Computational Methods for Astrophysics: Fourier Transforms

John T. Whelan

(filling in for Joshua Faber)

April 27, 2011

Outline: Fourier Transforms

- Continuous Fourier Transforms
- Discrete Fourier Transforms
- Sampling and Aliasing
- The Fast Fourier Transform

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt$$

$$h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i,

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{i2\pi f(t-t_0)}\, dt$$

$$h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{-i2\pi f(t-t_0)}\, df$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i, f vs ω,

$$h_\omega = \frac{1}{2\pi} \int_{-\infty}^{\infty} h(t)\, e^{-i\omega(t-t_0)}\, dt$$

$$h(t) = \int_{-\infty}^{\infty} h_\omega\, e^{i\omega(t-t_0)}\, d\omega$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i, f vs ω, normalization,

$$h_\omega = \int_{-\infty}^{\infty} h(t)\, e^{-i\omega(t-t_0)}\, dt$$

$$h(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} h_\omega\, e^{i\omega(t-t_0)}\, d\omega$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i, f vs ω, normalization,

$$h_\omega = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} h(t)\, e^{-i\omega(t-t_0)}\, dt$$

$$h(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} h_\omega\, e^{i\omega(t-t_0)}\, d\omega$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i, f vs ω, normalization, time origin

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f t}\, dt$$

$$h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f t}\, df$$

Continuous Fourier Transforms

- Fourier transform $\widetilde{h}(f)$ of time series $h(t)$:

$$\widetilde{h}(f) = \int_{-\infty}^{\infty} h(t)\, e^{-i2\pi f(t-t_0)}\, dt \Longleftrightarrow h(t) = \int_{-\infty}^{\infty} \widetilde{h}(f)\, e^{i2\pi f(t-t_0)}\, df$$

- Alternate conventions: sign of i, f vs ω, normalization, time origin
- If $h(t)$ is real, $\widetilde{h}(-f) = \widetilde{h}^*(f)$
- Convolution theorem:

$$g(t) = \int_{-\infty}^{\infty} A(t-t')\, h(t')\, dt' \Longleftrightarrow \widetilde{g}(f) = \widetilde{A}(f)\widetilde{h}(f)$$

- Can Fourier transform in space as well as time; e.g.,

$$\widetilde{h}(f_x, f_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y) e^{i2\pi(f_x x + f_y y)}\, dx\, dy$$

A Few Words About Convolution

$$g(t) = \int_{-\infty}^{\infty} A(t - t')\, h(t')\, dt'$$

- Can be written in different-looking but equivalent ways, e.g.

$$g(t) = \int_{-\infty}^{\infty} A(\tau)\, h(t - \tau)\, d\tau$$

- Because $A(\tau)$ is a function of a time difference, its Fourier transform is always defined w/zero time origin:

$$\widetilde{A}(f) = \int_{-\infty}^{\infty} A(\tau)\, e^{-i2\pi f \tau}\, d\tau$$

- Arises naturally in astrophysical science & technology: superposition of impulse responses, point-spread fcn of imaging device, linear transfer function, ...

Applications of the Fourier Transform

- Multiply Fourier transforms to calculate convolution
- Differential equations:

$$\frac{d^n}{dt^n}h(t) \Leftrightarrow (i2\pi f)^n \widetilde{h}(f)$$

- Spectral analysis
- Matched/optimal filtering when spectral properties of signal and/or noise are known

Discrete Fourier Transforms

- DFT of N-point sequence $\{h_j | j = 0, \ldots, N-1\}$:

$$\widehat{h}_k = \sum_{j=0}^{N-1} h_j\, e^{-i2\pi jk/N} \qquad \Longleftrightarrow \qquad h_j = \frac{1}{N} \sum_{k=0}^{N-1} \widehat{h}_k\, e^{i2\pi jk/N}$$

- Corresponds to CFT of discretized data:

$$h_j = h(t_0 + j\,\delta t) \qquad \Longleftrightarrow \qquad \widehat{h}_k\,\delta t \sim \widetilde{h}\left(\frac{k}{N\,\delta t}\right)$$

- Again, different conventions (mostly $\pm i$ & where to put $\frac{1}{N}$); always wise to check your FT package's documentation
- Note by construction $\widehat{h}_{N+k} = \widehat{h}_k$; means e.g., 8-pt FT packed

