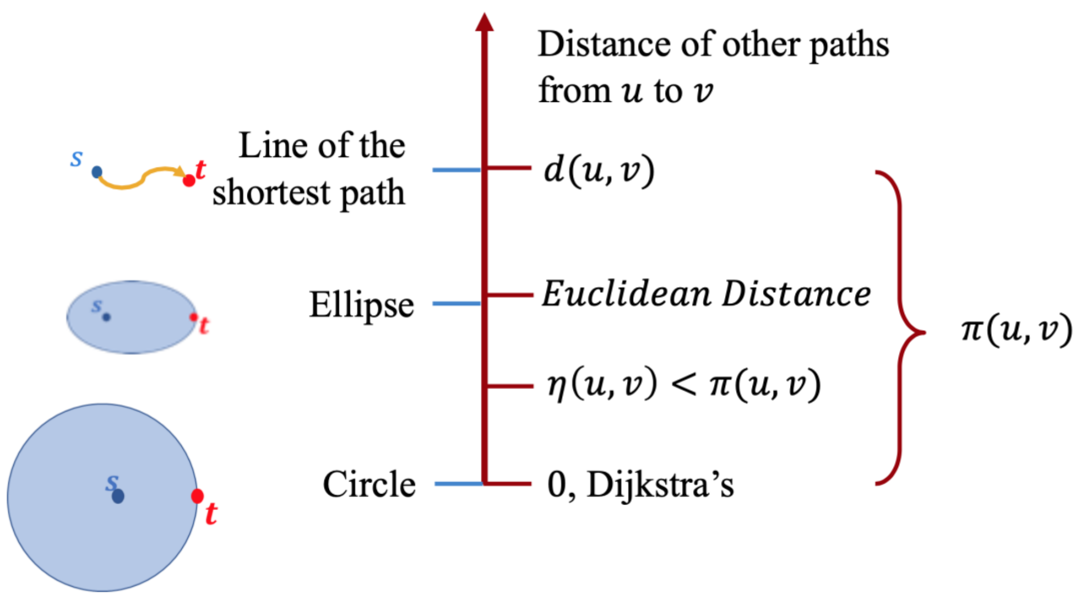
Spatiotemporal Data Management and Analytics

Question Set Day 4: Route Planning in Road Network

Semester 3, 2020

Question 1: The performance of the distance-estimation based heuristic search algorithms (A^* and Landmark) are highly affected by the quality of the estimation. Please discuss how the estimated distance influences the algorithm performance.

Sample Solution:
In the distance-estimation based heuristic search, the key in the priority queue is $d(s,u) + \pi(u,t)$ instead of $d(s,u)$. As long as $\pi(u,t) \leq d(u,t)$, the correctness can be guaranteed.



As for the influence of $\pi(u,t)$, when it is 0, the search space reduces to the same as the Dijkstra's (a circle). When $\pi(u,t) = d(u,t)$, the search space is just the shortest path itself (a line). When $0 \leq \pi(u,t) \leq d(u,t)$, the search space is in between, like an ellipse. When $\pi(u,t)$ is closer to 0, the ellipse is larger; when (u,t) is closer to $d(u,t)$, the ellipse is smaller.

Question 2: The Dijkstra's algorithm is essentially a single-source shortest path algorithm that can be used to answer the point-to-point path query. On the contrary, the A^* algorithm is essentially a point-to-point shortest path algorithm because its searching heuristic is towards a destination. Can A^* also be used as a single-source algorithm? If so, how does it work?

Sample Solution:
Yes, A^* can be used to find the distances from the source to all the other vertices in the graph.

The procedure is the same as the Dijkstra's: When a point pops out of the priority queue with $d(s,u) + \pi(u,t)$, this $d(s,u)$ is the shortest distance from s to u. We can run this procedure on and on, even passes t. The algorithm terminates when the priority queue becomes empty. In this way, we can find the distance from s to all the other vertices.

The proof is not required by this course.

Question 3: During the construction of CH, the shortcuts we add have the actual shortest distances between the vertices. However, this requires a large amount of Dijkstra's search to guarantee this property. If for any neighbour pair (v, w) of u, where v and w are contracted later than u, we add a shortcut (v, w) with distance $\min(d(v,u) + d(u,w), d(v,w))$ ($d(v,w) = \infty$ if there is no edge or shortcut between v and w), then the contraction process will become much faster. Can this method also answer the query correctly? What are the advantages and the disadvantages of this method compared with the classic CH?

Sample Solution:
Yes, the correctness is guaranteed. Because we still need to run the Bi-directional upwardly search, the Dijkstra's search is in a way postponed to the query answering stage. When we contract a vertex v, the standard CH adds the shortcut only when $u \to v \to w$ is the shortest path from u to w, while the new method is much looser and creates more shortcuts even when $u \to v \to w$ is not the shortest path. This redundant information guarantees the correctness of the latter contractions.

This approach has the advantage of fast construction because no Dijkstra is needed. But it takes longer time to answer a query than the standard CH because it has a much larger index size. In fact, its performance is near to A^*.

Question 4: Pruned Landmark Labeling is an easy way to construct the 2 Hop labels. However, it is more suitable for the $small\ world$ graph than the road network. The small world graph is a kind of graph that is composed of several densely connected communities, and the connections between the communities are very limited. Please discuss how the pruned landmark labeling can benefit from this property.

Sample Solution:
The small world graph has a property that small number of nodes have a very large number of degrees such that most of the remaining nodes connect to them. Therefore, these nodes are the natural hub of the graph. The shortest paths among other vertices have a very high chance to pass through them. Then if we run the pruned landmark labeling in the degree decreasing order, these important nodes would become labels in other nodes earlier, and they can help prune the search space a lot.