Subject: IDL 5.5, 2D FFT indexing confusion.
Posted by Pitufa on Tue, 19 Jul 2005 11:23:22 GMT
View Forum Message <> Reply to Message

Hi,

I have been trying to generate an real even function in fourier space that I can INVERSE FFT in order to get a function which has no imaginary part. I have no problems when the function is a vector, but I get an imaginary part when it is a two dimensional array.

Below is the test program that shows my problem. Am I defining wrong the variables 'centre' or 'nshift'?

I would be really grateful if someone could let me know what I am doing wrong.

Thanks!

Pitufa.

```
pro test_index

 npix   = 100
 centre = npix/2.d - 1.d
 nshift = npix/2 + 1

;1d example:

f    = abs(findgen(npix) - centre)
ifft = fft(shift(f,nshift),1,/double)

print,'1D: Imaginary maximum:', Max(Abs(Imaginary(ifft)))
print,'1D: Real maximum     :', Max(Abs(Double(ifft)))

;2d example:

;angle of position vector w.r.t x axis:

phi = dblarr(NPIX,NPIX)
FOR X = 0, NPIX-1 DO FOR Y = 0, NPIX-1 DO $
PHI[X,Y] = ATAN(Y*1.D - centre,X*1.D - centre)

TEST     = sin(2.d*phi)
IFFTTEST = FFT(SHIFT(TEST,nshift,nshift),1,/DOUBLE)

print,'2D: Imaginary maximum: ', $
 MAX(ABS(imaginary(IFFTTEST))),mean(ABS(imaginary(IFFTTEST)))
```

print,'2D: Real maximum   : ',MAX(ABS(double(IFFTTEST)))

end

Subject: Re: IDL 5.5, 2D FFT indexing confusion.
Posted by R.G. Stockwell on Thu, 21 Jul 2005 18:47:18 GMT
View Forum Message <> Reply to Message

"Pitufa" <c.c.calderon@gmail.com> wrote in message
news:1121966932.337140.282750@g49g2000cwa.googlegroups.com.. .

> but then it only works if the fourier array is real, and 2-5 don't make
> any sense physically! My conclusion from this is that you are better
> off using an odd N if you are doing this sort of calculation.
>
> Agree? Disagree?

If speed of calculation is a concern, "even N" is good (tends to be
more factorable), and powers of two are especially nice.
If you happen across a large prime number for N, then idl routines
are going to crawl. (and I have actually done that in the past).

Of course, actual measured time series are often even in N,
because the number of hours in day is even, minutes in hour, seconds in
minutes, etc all even
Fractions of a time unit are often in halfs or in tens, etc.

But I don't see any great compelling reason not use (nicely factorable) odd
numbers for N.

> Why the nyquist rows/columns need to be positive?

They don't. They (the symmetric nyquist pairs) just need to be the same
sign (and you had positive, so I said positive). Sorry, should have been
more clear.

As to why they must be the same, if you look at the math for a particular
nyquist pair,
(apply discrete inverse dft to just those two spectral components)
you basically get the following

using
image[x,y] = sum_u sum_v F[u,v] exp(i 2 pi (ux/M + vy/N))

input in the symmetric pair (A1 at (u,N/2) and A2 at (-u, N/2)
then you get
image(x,y) = (-1)^y * [A1*(cos(p1) + i sin(p1)) + A2*(cos(p1) - i

sin(p1))]

where the * are just normal multiplication, i've chosen to do the nyquist frequency in
the y direction which is where the (-1)^y comes from.
and p1 is shorthand for the argument p1 = i 2 pi ux/M.

So, requiring image to be real means that the sin components must cancel out therefore
A1sin(p1) = A2 sin(p1)
A1 = A2

[if p1 ~=0 of course. But that is your DC Nyquist points, and they do not have to be symmetric]

Cheers,
bob
