Loop Optimizations - Programs That Require More Memory Than You Have*

Charles Severance
Kevin Dowd

This work is produced by The Connexions Project and licensed under the
Creative Commons Attribution License †

People occasionally have programs whose memory size requirements are so great that the data can't fit in memory all at once. At any time, some of the data has to reside outside of main memory on secondary (usually disk) storage. These out-of- core solutions fall into two categories:

- Software-managed, out-of-core solutions
- Virtual memory–managed, out-of-core solutions

With a software-managed approach, the programmer has recognized that the problem is too big and has modified the source code to move sections of the data out to disk for retrieval at a later time. The other method depends on the computer's memory system handling the secondary storage requirements on its own, some- times at a great cost in runtime.

1 Software-Managed, Out-of-Core Solutions

Most codes with software-managed, out-of-core solutions have adjustments; you can tell the program how much memory it has to work with, and it takes care of the rest. It is important to make sure the adjustment is set correctly. Code that was tuned for a machine with limited memory could have been ported to another without taking into account the storage available. Perhaps the whole problem will fit easily.

If we are writing an out-of-core solution, the trick is to group memory references together so that they are localized. This usually occurs naturally as a side effect of partitioning, say, a matrix factorization into groups of columns. Blocking references the way we did in the previous section also corrals memory references together so you can treat them as memory "pages." Knowing when to ship them off to disk entails being closely involved with what the program is doing.

*Version 1.3: Aug 25, 2010 11:02 am -0500
†http://creativecommons.org/licenses/by/3.0/