

SGU 314. Shortest Paths ^{*†}

张晴川

qzha536@aucklanduni.ac.nz

December 13, 2020

大意

给一个 N 个点 M 条边的正权有向图和起点 s 、终点 t ，求前 K 短路。

注：不需要是简单路径，只需起点和终点符合即可。

数据范围

- $1 \leq N \leq 10000$
- $1 \leq M \leq 50000$
- $1 \leq K \leq 10000$
- $s \neq t$

题解

首先把图反向，构造以 t 为原点的最短路树，预处理出每个点 i 到终点的最短距离 $dis[i]$ ，我们把每个点到 t 的树上路径称为它的**默认路径**。

现在把所有无法到达终点的点以及与之有关的边删除。

现在将邻接表的每条边 $(u \rightarrow v, len)$ 按照所导致的最小额外代价 $(len + dis[v] - dis[u])$ 排序。不难发现可以认为每个边表中的第一条边构成了最短路树（除了 t 之外）。

考虑二分答案，转化为有多少条路径小于等于某个 $bound$ 。

首先排除 $dis[s] > bound$ 的情况。

我们使用 BFS 把所有可行解搜出来。每个状态为二元组 (v, len) ，表示一条从起点到 v ，长度为 len 的路径。初始状态为 $(s, 0)$ 。

^{*}<https://codeforces.com/problemsets/acmsguru/problem/99999/314>

[†]更多内容请访问：<https://github.com/SamZhangQingChuan/Editorials>

考虑如何扩展一个状态 (v, len) 。如果存在某条 v 的非树边使得 len 加上最小额外代价后依然满足 $bound$ 的要求，那么解数至少会加一，复杂度是被 K 限制住的。但是也有可能不存在可行的非树边，即只能沿着默认路径走。这样会导致最差情况下向前走 $N - 1$ 步才能走出一个新解（考虑一个大圆环）。

考虑在树上倍增，就可以用 $\log(N)$ 的代价求出从 v 到 t 的默认路径上第一个存在额外代价足够小的非树边的祖先。当然，如果一直到 t 都不存在，那么可以直接加上 $dis[v]$ 后退出。

搜索出 K 个解后直接退出即可。

复杂度

- 时间： $O(\log(N \cdot M \cdot \text{Max Weight}) \times K \log(N))$
- 空间： $O(N \log(N))$

代码

<https://gist.github.com/SamZhangQingChuan/94ffc666bb3233020df113c879d9f2fe>