Subject: Re: Searching for fast linear interpolation routine
Posted by Christian Soeller on Sun, 06 Apr 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Christian Marquardt <marq@strat01.met.fu-berlin.de> writes:

>
> I find that using INTERPOL to do 1-d interpolations on large irregular
> grids can be extremely time consuming. The problem with INTERPOL is
> that it uses a linear search to find where a given field point fits
> into the irregular grid. Below you'll find my solution to the
> problem. The procedure FINDEX uses a binary search to obtain a
> "floating point index" which can be used with INTERPOLATE. I have
> found that the FINDEX + INTERPOLATE method can be up to 70 times
> faster than using INTERPOL. I am donating this procedure to the IDL
> community in hopes of saving untold millions of machine cycles that
> would otherwise have been wasted in futile linear searches. But
> seriously, give it a try and let me know if it breaks.

If the indices you are calculating interpolates at are often close to each other you can even speed up the binary search some further by using the last calculated index as a starting guess for the next, see "How to search an ordered table" in "Numerical Recipes" (now available as an online web document), especially the hunt routine. I have implemented that method in a 'call external' like way in another data language (PDL = Perl Data Language == free(!)) and get for the equivalent of the published 'test6' example an execution time of 0.016s compared to 2.09s to the IDL interpolate example. So more than a factor of 100. If you go for the ultimate speed than a 'call external' (or if you prefer 'linkimage') implementation using hunt to find the indices should be the way to go.

Christian

-- --------
Christian Soeller   mailto: csoelle@sghms.ac.uk
St. Georges Hospital Medical School   Dept. of Pharmacology
Cranmer Terrace   London SW17 0RE