Hierarchical Graph Traversal for Aggregate k Nearest Neighbors Search in Road Networks

Tenindra Abeywickrama*, Muhammad Aamir Cheema^, Sabine Storandt#
*NUS-Grab AI Lab, Monash University^, University of Konstanz#

Key Points:
- Uses network distance in graph (more diverse + accurate metrics)
- Answered efficiently by using heuristics to retrieve candidates
- But existing heuristics only suitable for single agent
- Propose a new data structure to answer AkNN
- Significantly faster in practice but still lightweight

Key Benefits:
- Novel use of landmark lower-bounds (LLBs) for greater accuracy
- LLBs never previously used for hierarchical search until COLT!
- Landmarks are localized for each tree level => allows us to pinpoint best candidates
- Displays interesting property that makes it particularly efficient for AkNN search for convexity preserving aggregate functions
- Common functions such as SUM and MAX do preserve convexity
- Space/time complexity for pre-processing comes at modest cost in both theory and practice: still lightweight
- It is a generic graph data structure that can potentially be applied to other graphs and problems, e.g., shortest detour query

Experimental Results
- Experiments were conducted on real-world road network and POI datasets for United States
- COLT up to an order of magnitude faster than existing techniques!
- Performs better on dense POI sets because it better distinguishes close objects => i.e. a better heuristic
- COLT maintains improvement for varying parameters such as: number of results k, number of agents |Q|, the aggregate function, and machine independent heuristic efficiency
- Comes at small and/or comparable pre-processing cost in time and space

Top-Down Recursive Search
We find AkNN candidates by conducting a top-down search from the root node in the tree. Each child represents a subgraph of the parent. COLT stores certain values in each tree node that allow a lower-bound aggregate score to be compute for all objects within the subgraph using the equation to the right. Children with the best aggregate score are evaluated recursively until the best candidate object is found.