
Subject: Re: Need a function to multiply the elements of an array

Posted by [chase](#) on Fri, 28 Apr 1995 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

>>>> > "Rick" == Rick White <rlw@sundog.stsci.edu> writes:

In article <3nr9e0\$2dp@sundog.stsci.edu> rlw@sundog.stsci.edu (Rick White) writes:

Rick> In article <3nm2uk\$qqq@pinot.umd.edu>, barsam@Glue.umd.edu (Barsam Marasli) writes:

Rick> |> I am looking for a routine to multiply the elements of an array.

Rick> Here is one more approach for this problem. I did some tests and found

Rick> it to be roughly the same speed as the previous approaches (using logarithms)

Rick> for short vectors (< 100 elements), but it is faster by about a factor

Rick> of 3 for long vectors. It is about 15 times faster than a direct loop

Rick> for long vectors.

It would be nice if IDL had the equivalent of the Scan operator in the APL language. This is equivalent to partial sums for a vector using an arbitrary binary operator (+, x, min, max, etc.).

A related recursive operation that I have wanted on occasion is an auto-regressive (AR) filter. IDL can easily implement moving-average filters (via convol) but it does not have an efficient function for implementing AR (or ARMA) filters. The most useful generalization of this would be implementations of multi-input, multi-output linear time-invariant (or time-varying) state space equations like those in the matlab signal processing and image processing packages.

Native functions for these types of operations would be an order of magnitude faster over IDL loop implementations.

Just wishing,

Chris

--

=====

Bldg 24-E188

The Applied Physics Laboratory

The Johns Hopkins University

Laurel, MD 20723-6099

(301)953-6000 x8529

chris.chase@jhuapl.edu