$\widehat{h}_0$	$\widehat{h}_1$	$\widehat{h}_2$	$\widehat{h}_3$	$\widehat{h}_{-4}$	$\widehat{h}_{-3}$	$\widehat{h}_{-2}$	$\widehat{h}_{-1}$

Sometimes use `fftshift` fcn to swap halves of array & get

$\widehat{h}_{-4}$	$\widehat{h}_{-3}$	$\widehat{h}_{-2}$	$\widehat{h}_{-1}$	$\widehat{h}_0$	$\widehat{h}_1$	$\widehat{h}_2$	$\widehat{h}_3$

DFTs of Real and Complex Data

- For complex data, N cmplx data points $\{h_i | i = 0, \ldots, N-1\}$ $\Longleftrightarrow$ N independent complex Fourier components $\{\widehat{h}_k | k = 0, \ldots, N-1\}$ or $\{\widehat{h}_k | k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1\}$

h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7

$\widehat{h}_0$	$\widehat{h}_1$	$\widehat{h}_2$	$\widehat{h}_3$	$\widehat{h}_{-4}$	$\widehat{h}_{-3}$	$\widehat{h}_{-2}$	$\widehat{h}_{-1}$

- For real data, symmetry of FT means $\widehat{h}_k = \widehat{h}^*_{-k} = \widehat{h}^*_{N-k}$ N real data points $\{h_i | i = 0, \ldots, N-1\}$ $\Longleftrightarrow$
  - 2 real Fourier cmpts $\widehat{h}_0$ & $\widehat{h}_{N/2}$
  - $\frac{N}{2} - 1$ indep cmplx Fourier cmpts $\{\widehat{h}_k | k = 1, \ldots, \frac{N}{2} - 1\}$

h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7

$\widehat{h}_0$	$\widehat{h}_1$	$\widehat{h}_2$	$\widehat{h}_3$	$\widehat{h}_4$	$[\widehat{h}^*_3]$	$[\widehat{h}^*_2]$	$[\widehat{h}^*_1]$

- These assume N even; modification for odd N straightforward

Time and Frequency Resolution

- Recall correspondence between continuous & discrete FT:

$$h_j = h(t_0 + j\,\delta t) \qquad \Longleftrightarrow \qquad \widehat{h}_k\,\delta t \sim \widetilde{h}\,(f_k)$$

where

$$f_k = k\,\delta f \qquad \delta f = \frac{1}{N\,\delta t} \equiv \frac{1}{T}$$

Time and Frequency Resolution

- Recall correspondence between continuous & discrete FT:

$$h_j = h(t_0 + j\,\delta t) \qquad \Longleftrightarrow \qquad \widehat{h}_k\,\delta t \sim \widetilde{h}\,(f_k)$$

where

$$f_k = k\,\delta f \qquad \delta f = \frac{1}{N\,\delta t} \equiv \frac{1}{T}$$

- Aside: why not factor $\frac{1}{N} = \delta t\,\delta f$ & define DFT as $\widetilde{h}_k \equiv \widehat{h}_k\,\delta t$

$$\widetilde{h}_k = \delta t \sum_{j=0}^{N-1} h_j\, e^{-i2\pi jk/N} \qquad \Longleftrightarrow \qquad h_j = \delta f \sum_{k=-N/2}^{N/2-1} \widetilde{h}_k\, e^{i2\pi jk/N} \quad ?$$

Time and Frequency Resolution

- Recall correspondence between continuous & discrete FT:

$$h_j = h(t_0 + j\,\delta t) \qquad \Longleftrightarrow \qquad \widehat{h}_k\,\delta t \sim \widetilde{h}\,(f_k)$$

where

$$f_k = k\,\delta f \qquad \delta f = \frac{1}{N\,\delta t} \equiv \frac{1}{T}$$

- Aside: why not factor $\frac{1}{N} = \delta t\,\delta f$ & define DFT as $\widetilde{h}_k \equiv \widehat{h}_k\,\delta t$

$$\widetilde{h}_k = \delta t \sum_{j=0}^{N-1} h_j\,e^{-i2\pi jk/N} \quad \Longleftrightarrow \quad h_j = \delta f \sum_{k=-N/2}^{N/2-1} \widetilde{h}_k\,e^{i2\pi jk/N} \quad ?$$

