 2, or π , according to taste).

7 UNITARY UNBIASED HANKEL TRANSFORM

The discrete Hankel transform with both low-ringing $k_0 r_0$ and no power law bias, $q = 0$, is of particular interest because it is unitary, like the Fourier transform. Indeed, being also real, the low-ringing unbiased Hankel transform is orthogonal, i.e. self-inverse, like the Fourier sine and cosine transforms. This is like the continuous unbiased ($q = 0$) Hankel transform, equations (1), which is self-inverse.

The discrete Hankel modes $v_{m+n}(\mu, 0) = v_{m+n}^+(\mu, 0) = v_{m+n}^-(\mu, 0)$ in the low-ringing unbiased ($q = 0$) case are periodic, orthonormal, and self-similar, equation (26),

$$\sum'_l v_{m+l}(\mu, 0) v_{l+n}(\mu, 0) = \delta_{mn} . \quad (31)$$

Like any orthogonal transformation, the low-ringing unbiased ($q = 0$) Hankel transform commutes with the operations of matrix multiplication, inversion, and diagonalization (for non-low-ringing or biased Hankel transforms, $q \neq 0$, the operations do not commute). That is, the Hankel transform of the product of two matrices is equal to the product of their Hankel transforms, and so on.

All else being equal (which it may not be), given a choice between applying an unbiased ($q = 0$) or biased ($q \neq 0$) Hankel transform, and between a low-ringing $k_0 r_0$, equation (30), or otherwise, one would be inclined to choose the low-ringing unbiased transform, because of its orthogonality property.

8 RINGING AND ALIASING

FFTLog suffers from the same problems of ringing (response to sudden steps) and aliasing (periodic folding of frequencies) as the normal FFT.

Usually one is interested in the discrete Fourier or Hankel transform not for its own sake, but rather as an approximation to the continuous transform. The usual procedure would be to apply the discrete transform to a finite segment of the function $a(r)$ to be transformed. For FFTLog, the procedure can be regarded as involving two steps: truncating the function to a finite logarithmic interval, which causes ringing of the transform; followed by periodic replication of the function in logarithmic space, which causes aliasing.

Figure 3 illustrates these steps for the unbiased ($q = 0$) Hankel transform, equation (1), of order $\mu = -1/2$ of a function that is Gaussian in the log

$$a(r) = \exp[-(\ln r)^2/2] . \quad (32)$$

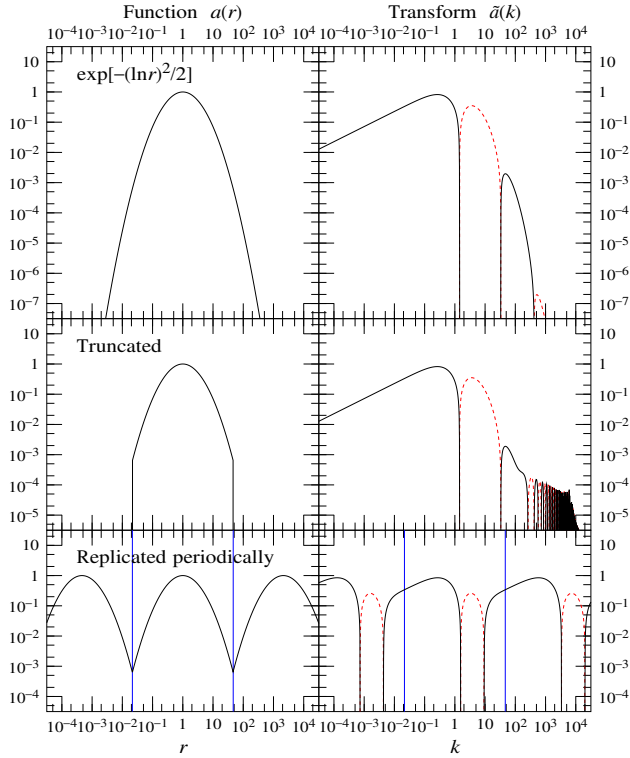


Figure 3. Illustrating the ringing and aliasing that occurs when the continuous Hankel transform of a function is approximated by the discrete Hankel transform of a finite segment of the function. Lines are dashed where values are negative. The function $a(r)$ is shown to the left, and its corresponding Hankel transform $\tilde{a}(k)$ to the right. The panels from top to bottom are: (top) the original function $a(r)$ and its Hankel transform $\tilde{a}(k)$; (middle) the truncated function $a(r)$ and its Hankel transform $\tilde{a}(k)$, which rings at high frequencies k ; and (bottom) the truncated, periodically replicated function $a(r)$ and its corresponding periodically replicated Hankel transform $\tilde{a}(k)$, which is aliased. Vertical lines demarcate periodic intervals.

