
Subject: Re: Inner product of multi-dimensional arrays
Posted by [thompson](#) on Tue, 07 Feb 1995 15:06:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Fergus Gallagher <f.Gallagher@nerc.ac.uk> writes:

> I want to form a matrix mulipication of the form

> $C(i,j,k) = B(i,j,r) A(r,k)$ (summed over r)

> With some index cleverness, I could form a 4-D intermediate array and
> sum this over one index, but this intermediate array wouldn't fit
> into memory in my case.

> Does anyone have a suitable (fast) algorithm for this sum. Even
> better would a generalized procedure for

> $C = A(\dots r \dots) B(\dots r \dots)$

Try this:

```
DB = (SIZE(B))(1:3) ;Extract dims. Assume 3D
DA = (SIZE(A))(1:2) ; " " " 2D
B = REFORM(B, DB(0)*DB(1), DB(2), /OVERWRITE) ;Reformat into 2D array
C = B # A ;Calculate result
C = REFORM(C, DB(0), DB(1), DA(1), /OVERWRITE) ;Put into correct format
B = REFORM(B, DB(0), DB(1), DB(2), /OVERWRITE) ;Restore B to original format
```

This technique would only work if the dimension to be summed over is the last one in B and the first one in A.

I do have a routine called REARRANGE which can rearrange the order of dimensions in an array. It can found at URL

`file://umbra.gsfc.nasa.gov/pub/soho/soft/cds/util/array/rear range.pro`

It works best when supported by some CALL_EXTERNAL software written in C, which can be found at URL

`file://umbra.gsfc.nasa.gov/pub/soho/soft/cds/external/`

It is designed to operate without this C routine, but it is slower then.

Hope this helps,

Bill Thompson
