Subject: Re: array concatenation and optimization
Posted by Craig Markwardt on Thu, 27 Sep 2001 16:34:52 GMT
View Forum Message <> Reply to Message

Paul van Delst <paul.vandelst@noaa.gov> writes:
>> This is working but it has increased the processing time of my loop by an
>> order of magnitude. Is there a better way to do this? Is there a reason
>> this is so slow?
>
> A while back someone else in this newsgroup more knowledgable than
> me (I think Craig?) explained why this is a bad method when the
> number of points is very large. Suffice it to say for something like
> up to maybe a couple hundred points, concatenation is o.k. Any more
> and you'd be needlessly sucking up CPU cycles shifting/concatenating
> large blocks of data on the fly.

Yes, this was me, among a lot of other people on the newsgroup. I have always been a fan of "chunking," which means to read your data in reasonably large chunks. Instead of reading 1 line at a time, read 1000 lines at a time.

The whole point is that you want your loop overhead to be smaller than the actual work you do in the loop. By having an "x = [x, new]" command at every step in the loop, most of the time of the loop is spent allocating, appending, assigning, and unallocating.

My pet favorite is to read the file line by line, but grow the array in chunks. I usually grow it by powers of two until a certain limit. Example (not tested),

```
nmax = 100L             ;; Start array with 100 elements
xarray = fltarr(nmax)
ntot = 0L
while NOT eof(unit) do begin
  readf, unit, x
  if nmax LE ntot then begin
    ;; Grow the array if needed
    nnew = nmax < 500000 ;; Double the array, up to 500,000 elements
    xarray = [temporary(xarray), fltarr(nnew)]
    nmax = nmax + nnew
  endif
  ;; Assign the array
  xarray(ntot) = x
  ntot = ntot + 1
endwhile
```

Craig

--

---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------