
Subject: Re: Again an FFT question

Posted by [Kenneth P. Bowman](#) on Wed, 27 Jun 2012 15:53:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article <614d4d33-39a8-4059-bc5f-f99f08fbc966@googlegroups.com>, Helder <helder@marchetto.de> wrote:

> Dear FFTers,
> I was just wondering about the translation of the FFT so that the what is
> located at (0,0) goes in the middle of the image (N/2,M/2).
> Until now I did this doing simply a shift of the FFT image, that is:
> `FFT_Img = SHIFT(FFT(Img),N/2,M/2)`
> Now I have seen that some people use a FFT "trick" to shift the image. They
> switch every second pixel of an image to its negative value. This is
> justified by the translation properties of the DFTs and results in a
> translation of half the image size (for those seeking to understand the math,
> try to multiply the function (image) by $\exp(i*2*\pi(u_0*x/N))$ and after
> integration you will get a translation of the Fourier image (signal for 1D)
> of $F(u-u_0)$.

>
> The result is that the two are not exactly the same. Very similar, but not
> the same.
> I have tried varying the size of the image or switching even instead of odd
> numbers index numbers , but could not get any improvement (with odd image
> sizes, the difference is even higher).
>
> Does anybody have a reason to use one way (`shift(fft(Img...))`) rather than
> the other (switch pixels with index $((x+y) \text{ MOD } 2 \text{ EQ } 1)$)?
>
> I'm more confident using SHIFT, but I would just like to understand why the
> other method gives different values.
>
> Thanks,
> Helder

If you do an FFT of an 8-point array, the output frequencies are stored in this order

$f = [0, 1, 2, 3, 4, -3, -2, -1]$

The IDL FFT always does a complex FFT. If the input data are real, then the real part of $f=0$ is the mean, the imaginary part is 0 to within roundoff error.

The positive and negative frequencies are complex conjugates of each other - again within roundoff error.

I think the differences that you are seeing are due to round-off error. Check the magnitudes of the differences.

Ken Bowman

Subject: Re: Again an FFT question

Posted by [lecacheux.alain](#) on Wed, 27 Jun 2012 16:22:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 26 juin, 23:11, Helder <hel...@marchetto.de> wrote:

```
> Dear FFTers,
> I was just wondering about the translation of the FFT so that the what is located at (0,0) goes in
the middle of the image (N/2,M/2).
> Until now I did this doing simply a shift of the FFT image, that is:
> FFT_Img = SHIFT(FFT(Img),N/2,M/2)
> Now I have seen that some people use a FFT "trick" to shift the image. They switch every
second pixel of an image to its negative value. This is justified by the translation properties of the
DFTs and results in a translation of half the image size (for those seeking to understand the math,
try to multiply the function (image) by  $\exp(i*2*\text{Pi}(u_0*x/N))$  and after integration you will get a
translation of the Fourier image (signal for 1D) of  $F(u-u_0)$ .
>
> I have tested the difference between the two shifting methods with the following code (in double
precision just to be sure I wouldn't get rounding errors or so):
> ;*****
> n = 256
> img = RANDOMU(S, n, n) ; Eventually put your image here, with the *right* size
> xx = LINDGEN(n) # (LONARR(n)+ 1)
> yy = (LONARR(n)+ 1) # LINDGEN(n)
> xxyy = xx+yy
> OddCenteredImg = Img
> EvenCenteredImg = Img
> Odd = WHERE((xxyy MOD 2), COMPLEMENT=Even)
> OddCenteredImg[Odd] = -Img[Odd]
> EvenCenteredImg[Even] = -Img[Even]
> WINDOW, XSIZE=2*n, YSIZE=3*n
> TVSCL, Img, 0 ;Transform by standard shifting
> TVSCL, OddCenteredImg, 1 ;Transform by standard shifting
> TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2))), 2 ;Transform by standard shifting
> TVSCL, ALOG(ABS(FFT(OddCenteredImg, /DOUBLE))), 3 ;Transform by pixel sign
inversion
> TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2)) - ABS(FFT(OddCenteredImg,
/DOUBLE))), 4 ;Power Spectrum Image Difference
> TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2)) - ABS(FFT(EvenCenteredImg,
/DOUBLE))), 5 ;Power Spectrum Image Difference
> ;*****
>
```

> The result is that the two are not exactly the same. Very similar, but not the same.
> I have tried varying the size of the image or switching even instead of odd numbers index numbers , but could not get any improvement (with odd image sizes, the difference is even higher).
>
> Does anybody have a reason to use one way (shift(fft(lmg...))) rather than the other (switch pixels with index ((x+y) MOD 2 EQ 1)?
>
> I'm more confident using SHIFT, but I would just like to understand why the other method gives different values.
>
> Thanks,
> Helder
>
>

You can use the /CENTER keyword in FFT function since IDL7.1
alx.

