

Re: Fortran to find nearest point from set in 3-D

Source: <http://sci.tech-archive.net/Archive/sci.math.num-analysis/2006-04/msg00202.html>

- *From:* Ronald Bruck <bruck@xxxxxxxxxxxx>
 - *Date:* Sat, 15 Apr 2006 19:51:13 -0700
-

In article <1145142971.087717.117490@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>, <David.Paterson@xxxxxxx> wrote:

I'm looking for Fortran source code if possible.

Given set A with roughly 10,000 (x,y,z) triples and set B with roughly 1,000,000 (x,y,z) triples.

For each point in set A find the nearest point from set B.

What's the quickest algorithm?

A very slow algorithm would be to check all 1,000,000 triples in B for every point in A.

A slightly better algorithm would be to sort B by (x) in advance. If I was programming it from scratch that's what I'd do.

A slightly better algorithm than that would be to allocate every point in B to a 3-D cubic grid. Find the grid node closest to each point from A and search that and the surrounding 26 grid cubes for the closest.

I seem to remember that there's an even better algorithm than that, using a heirarchy of 3-D grids.

I've not thought about the problem before (which is strange—it's an obvious proximal-point problem), but it does occur to me that once you've done one point in A, and are about to do another, MOST of B can be eliminated at a glance. Roughly, you're looking near the surface of the sphere with radius $R + \text{eps}$, where R was the last proximal distance found and eps is the distance between two points of A. "At a glance" requires more computational effort than glancing, of course, but presumably you're already computed the distances from the grid cubes to the old point, and most of the cubes which were eliminated at the last point are already eliminated for the current point.

Re: Fortran to find nearest point from set in 3-D

And if it's good enough to do this in the sup (max) norm instead of the L^2 norm... :o)

It would help tremendously if you could be sure that B or A was a discrete approximation to a convex set... Do you know anything more about the sets?

--

Ron Bruck

.