ICL-ESE-MSc Advanced Programming Coursework 2 – 2023/2024

HUANG, FAN – acse-fh223

Overview of Implemented Optimisations:

We implement an octree data structure to optimize the search task of finding the nearest node in 3D space. Compared to a brute force search directly in a dynamic array, the octree speeds up the search process by reducing the number of necessary comparisons through spatial partitioning, and the optimized algorithm has a lower time complexity than a direct traversal search. The error calculation is based on round-trip mapping. Instead of simply using a one-to-one mapping strategy during mesh mapping, we use a distance-based weighting strategy to compute the physical quantity values of the target nodes. This approach better takes into account the geometric relationships in space and can provide more accurate mapping results.

Changes to Data Structures: Optimized methods implement an octree data structure that recursively splits the 3D space into smaller regions and stores information about points within each region. This structure allows us to quickly locate the region that contains the target point or the closest point to it, thus reducing the number of comparisons required for the search. The basic concepts of nodes and elements remain unchanged, but we have changed the way they are organized in space through the octree.

Changes to Functions:

The "insert" function: To build the octree, we implement an insert function that recursively inserts points into the correct octree node based on their coordinates.

The "findNearest" function: We implement a findNearest method that uses the octree to quickly find the nearest neighbor of a given point. This method is much faster than searching directly through all the points, especially when the number of points is large.

Quantification of Performance: By using a weighted mapping instead of a simple one-to-one mapping, we are able to more accurately transfer information about physical quantities between meshes. By looking at the picture we have drawn of the error of the physical quantities, we can see that the error of the optimized interpolation method is greatly reduced and relatively stable, and does not fluctuate significantly with the number of nodes. And, using the octree search method, we achieve a significant performance improvement in finding the nearest node, consuming significantly less time. Moreover, as the number of nodes increases, the search time using octree grows much slower than direct search.