Subject: Re: nonuniform FFT
Posted by MKatz843 on Mon, 07 Apr 2003 07:09:58 GMT
View Forum Message <> Reply to Message

b_gom@hotmail.com (Brad Gom) wrote in message
> has anyone out there implemented a FFT routine that handles
> nonuniformly gridded samples?

If your samples aren't uniformly gridded, I'm not sure how you're going to get an "FFT" to work. However, there's no reason why you can't implement a simple (discreet) Fourier Transform minus the "fast" part.

If you know your non-uniform x values, then for arbitrary k values you could always Fourier transform your input array, A, like this

```
i = complex(1,0)
f_k = total( A * exp( i*k*x) )
```

That's just one Fourier component, and one dimension for x, but scaling that is trivial.
Of course, computing the FT in this way, one component at a time, should probably be called a SFT (Slow Fourier Transform).

Depending on your input/output needs, you may be able to implement a DFT (Discreet Fourier Transform) using IDL's vector/matrix math and then it can be Much faster. You have a vector of x values where you've sampled A. You have a vector of k values where you want to know the Fourier components. If memory serves, the outer product of those two vectors gives a matrix which is the k*x part of exp(i * k*x). The proper matrix multiplication by the input array A can yield your FT in a jiffy--much faster than using a for-do loop.

MKatz843