**1.2何智鹏-Workload-Aware Shortest Path Distance Querying in Road Networks**

**【 】**

**一、总结做法【 】**

正最短路径距离查询是指给定一个网络上的起点和终点，要求返回这两点间的最短路径距离。其作为一项基本操作，是许多基于位置服务的应用中的基础构建模块，例如GPS导航、打车服务、物流运输、自动驾驶等。随着移动互联网的发展，大量的并发查询请求在不同时间段内组成不同的查询负载，给服务器资源带来了巨大的压力。如何在尽可能小的资源占用的情况下，快速高效地处理最短路径查询负载，面临巨大挑战。为了能够有效地计算这样的距离查询，现有的方法频繁应用2-hop标记，即为每个顶点构造一个标签，并通过只对标签执行线性扫描来计算查询。然而，很少有方法考虑到空间查询工作负载的主要特征。作者观察到，现实世界的工作负载表现出空间倾斜的特征，这意味着只有一小部分顶点被频繁查询。

根据上述观察，作者首先分别修改了目前最先进的算法：PLL和H2H，开发了一种工作负载感知的修剪地标标记（wPLL）方法和一种工作负载感知的分层2-hop（wH2H）方法。虽然这些方法由于高频顶点的标签尺寸较小，因此能够实现更有效的查询，但它们增加了空间消耗和预处理时间。

为了克服这些限制，作者开发了一个工作负载感知的核心-森林标签索引结构（WCF），它利用了wPLL和wH2H的优势。WCF维护了一个由两层组成的核心-森林结构。上层是通过考虑拓扑中心性和查询频率，在高频顶点的覆盖图上构建wPLL的核心。底层采用树分解，从道路网络中获得一组低频顶点的子图。通过为每个子图构建一个树形索引来构建一个森林。

**二、改进方法【 】**

作者对目前State-of-art的两个算法：PLL和H2H进行了改进和优化。

对于wPLL，作者将原来PLL算法节点排序这一步骤，由此前根据结构特征例如度大小、路径重要性、中介中心度等改为根据路网结构特征和查询负载特征综合考量。这样使得高频点被更早处理，减小索引大小。

对于wH2H，基于树结构的最短距离查询方法，将原来H2H按照节点度数进行节点删除，并通过权值更新和填边保证删除点后剩余图的正确性，构建完树结构后按照从上到下的方式计算每个节点的标签索引，然后利用标签索引进行最短路径查询，论文中基于此改进的wH2H算法将查询频率纳入考量，使得形成的树结构中高频点更靠近根节点，减小高频点查询开销。

但是wPLL和wH2H有各自的优缺点，wPLL适合复杂且度数高的图，查询快但是预处理比较慢；wH2H得益于树分解的优势，构建标签速度更快。因此作者结合两者优势，提出了WCF算法。

基于层次结构的混合方法WCF综合利用查询频率和路网结构提取出层次结构，在高层结构上利用基于节点排序的方法来减小高频查询点的索引大小，从而减小总的查询开销，然后在包含大部分低频查询点的低层次结构上利用基于树分解的方法来减小索引构建开销。