$\{\widehat{h}_k\}$ involves only data $\{h_j\}$; $\{\widetilde{h}_k\}$ mixes in additional metadata δt

Time and Frequency Resolution

- Recall correspondence between continuous & discrete FT:

$$h_j = h(t_0 + j\,\delta t) \qquad \Longleftrightarrow \qquad \widehat{h}_k\,\delta t \sim \widetilde{h}\,(f_k)$$

where

$$f_k = k\,\delta f \qquad \delta f = \frac{1}{N\,\delta t} \equiv \frac{1}{T}$$

- N time points represent duration of $T = N\,\delta t$
- If freq indices are $-\frac{N}{2} \leq k \leq \frac{N}{2} - 1$ (complex) or $0 \leq k \leq \frac{N}{2}$ (real), then $|f_k| \leq \frac{N\,\delta f}{2} = \frac{1}{2\,\delta t} \equiv f_{\mathrm{Ny}}$.
- The Nyquist frequency f_{Ny} is half the sampling rate $\frac{1}{\delta t}$ & is the largest independent frequency represented in the DFT

Sampling and Aliasing

- $\widehat{h}_{N+k} = \widehat{h}_k$, but we said $\widehat{h}_k\,\delta t \sim \widetilde{h}(f_k)$ and in general $\widetilde{h}(f_{N+k}) \neq \widetilde{h}(f_k)$. Actually

$$\widehat{h}_k\,\delta t \approx \cdots + \widetilde{h}(f_{-N+k}) + \widetilde{h}(f_k) + \widetilde{h}(f_{N+k}) + \widetilde{h}(f_{2N+k}) + \cdots$$

- If sampled time series had Fourier cmpts w/ $|f| > f_{N/2} = f_{\mathrm{Ny}}$, those will be aliased with data in the range $-f_{\mathrm{Ny}} \leq f \leq f_{\mathrm{Ny}}$
- Generally low-pass filter time series to discard $|f| > f_{\mathrm{Ny}} = \frac{1}{2\,\delta t}$ content before sampling at rate $\frac{1}{\delta t}$
- Alternately, if data already band-limited ($\widetilde{h}(f) = 0$ for $|f| > B$) avoid aliasing by choosing δt so $f_{\mathrm{Ny}} > B$ i.e., $\frac{1}{\delta t} > 2B$ Confusingly, $2B$ is sometimes called the "Nyquist rate"

