

《tourbus》解题报告

一、题目大意

给定一个N个点的无向平面图，要求用最少的合法路径覆盖它的每一条边恰好一次。

一条合法路径或者是一条简单路径，或者是一个环，并且不能自交（平面图意义）。

二、初步分析

这是一道challenge题，很显然此题是一个NP-Hard问题，所以我们着眼于找到一个近似方法/部分方法。

针对这种情况常用的方法有搜索、贪心、DP（错误DP）等等或它们的综合，无论使用哪种方法，都建议自己写一个check验证答案/统计最优性。

我们可以发现，一个朴素的解是尽量连上相邻的一对边，在平面图上相邻的两条边显然不会相交。

可以得知这样需要 $(N+M)/2$ 条路径。

三、小范围算法讨论

然而我们仍然对解题并没有太多头绪，所以尝试考虑缩小范围。

比如如果边数较小，我们可以采用状压DP的方法获得答案。

第一步，生成所有合法路径，标记为这些路径所对应bit的或。

第二步，对于任意边集S，通过子集枚举的方法dp出覆盖S上的每条边恰好一次所需的最少路径数。

子集枚举的时间复杂度是 $O(3^M)$ 的，可以处理 $M \leq 15$ 的情况。

伪代码：

```
f[0]=0;
for (int S=1;S<2^M;S++){
    f[S]=+inf;
    for (sub=S;sub>0;sub=S&(sub-1)){
        if (S/sub is valid){
            f[S]=min(f[sub]+1,f[S]);
        }
    }
}
```

四、较大范围

对于较大的范围，我们不妨先采取贪心的策略。

例如，不断找到最长的路径并删除。

但是我们发现，找到最长路径仍然是一个NP-Hard问题。

我们尝试使用一个启发式方法：不断加长当前路径。

我们尝试加入一条边之后要与已有的边判交（平面图意义），如果不能通过，也可以考察加入这条边之后哪些边不可行了，根据拓扑关系顺列边。

我们还可以随机断掉一条边或者翻转路径加边。

五、算法优化

朴素的搜索算法不断拓展当前路径。

然而每次的所有合法拓展可能有优劣之分，我们尝试贪心地在每次拓展时找到“最优”的边。

我们混合使用搜索和贪心，或者在“弹性大”的地方使用搜索确定大分支。

最后，我们可以执行算法多次，通过加入随机化元素与卡时获得更优解。

在特定的数据下，也可对不同数据采用不同的启发式算法。

因为用到了卡时，所以我们可以对算法进行一些精细的常数优化（可以发现这个算法的时间瓶颈在于判断线段是否相交，可以用快速排斥方法优化），优化后程序可以执行更多次，获得更优解。

因为卡时所以时间复杂度为时限。

空间复杂度看具体使用的启发式方法，大致为 $O(N+M)$ 。