Subject: Different FFT times for same array size ?
Posted by jcami on Wed, 24 Aug 2005 05:55:26 GMT
View Forum Message <> Reply to Message

Hi,

First of all :

IDL> print, !version
{ x86 linux unix linux 6.2 Jun 20 2005 32 64}

I am using the blk_con program that comes with the IDL distribution. Basically, this uses fft's to do a convolution. I have a large number of arrays that all have the same number of elements, and I convolve them all with the same kernel using the blk_con program. I would think that the execution time should then be roughly the same, but apparently it is not. To get some more information on the timing, I added a short timing command in the while loop so that that part now reads :

```
  while loop le k-1L do begin
    input  = [input[blen-p+1:blen-1], signal[lower:upper]]
    tt=systime(1)
    output = fft((z_filter * fft(input, -1) ),1) * blen
    print, "blk_con ", loop, k-1L, systime(1)-tt
 ;Next output segment.
    result[ip] = double ?
DOUBLE(output[p-1:blen-1]):FLOAT(output[p-1:blen-1])
    ip = ip + blen - p + 1L
    lower = upper + 1L
    upper = lower + blen - p
    loop  = loop + 1L
  endwhile
```

Since the sizes of the chunks that go into the fft are always the same, and always a power of 2, the execution time should always be (roughly) the same, right ??
Now here's an example of what the print statement says for one of my arrays :

```
blk_con          2        150     0.034678936
blk_con          3        150     0.036569834
blk_con          4        150     0.033971071
blk_con          5        150     0.036265135
blk_con          6        150     0.039370060
blk_con          7        150     0.049736023
blk_con          8        150     0.039547920
blk_con          9        150     0.054105043
blk_con         10         150     0.036910057
```

```
blk_con         11        150     0.046661854
blk_con         12        150      0.13956809
blk_con         13        150     0.046625137
blk_con         14        150     0.058930874
blk_con         15        150     0.036964893
blk_con         16        150     0.045490980
blk_con         17        150     0.063371181
blk_con         18        150     0.037928820
blk_con         19        150     0.048100948
blk_con         20        150     0.032199144
blk_con         21        150     0.042580128
blk_con         22        150     0.034416914
blk_con         23        150     0.036769152
blk_con         24        150     0.033674002
blk_con         25        150     0.045473099
blk_con         26        150     0.035262823
blk_con         27        150     0.039651871
blk_con         28        150     0.037656784
blk_con         29        150     0.040482998
blk_con         30        150       2.7050159
blk_con         31        150       2.8096600
blk_con         32        150       2.8986051
blk_con         33        150       2.7503240
blk_con         34        150       2.8176029
...
blk_con         73        150       2.6031480
blk_con         74        150     0.032071114
blk_con         75        150     0.029819965
blk_con         76        150       2.5321939
...
blk_con        150        150       2.5533669
```

So the first 28 segments go fast, and from then on things slow down by a factor of 50 (well, except those 2 funny ones) !!

Am I missing something here ???

Note : I get the same sort of troubles if I use the astrolib's convolve function, which does pretty much the same.
