Subject: FFTs in IDL
Posted by tom on Thu, 15 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

Newsgroups: comp.lang.idl-pvwave
Expires:
Sender:
Followup-To: tom@quake.stanford.edu
Distribution:
Organization: Stanford University, CA USA
Cc:
Subject: IDL FFT Efficiency and Alternatives
Summary:

Does anyone know which algorithm the FFT.PRO routine uses? The manual
implies that it is the Cooley-Tukey with "convert-to-complex" algorithm.
This should be very much slower than the algorithms tailored for real
data (especially for large 2-D images like the ones I'm trying to filter).

Numerical Recipes lists some but they are the old "N = factor of 2 or die"
variety and zero padding a 1300x1024 array to 2048x1200 just seems like
a really stupid thing to do. The Winograd algorithms would be ideal here.

A related question: if I end up having to not be lazy and type in a book
algorithm, would it be faster to implement as a CALL_EXTERNAL or a native
IDL routine? I have no experience with external calls; someone told me they
can be slow due to translation (?).

Replies by email please.

Tom Berger
Center for Space Science and Astrophysics
Stanford University

tom@quake.stanford.edu

Subject: Re: FFTs in IDL
Posted by Sergei Senin on Fri, 23 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

Achim Hein <hein@maxwell.nv.et-inf.uni-siegen.de> wrote:

>>> skipped<<<
> -It is very important to know that there is a little mistake in the
> WAVE-Algorithm not in the IDL-Algorithm, try to plot test=findgen(4097)
> and oplot,fft(fft(test,-1),1) you will see what I mean.

Sorry, a I'm a bit late for this discussion, but nevertheless:

```
wave> test=findgen(4097)
wave> print, max(test-fft(fft(test,-1),1))
wave> (      29.4698,      0.247685)
```

and:

```
wave> test=findgen(4096)
wave> print, max(test-fft(fft(test,-1),1))
wave> (    0.0163574, -0.000450487)
```

There seems to be no mistake, but simply no check in the procedure for the number of elements in the array being 2^X.

> -In my opinion zero padding is definitly not a stupid thing to do.

No, but sometimes painfull :-) Could REBIN help in doing this for 2D arrays?

> Achim Hein
> Center of Sensor Systems
> Remote Sensing and optimal Signal Processing
> University of Siegen
>

--
Sergei Senin
University of Portsmouth
Department of Electrical and Electronic Engineering
Microwave, Telecommunications and Signal Processing Research Group
Anglesea Building, Anglesea Road,
Portsmouth,P01 3DJ,England.
ss@ee.port.ac.uk, http://www.ee.port.ac.uk:80/~ss-www/

Subject: Re: FFTs in IDL
Posted by Sergei Senin on Tue, 27 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

Achim Hein <hein@maxwell.nv.et-inf.uni-siegen.de> wrote:
> And thats exactly the point where this discussion began.'Is zero-padding a
> stupid thing to do'?!

There are algorithms which allow fft without zero padding

> But I have to transform arrays gt 2 GByte (naturally power of two) and I
> am very interested in 'high speed' Fourier-Transform-Algorithms. Is there

> a paper of Ron Mayer or anything else?

I found this stuff on the Net and don't remember the address, but can mail it
to you, (or anybody interested as it matters) or put it on my WWW page. Which
one do you prefer ?

Cheers

--
Sergei Senin
University of Portsmouth
Department of Electrical and Electronic Engineering
Microwave, Telecommunications and Signal Processing Research Group
Anglesea Building, Anglesea Road,
Portsmouth,P01 3DJ,England.
ss@ee.port.ac.uk, http://www.ee.port.ac.uk:80/~ss-www/

Subject: Re: FFTs in IDL
Posted by Achim Hein on Tue, 27 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

<As far as I understand the FFT pro in PV-Wave is using the basic Cooley -
<Tukey
<algorithm which works best with the arrays that are power of two. There
<is a
<number of better routines, one, for instance written by Ron Mayer which
<includes many optimizations and is quite fast.

<I wonder if anybody (I mean users) managed to write or adopt any other
<than
<the abovementioned "basic" pro for the Wave....

And thats exactly the point where this discussion began.'Is zero-padding a
stupid thing to do'?!

But I have to transform arrays gt 2 GByte (naturally power of two) and I
am very interested in 'high speed' Fourier-Transform-Algorithms. Is there
a paper of Ron Mayer or anything else?

Thanks

Achim Hein
Center for Sensor Systems
Optimal Signal Processing, Remote Sensing - SAR
University of Siegen
Germany

Subject: Re: FFTs in IDL
Posted by Sergei Senin on Tue, 27 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

Achim Hein <hein@maxwell.nv.et-inf.uni-siegen.de> wrote:
> It seems so, but I think a Fourier-Transform-Algorithm has to
> run for every array length and ist is IDL or MathLab that shows
> correct transformation results for every array length. The algorithm
> using no base two array length works a lot slower but he has to be correct or
> not ?

As far as I understand the FFT pro in PV-Wave is using the basic Cooley - Tukey
algorithm which works best with the arrays that are power of two. There is a
number of better routines, one, for instance written by Ron Mayer which
includes many optimizations and is quite fast.

I wonder if anybody (I mean users) managed to write or adopt any other than
the abovementioned "basic" pro for the Wave....

Cheers

--
Sergei Senin
University of Portsmouth
Department of Electrical and Electronic Engineering
Microwave, Telecommunications and Signal Processing Research Group
Anglesea Building, Anglesea Road,
Portsmouth,P01 3DJ,England.
ss@ee.port.ac.uk, http://www.ee.port.ac.uk:80/~ss-www/

Subject: Re: FFTs in IDL
Posted by Achim Hein on Tue, 27 Feb 1996 08:00:00 GMT
View Forum Message <> Reply to Message

<There seems to be no mistake, but simply no check in the procedure for
<the number of elements in the array being 2^X.

It seems so, but I think a Fourier-Transform-Algorithm has to
run for every array length and ist is IDL or MathLab that shows
correct transformation results for every array length. The algorithm
using no base two array length works a lot slower but he has to be correct or
not ?

<No, but sometimes painfull :-) Could REBIN help in doing this for 2D
<arrays?

REBIN interpolates the array and gives you no more information but it costs a lot of time, so I think it is not the one solution.

Thanks

Achim Hein
Center for Sensor Systems
Optimal Signal Processing, Remote Sensing - SAR
University of Siegen
Germany