Subject: Re: Again an FFT question
Posted by [David Fanning](#) on Wed, 27 Jun 2012 16:33:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

alx writes:

> You can use the /CENTER keyword in FFT function since IDL7.1

Well, it was *present* in IDL 7.1. I don't think
it actually worked until later.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Again an FFT question
Posted by [lecacheux.alain](#) on Wed, 27 Jun 2012 16:45:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 27 juin, 18:33, David Fanning <n...@idlcoyote.com> wrote:

> alx writes:

>> You can use the /CENTER keyword in FFT function since IDL7.1

>

> Well, it was *present* in IDL 7.1. I don't think

> it actually worked until later.

>

> Cheers,

>

> David

>

> --

> David Fanning, Ph.D.

> Fanning Software Consulting, Inc.

> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>

> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

>

>

?

```
IDL> print,fft(cos(2*!pi*findgen(16)/16*3))
```

```
( 5.06635e-008,  0.000000)( 8.06580e-008, 3.52675e-008)
```

```
(-9.04610e-009, 2.63418e-008)(  0.500000, 1.28007e-007)
```

```
(-9.08976e-008, 1.49012e-008)
```

```
(-2.65443e-008,-1.35536e-008)( 1.20273e-008, 2.63418e-008)
```

```
(-7.60014e-008,-9.51177e-008)( 1.25169e-007, -0.000000)
```

```
(-7.60014e-008, 9.51177e-008)
```

```
( 1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)
```

```
(-9.08976e-008,-1.49012e-008)(  0.500000,-1.28007e-007)
```

```
(-9.04610e-009,-2.63418e-008)
```

```
( 8.06580e-008,-3.52675e-008)
```

```
IDL> print,fft(cos(2*!pi*findgen(16)/16*3),/CENTER)
```

```
( 1.25169e-007, -0.000000)(-7.60014e-008, 9.51177e-008)
```

```
( 1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)
```

```
(-9.08976e-008,-1.49012e-008)
```

```
(  0.500000,-1.28007e-007)(-9.04610e-009,-2.63418e-008)
```

```
( 8.06580e-008,-3.52675e-008)( 5.06635e-008,  0.000000)
```

```
( 8.06580e-008, 3.52675e-008)
```

```
(-9.04610e-009, 2.63418e-008)(  0.500000, 1.28007e-007)
```

```
(-9.08976e-008, 1.49012e-008)(-2.65443e-008,-1.35536e-008)
```

```
( 1.20273e-008, 2.63418e-008)
```

```
(-7.60014e-008,-9.51177e-008)
```

Both statements give expected results: i.e. non zero values at positions 3 and 13 (CENTER=0) or 5 and 11 (CENTER=1).

alx.

Subject: Re: Again an FFT question

Posted by [lecacheux.alain](#) on Wed, 27 Jun 2012 16:45:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 27 juin, 18:45, alx <lecacheux.al...@wanadoo.fr> wrote:

> On 27 juin, 18:33, David Fanning <n...@idlcoyote.com> wrote:

>

>

>

>

>

>> alx writes:

>>> You can use the /CENTER keyword in FFT function since IDL7.1

>

>> Well, it was *present* in IDL 7.1. I don't think

>> it actually worked until later.

>

>> Cheers,

>

>> David

>

>> --

>> David Fanning, Ph.D.

>> Fanning Software Consulting, Inc.

>> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>

>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

>

> ?

> IDL> print,fft(cos(2*!pi*findgen(16)/16*3))

> (5.06635e-008, 0.000000)(8.06580e-008, 3.52675e-008)

> (-9.04610e-009, 2.63418e-008)(0.500000, 1.28007e-007)

> (-9.08976e-008, 1.49012e-008)

> (-2.65443e-008,-1.35536e-008)(1.20273e-008, 2.63418e-008)

> (-7.60014e-008,-9.51177e-008)(1.25169e-007, -0.000000)

> (-7.60014e-008, 9.51177e-008)

> (1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)

> (-9.08976e-008,-1.49012e-008)(0.500000,-1.28007e-007)

> (-9.04610e-009,-2.63418e-008)

> (8.06580e-008,-3.52675e-008)

> IDL> print,fft(cos(2*!pi*findgen(16)/16*3),/CENTER)

> (1.25169e-007, -0.000000)(-7.60014e-008, 9.51177e-008)

> (1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)

> (-9.08976e-008,-1.49012e-008)

> (0.500000,-1.28007e-007)(-9.04610e-009,-2.63418e-008)

> (8.06580e-008,-3.52675e-008)(5.06635e-008, 0.000000)

> (8.06580e-008, 3.52675e-008)

> (-9.04610e-009, 2.63418e-008)(0.500000, 1.28007e-007)

> (-9.08976e-008, 1.49012e-008)(-2.65443e-008,-1.35536e-008)

> (1.20273e-008, 2.63418e-008)

> (-7.60014e-008,-9.51177e-008)
> Both statements give expected results: i.e. non zero values at
> positions 3 and 13 (CENTER=0) or 5 and 11 (CENTER=1).
> alx.