Illustration of Aliasing

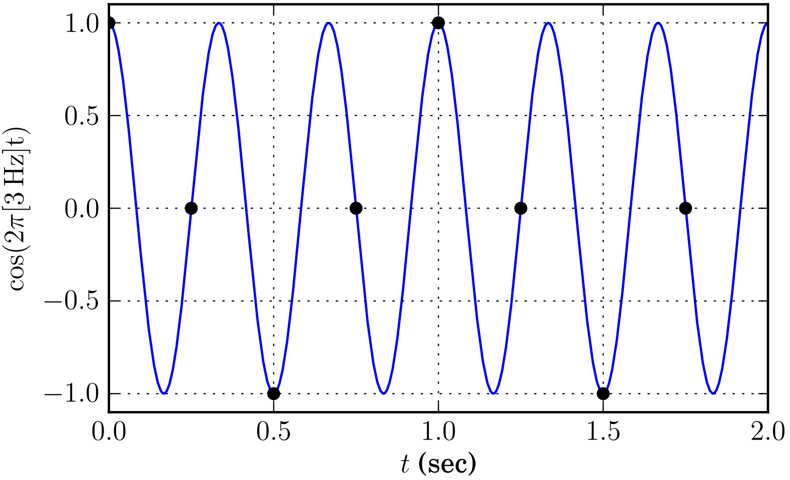


Illustration of Aliasing

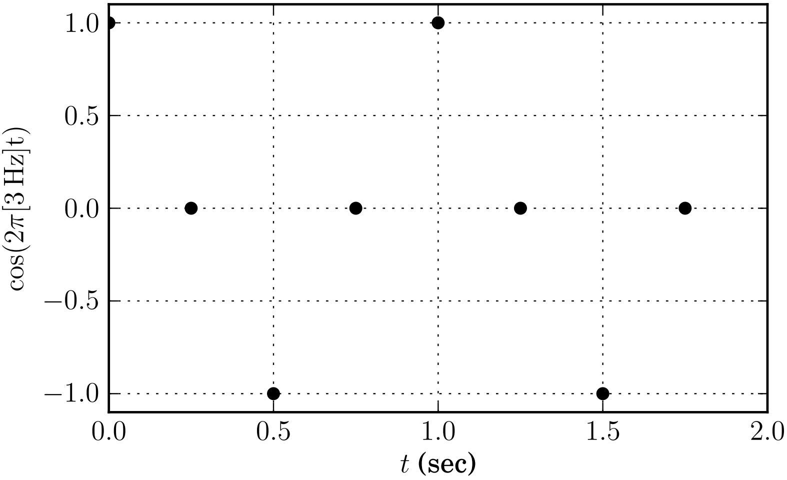
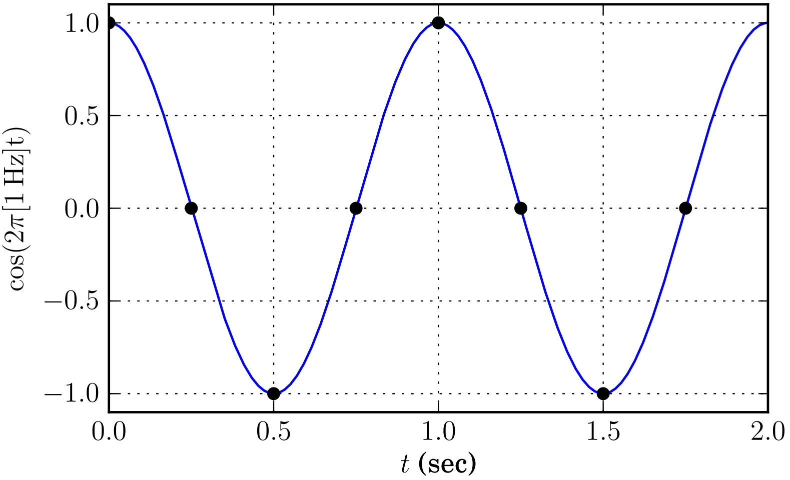


Illustration of Aliasing



The Fast Fourier Transform

- Naïve implementation of discrete FT as

$$\widehat{h}_k = \sum_{j=0}^{N-1} \left(e^{-i2\pi/N}\right)^{kj} h_j$$

would require $\mathcal{O}(N^2)$ ops & not be any faster than convolution

$$g_j = \sum_{m=0}^{N-1} A_{j-m} h_m$$

- Fast Fourier Transform (FFT) algorithms [Tukey & Cooley 1965] cut that to $\mathcal{O}(N \log N)$ by writing

$$N\text{-pt FT} \;\equiv\; \text{two } N/2\text{-pt FTs} \;\equiv\; \text{four } N/4\text{-pt FTs} \;\equiv \cdots$$

- Speedup is greatest for $N =$power of 2, but products of small prime factors are generally good
- Popular/fast GPLed implementation is FFTW

Exercise

- For $N = 16$, generate the discrete time series

$$h_j = \cos \frac{3\pi(j-2)}{8}$$

- Use an FFT routine to transform it & examine resulting $\widehat{h}_k$
- Apply an inverse FFT routine to $\widetilde{h}_k$ & compare results to h_j
- Use the angle sum formula to write h_j as a linear combination of $\cos \frac{3\pi j}{8} = \frac{e^{i3\pi j/8} + e^{-i3\pi j/8}}{2}$ and $\sin \frac{3\pi j}{8} = \frac{e^{i3\pi j/8} - e^{-i3\pi j/8}}{2i}$ and therefore as a linear combination of $e^{\pm i3\pi j/8}$ and compare the coëfficients to the components of the discrete Fourier transform.

Grant Tremblay's PhD Defense

Time	11:30
Location	Carlson Auditorium, 76-1125
Title	"Feedback-Regulated Star Formation in Cool Core Clusters of Galaxies"