Subject: Re: FFT and Parseval
Posted by Moritz Fischer on Mon, 28 Apr 2014 11:30:04 GMT
View Forum Message <> Reply to Message

This all about the conventions used when defining the Fourier transform:
- sometimes it's multiplied by a factor of 1/n in the forward transformation, and not in the inverse (as in the IDL forward transformation)
- sometimes it's scaled in the inverse transformation (as in the wikipedia definition of the DFT following the link you gave)
- and I as a mathematician prefer the scaling by 1/sqrt(n) in both forward and inverse transformation, because a) its symmetric and b) there won't be a factor in the formulation of the *conservation* (not scaling ...) of energy.

In either of the above cases you get
 x = FFT(FFT(x,-1) ,1)
but comparing the energies of time and spectrum you have to compensate the 1/sqrt(n) of the applied 1/n scaling factor, and your (1) reads:

 total(abs(x)^2.) = total(abs(fft0 * sqrt(N))^2.)

I guess the scaling convetion in IDL is choosen for performance reasons.

And note that the direction parameter is the sign of the argument of exp(.), i.e. negative for forward transformation, by most conventions.

Am 28.04.2014 12:58, schrieb baptiste.cecconi@obspm.fr:
> Dear IDL guys,
>
> I recently tried to check the conservation of energy (Parseval's
> theorem) through the IDL implementation of FFT, and I came to a
> somewhat surprising result.
>
> Here a sample code that shows my point:
>
> -----
>
> N=1000 x = randomn(0,N) ; random series of data with 1000 elements
> fft0 = fft(x,-1) ; fourier transform (to freq domain) of x
>
> print,total(x^2.) ; total energy of the signal in time domain
> print,total(abs(fft0)^2.)/N; total energy of the signal in freq
> domain (according to Parseval's theorem)
>
> fft1 = fft(x,1) ; inverse fourier transform (freq to time domain) of
> x print,total(abs(fft1)^2.)/N; total energy of the signal in freq

> domain (using inverse fft)
>
> -----
>
> From this little code, it is clear that
>
> (1) total(x^2.) = total(abs(fft0)^2.)*N (2) total(x^2.) =
> total(abs(fft1)^2.)/N
>
> While quation (2) is fully consistent with Parseval's equation, (1)
> is not, by a N^2 factor. In the IDL documentation, it is stated that
> "A normalization factor of 1/N, where N is the number of points, is
> applied during the forward transform." However, I'm not sure this
> solves anything here.
>
> I have some difficulties to convince myself that the direct FFT
> transform is invoked with a negative "direction" parameter (as stated
> in IDL documentation).
>
> Parseval theorem is recalled here:
> http://en.wikipedia.org/wiki/Parseval's_theorem (see DFT equation)
>