Subject: Re: Timing results - matrix multiply vs. indexing
Posted by chase on Thu, 30 Jun 1994 19:03:47 GMT
View Forum Message <> Reply to Message

>>>> > "David" == David Landers <landers@tsunami.dseg.ti.com> writes:
In article <1994Jun30.143333.2221@mksol.dseg.ti.com> landers@tsunami.dseg.ti.com (David Landers) writes:

David> Hi gang, About a week or two ago we were discussing ways to
David> make a 2-D array from a 1-D array (which represented constant
David> rows or columns). To summarize, the two (primary) ways to do
David> this are:

David> m = n_elements(array1)

David> array2 = replicate(1.0, n) # array1
David> or
David> array2 = array1(lindgen(n,m) / n))

I seem to recall someone suggesting another method using:

array2 = rebin(array1,m,n,/sample)

I compared this method using the test_arr.pro program David attached to his post. It was almost twice as fast as the previous best when array1 is large. It is slightly faster if array1 is 1xN, i.e. array1 = transpose(array1) then use:

array2 = rebin(array1,n,m,/sample)

Note that rebin can not be used with complex or string arrays. For this reason I think that I would prefer the slower indexing method which can handle all types. In all situations where I might have used this kind of replication speed is not important.

Regards,
Chris
--
==========
Bldg 24-E188
The Applied Physics Laboratory
The Johns Hopkins University
(301)953-6000 x8529
chris_chase@jhuapl.edu