I forgot to say, by using 8.2.
alx.

Subject: Re: Again an FFT question
Posted by [Helder Marchetto](#) on Thu, 28 Jun 2012 09:00:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Wednesday, June 27, 2012 6:45:46 PM UTC+2, alx wrote:
> On 27 juin, 18:45, alx <lecacheux.al...@wanadoo.fr> wrote:
>> On 27 juin, 18:33, David Fanning <n...@idlcoyote.com> wrote:
>>
>>
>>
>>
>>
>> alx writes:
>>> You can use the /CENTER keyword in FFT function since IDL7.1
>>
>> Well, it was *present* in IDL 7.1. I don't think
>> it actually worked until later.
>>
>> Cheers,
>>
>> David
>>
>> --
>> David Fanning, Ph.D.
>> Fanning Software Consulting, Inc.
>> Coyote's Guide to IDL Programming:<http://www.idlcoyote.com/>
>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
>>
>> ?
>> IDL> print,fft(cos(2*pi*findgen(16)/16*3))
>> (5.06635e-008, 0.000000)(8.06580e-008, 3.52675e-008)
>> (-9.04610e-009, 2.63418e-008)(0.500000, 1.28007e-007)
>> (-9.08976e-008, 1.49012e-008)
>> (-2.65443e-008,-1.35536e-008)(1.20273e-008, 2.63418e-008)
>> (-7.60014e-008,-9.51177e-008)(1.25169e-007, -0.000000)
>> (-7.60014e-008, 9.51177e-008)
>> (1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)
>> (-9.08976e-008,-1.49012e-008)(0.500000,-1.28007e-007)
>> (-9.04610e-009,-2.63418e-008)

```
>> ( 8.06580e-008,-3.52675e-008)
>> IDL> print,fft(cos(2*!pi*findgen(16)/16*3),/CENTER)
>> ( 1.25169e-007, -0.000000)(-7.60014e-008, 9.51177e-008)
>> ( 1.20273e-008,-2.63418e-008)(-2.65443e-008, 1.35536e-008)
>> (-9.08976e-008,-1.49012e-008)
>> ( 0.500000,-1.28007e-007)(-9.04610e-009,-2.63418e-008)
>> ( 8.06580e-008,-3.52675e-008)( 5.06635e-008, 0.000000)
>> ( 8.06580e-008, 3.52675e-008)
>> (-9.04610e-009, 2.63418e-008)( 0.500000, 1.28007e-007)
>> (-9.08976e-008, 1.49012e-008)(-2.65443e-008,-1.35536e-008)
>> ( 1.20273e-008, 2.63418e-008)
>> (-7.60014e-008,-9.51177e-008)
>> Both statements give expected results: i.e. non zero values at
>> positions 3 and 13 (CENTER=0) or 5 and 11 (CENTER=1).
>> alx.
>
> I forgot to say, by using 8.2.
> alx.
```

Thank you all for the answers. I guess the difference between changing sign of every second element and shift the array is due to a round-off error. This becomes clear when analyzing the differences between the two results using double precision or not.

Thanks also for the explanation of how the frequencies in an FFT are displayed.

I guess I'll have to update old code and remove the shift and use instead the center option.

Regards,
Helder

Subject: Re: Again an FFT question

Posted by [Kenneth P. Bowman](#) on Thu, 28 Jun 2012 13:07:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

In article <27bbd728-a65d-4925-bd54-d003cfad6493@googlegroups.com>,
Helder <helder@marchetto.de> wrote:

```
> Thank you all for the answers. I guess the difference between changing sign
> of every second element and shift the array is due to a round-off error. This
> becomes clear when analyzing the differences between the two results using
> double precision or not.
> Thanks also for the explanation of how the frequencies in an FFT are
> displayed.
> I guess I'll have to update old code and remove the shift and use instead the
> center option.
```

I didn't plug it yesterday, but you might want to look at the chapter on FFTs in my book.

http://www.amazon.com/An-Introduction-Programming-IDL-Interactive/dp/012088559X/ref=sr_1_1?ie=UTF8&qid=1340888241&sr=8-1&keywords=idl+bowman

I used to center FFTs, but have gotten away from that. For real input data, the positive and negative frequencies are complex conjugates, so the negative frequencies are redundant and you can get all of the information by plotting only the positive frequencies.

For calculations in the spectral domain, such as filtering, I usually compute the frequencies (or periods) in dimensional units as a function of the transform array indices and use that. Then you don't have to remember to uncenter when transforming back.

Ken Bowman
<http://idl.tamu.edu/>
