

---

Subject: Re: IDL performance and FFTs (was: call external speed)

Posted by [stevenj](#) on Wed, 16 Sep 1998 07:00:00 GMT

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

---

steinhh@ulrik.uio.no (Stein Vidar Hagfors Haugan) wrote:

>> - If the FFTW (which is free) outperforms the native FFT in IDL,  
>> why don't RSI use that implementation? Is this a silly question?  
>  
> Since FFTW is free under the Gnu Public License (GPL), RSI would  
> have to \*give away\* IDL under the GPL license in order to include  
> FFTW as part of IDL (I think). Not a good business practice! There  
> are two options, however:

That's not entirely accurate, although I agree with your conclusion that RSI is unlikely to use FFTW under the GNU license.

It *is* legal to sell GPL'ed software (e.g. Red Hat Linux); you can also sell technical support, documentation, etcetera. However, you have to allow unrestricted redistribution and you also have to make your entire program's source code available under the GPL, so most companies are unwilling to use this business model. (In practice, you can't charge very much for the program itself if it is GPL'ed, although you can charge more for support.)

> 1. RSI could produce a wrapper for FFTW and make it available  
> through their web site so the user could make a dynamically loadable  
> module (and the user would have to fetch the FFTW separately). AFAIK  
> this would mean that FFTW is not "sold as a part of IDL".

Nope, you can't get around the GPL in this way. To quote R. Stallman of GNU, "A GPL-covered plug-in that is designed to be combined with [only] a non-free master program is a form of combined work, and a violation of the GPL." (The original authors can make an exception allowing their code to be used in such a plugin, but no one else can do so.)

(Otherwise, the GPL would essentially devolve to the LGPL--you could link any GPL'ed code you wanted into a non-free program just by making it a "plugin." For more info, do a search on Dejanews for: ~a (rms@santafe.edu) & ~g (gnu.misc.discuss) & "Plug-ins")

You can make such a plugin for your own use, but you can't distribute it. (In any case, you are probably right in that such a plugin wouldn't be widely useful unless it came with IDL.)

> 2. The FFTW site mentions the possibility of non-free licenses. This  
> may be an idea to look into (after all, they are paying licences  
> for routines from Numerical Recipes).

Yup, in the case of FFTW, unrestricted licenses are available, for a fee, from MIT. That's up to RSI, though.

- > However, I'm not so sure the \*algorithm\* of IDL's FFT is so bad,
- > I suspect (lacking) compiler optimizations to play a part here.
- > This is probably also the reason for the drop in performance
- > experienced in going from Fortran to C source code (Fortran
- > code is easier to optimize - the compiler can make stronger
- > assumptions on what is going on).

I tend to think that this is overstated, since the aliasing problems of C can be avoided by an alert programmer (you just have to store dereferenced pointers in local variables for performance-critical regions). However, uncared translations of Fortran programs do have a tendency to suffer.

Cordially,  
Steven G. Johnson

---