

---

Subject: Re: Searching for fast linear interpolation routine  
Posted by [Roger J. Dejus](#) on Mon, 07 Apr 1997 07:00:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

In response to suggestions for a fast linear interpolation routine:

1) Liam Gumley wrote:

- > If you can find a way to use the built-in routine INTERPOLATE, you
- > will see a dramatic speed increase - it is very fast on 1D, 2D or 3D
- > arrays.

-- Yes, I can use INTERPOLATE rather than INTERPOL and there was a noticeable speed increase. Note, the time will now be determined by the time spent in the routine preparing to call INTERPOLATE (see 2) and 3) below).

2) Wayne Landsman wrote:

- > The INTERPOL function consists of two steps: (1) finding the effective
- > index of the interpolation value (e.g. it is located between indices
- > 12 and 13 in the abscissa), and (2) performing the interpolation. I
- > know of at least 3 ways in which the speed of the INTERPOL function
- > can be improved:
- >
- > (1) INTERPOL uses an incremental search algorithm to find the
- > effective index, whereas the quickest way to search a monotonic array
- > is a binary search (divide and conquer).
- > (2) the effective index search is not vectorized
- > (3) the intrinsic INTERPOLATE function (available since V2.2) is not
- > used to do the interpolation
- >
- > The program LINTERP in the IDL Astronomy Library incorporates these
- > three improvements and I get a factor of four improvement in speed on
- > my Ultra-2.
- >
- > <http://idlastro.gsfc.nasa.gov/ftp/pro/math/linterp.pro>
- >
- > LINTERP calls the program TABINV to find the effective index
- >
- > <http://idlastro.gsfc.nasa.gov/ftp/pro/math/tabinv.pro>
- >
- > These procedures also call ISARRAY (from the JHUAPL library) and
- > ZPARCHECK
- >
- > <http://idlastro.gsfc.nasa.gov/ftp/pro/jhuapl/isarray.pro>
- > <http://idlastro.gsfc.nasa.gov/ftp/pro/misc/zparcheck.pro>

-- I agree, and I also got a factor of four speed increase on my DEC

Alpha using LINTERP (and TABINV, ISARRAY, and ZPARCHECK).

3) Chris Marquardt wrote:

> Recently, Paul Richiazzi posted a solution to this problem to the  
> newsgroup; I'll attach his posting...  
>  
> Hope this helps,  
>  
> Chris Marquardt.  
>

-- The solution from Paul Richiazzi (using the routine called FINDEX) is similar to the solution by Wayne Landsman (FINDEX replaces the call to TABINV above), although slower by a factor of two (for my sample problem). The FINDEX routine will show a speed increase by about 25% by switching indices so that the array JLIM=LONARR(2,NV) is replaced by JLIM=LONARR(NV,2). Remember, IDL (and FORTRAN) are column oriented = first index varies the fastest.

4) Christian Soeller wrote:

> 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  
>

-- Yes, the starting index (guess) should be the previous index when calculating indices which are close. I don't know whether this can be vectorized though. Also, I would like to see this implemented in an IDL routine rather than calling an external program (which I'm familiar with but it is more straightforward to call an IDL program and it causes less problems when making portable codes). I'm also questioning the quoted time above of 0.016 s which is approx. the time spent in the routine INTERPOLATE without first calling an index searching routine (such as TABINV or FINDEX above).

Please feel free to comment on the above.

Thanks, Roger. (dejus@aps.anl.gov).

---