Subject: Re: IDL performance and FFTs (was: call external speed)
Posted by steinhh on Wed, 16 Sep 1998 07:00:00 GMT
View Forum Message <> Reply to Message

In article <35ff8db4.0@d2o203.telia.com> roy.hansen@triad.no
(Roy E. Hansen) writes:
> We did a small comparison of the FFT performance in IDL 5.1.1
> compared with the Matlab 5.2 version for a PII-400 with Win-NT,
> and found that Matlab was approx 4 times faster. We also found
> that the FFT in IDL 5.1.1 was faster than in IDL 5.1 on an other
> PII-400 with Win95.

I'm surprised the difference to Matlab 5.2 was so large.

However, I know that the general performance of IDL went down
with the switch to C (rather than Fortran) as the source
language. I think this happened with the 3.6 -> 4.0 version
change. I did post some comparative timing estimates at
the time (not for the FFT though, as I recall).

>
> This raises a few questions:
>
> - Does there exist any optimized versions of IDL for the PII and
>   PPro with W95 and Win-NT?
>
> - Does anybody know what the performance gain is using an optimized
>   version compared to the standard version?
>
> - Is the IDL performance operating system dependent for the INTEL
>   platform?

It may be that different compilers (thus different optimization
strategies) are used??

> - Are there any benchmarks of numerical performance for IDL
>   compared to other software packages, like Matlab?

I would like to see some of those comparisons, too. Not just
for PCs, though.

> - 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:

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". Not really a good solution, since programs using this module would crash on any system that *didn't* have this "extra" installed.

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).

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).

Anyway, RSI should keep an eye out to the competition wrt. the efficiency of basic mathematical routines in IDL vs other packages. Switching compilers (or maybe just turning on compiler switches :-) may be an option.

Regards,

Stein Vidar