一道最短路问题

成都市第七中学

*zms_*

March 11, 2017

今天给大家分享一道简单的最短路问题
题目来源是$bzoj$2069

Description

n个点m条边的无向图，每条边有正权值。
现在要求一条从1号点出发又回到1号点的最短回路，且满足至少经过两个点并且经过的边不能重复。
$n, m \leq 10^5$

这道题的问题主要在于不走重复的边。
要如何构造一个新图跑最短路来避免这个问题?

今天准备给大家分享两种做法。

先考虑一种暴力的做法，我们枚举路径上除去1以外第一个点x和最后一个点y
那么只要x, y互不相同，$w(1, x) + dis(x, y) + w(y, 1)$就可以用来更新答案。(因为都是正权，所以一定没有走重复的边)

先考虑一种暴力的做法，我们枚举路径上除去1以外第一个点x和最后一个点y
那么只要x, y互不相同，$w(1, x) + dis(x, y) + w(y, 1)$就可以用来更新答案。(因为都是正权，所以一定没有走重复的边)
也就是说我们在与1相邻的点集中求最近点对即可。
显然直接枚举是不行的。怎么办?

我们其实并不关心x, y具体的值，只要知道所有点对中的最小距离。
考虑将点集划分为两个集合s_1, s_2，将s_2与1的边去掉，以1为起点跑一次最短路算法。再用s_2中每个点的距离加上它到1的边权来更新答案。
为了保证求到最优解，只做一次显然是不行的，我们需要做多次使得每对点都被划分到不同的集合中过。

如何划分集合呢?我们可以二进制分组。
具体来说，一共要做$logn$次，第i次划分集合的依据是每个点编号在二进制中第i位的值(0/1)。
因为x, y这两个编号不相等，所以一定有某一位它们是不同的。
那显然就包含了所有的情况。
这个做法需要跑$O(logn)$次单源最短路

当然，我们还有一个只需要跑常数次最短路的做法。
下面这个做法需要利用到最短路树。

首先显然，最终答案路径一定是先走了一段最短路，再走了一些边，最后沿着最短路回到起点的。
如果经过的非最短路边不连续，是可以把其中一部分变为走最短路，使得只有一段连续的非最短路边。

首先显然，最终答案路径一定是先走了一段最短路，再走了一些边，最后沿着最短路回到起点的。
如果经过的非最短路边不连续，是可以把其中一部分变为走最短路，使得只有一段连续的非最短路边。
但是，这样没法判定两次走最短路时有重边的问题。

考虑以1为起点建出最短路树，每个点是在1的哪一个子树里(即最短路上除起点外经过的第一个点)记为$belong[i]$
答案的形式是$dis(1,x)+dis(x,y)+dis(y,1)$
满足$belong[x]!=belong[y]$
接下来，我们需要将图重构一下。

我们分类讨论每条有向边$u->v$权值为w

我们分类讨论每条有向边$u->v$权值为w
如果$u==1\&\&belong[v]==v$ 不连这条边

我们分类讨论每条有向边$u->v$权值为w
如果$u==1\&\&belong[v]==v$ 不连这条边
如果$v==1\&\&belong[v]!=v$ 用$dis[v]+w$更新答案

我们分类讨论每条有向边$u->v$权值为w
如果$u == 1 \&\& belong[v] == v$ 不连这条边
如果$v == 1 \&\& belong[v] != v$ 用$dis[v]+w$更新答案
如果$v == 1 \&\& belong[v] == v$ 连$(u->t, w)$,t是新建的终点

我们分类讨论每条有向边$u->v$权值为w
如果$u==1\&\&belong[v]==v$ 不连这条边
如果$v==1\&\&belong[v]!=v$ 用$dis[v]+w$更新答案
如果$v==1\&\&belong[v]==v$ 连$(u->t,w)$,t是新建的终点
如果$belong[u]!=belong[v]$ 连$(1->v,dis[u]+w)$，表示强制走了这条边,此时已经从一个子树到达另一个子树

我们分类讨论每条有向边$u->v$权值为w
如果$u==1\&\&belong[v]==v$ 不连这条边
如果$v==1\&\&belong[v]!=v$ 用$dis[v]+w$更新答案
如果$v==1\&\&belong[v]==v$ 连$(u->t,w)$,t是新建的终点
如果$belong[u]!=belong[v]$ 连$(1->v,dis[u]+w)$，表示强制走了这条边,此时已经从一个子树到达另一个子树
不属于上面几种情况的连原边。最后计算$1->t$的最短路即可。

这个算法利用最短路树的信息重构图，强制起点到终点一定经过了不同的子树，解决了走重边的问题。并且只用两次单源最短路算法，时间是比较优秀的。

昨天我校最强选手告诉了我一个很妙的做法，现在让他来分享一下

今天给大家分享了一道利用最短路树重构图的题，这里还有一道想法类似的问题，大家可以下来看一看。
题号是$bzoj4283$

谢谢大家