Subject: Re: Large array memory problem.
Posted by Marius Hagen on Tue, 22 Nov 2005 22:16:10 GMT
View Forum Message <> Reply to Message

Carolina wrote:
> Unfortunately, I need to
> do an array multiplication where the arrays are double precision
> complex 11000 by 11000 square arrays, and IDL will not even give me
> enough memory for a single dcomplexarr(6500,6500) array!

I often find myself working with large arrays. Usually, these are sparse, and so I can make use of sparse storage techniques to make matrix multiplication and the like tractable. However, if the matrices are not sparse, then one method I've made use of in the past is to break the matrix multiplication up into pieces. That is, rather than store the matrices in full, you can store in IDL just a single row/column vector of each of the matrices, then apply the multiplication, save the result, and loop over the next row/column. It's a little slow, but there are quite a number of operations taking place in each iteration of the loop, so it's faster than you might expect. If accessing the next row/column vector requires reading a large file over and over again, you might see the loop taking an enormous amount of time. On the other hand, if you can actually generate the matrix elements on the fly, then you can generate the row vector and column vector inside the matrix multiplication loop.

> I though IDL was the best language for large array manipulations, is it
> true? or are there other languages that are better suited for this?

Of course, C/C++ can do this stuff a lot better than IDL, but you have to deal with a lot of programming hassles and a rather unfriendly programming environment (compared to IDL) that really slows down the process of writing routines. I have some experience doing this in Matlab and Mathematica, but Matlab generally is more limited than IDL as far as speed of operation and array size limitations go. Mathematica is even more so, and can sometimes crash with surprisingly small matrices. Not to mention Mathematica's well-known tendency to corrupt datasets and program files.

- Marius