Truncation of the function $a(r)$ leads to ringing of its transform $\tilde{a}(k)$ at high frequencies k , as seen in the middle right panel of Figure 3. The oscillations at large k are actually uniformly spaced in k , but appear bunched up because of the logarithmic plotting.

Periodic replication means taking a sum of copies shifted by integral periods. From the definition (1) of the continuous Hankel transform, it can be seen that periodically replicating a function $a(r)$ in logarithmic space $\ln r$ and then taking its continuous Hankel transform is equivalent to Hankel transforming the function $a(r)$ and then periodically replicating the transform $\tilde{a}(k)$ in $\ln k$. But truncating a function does not truncate its transform. So whereas a truncated, periodically replicated function $a(r)$ contains contributions from only one period at each point r , the periodically replicated transform contains overlapping contributions from many periods at each point k . This is aliasing. In Figure 3 aliasing is visible as an enhancement of the periodically replicated transform $\tilde{a}(k)$ on the high k side of the periodic interval.

Ringing and aliasing can be reduced by taking suitable precautions.

The ringing that results from taking the discrete transform of a finite segment of a function can be reduced by arranging that the function folds smoothly from large to small scales. It may help to bias the function with a power law before transforming it, as in the second algorithm in §5. It may also help to use a low-ringing value of $k_0 r_0$, §6.

Aliasing can be reduced by enlarging the periodic interval. Aliasing can be eliminated (to machine precision) if the interval can be enlarged to the point where the transform $\tilde{a}(k)$ goes sensibly to zero at the boundaries of the period. Note that it is not sufficient to enlarge the interval to the point where $a(r)$ is sensibly zero at the period boundaries: what is important is that the transform $\tilde{a}(k)$ goes to zero at the boundaries.

9 TROUBLESHOOTING

FFTLLog does not work well with my function. What should I do?

- Diagnose the problem. Is there ringing and aliasing? Read §8. Is your function ‘smooth’ (contains only low frequencies) in logarithmic space? If not, then FFTLog may not be appropriate to your problem.
- Use a low-ringing value of $k_0 r_0$, §6.
- Experiment with different values of the bias index q .
- Enlarge the periodic interval over which you specify your function. Extrapolate your function sensibly: padding with zeros may not be enough.
- Increase the resolution, by reducing the logarithmic spacing of points. If your function is adequately ‘smooth’, then increasing the resolution should, eventually, have no effect. If continuing to increase the resolution continues to have an effect, then your function is not ‘smooth’.
- Use another code.

REFERENCES

- Anderson W. L., 1982, ACM-Trans. Math. Software, 8, no. 4, 344. Hankel transform code at <http://www.acm.org/calgo/contents/588.gz>
- Candel S. M. 1981, IEEE Trans. ASSP, 29, 963
- Fanning G., 1996, Hankel transform code at <http://www.stanford.edu/~fanning/hankel/hankel.html>
- Eisenstein D. J., Hu W., 1998, ApJ, 496, 605
- Hamilton A. J. S., 2000, MNRAS, 312, 257 <http://xxx.lanl.gov/abs/astro-ph/9905191>
- Hansen E. W., 1985, IEEE Trans. ASSP, 33, 666
- Ooura T., 1996, <http://momonga.t.u-tokyo.ac.jp/~ooura/gamerf.html>
- Peacock J. A., Dodds S. J., 1996, MNRAS, 280, L19
- Siegmán A. E., 1977, Quasi fast Hankel transform, Opt. Lett., 1, 13
- Swarztrauber P. N., 1979, NCAR FFT routines at <http://www.netlib.org/bihar/>
- Talman J. D., 1978, J. Comp. Phys., 29, 35
- Turner M. S., 1997, in Schramm D. N., Galeotti P., eds, Generation of Cosmological Large-Scale Structure, Kluwer, Dordrecht, p. 153 <http://xxx.lanl.gov/abs/astro-ph/9704062>