

# 有向无环图（DAG）的最小路径覆盖

## DAG的最小路径覆盖

**定义：**在一个有向图中，找出最少的路径，使得这些路径经过了所有的点。

最小路径覆盖分为**最小不相交路径覆盖**和**最小可相交路径覆盖**。

**最小不相交路径覆盖：**每一条路径经过的顶点各不相同。如图，其最小路径覆盖数为3。即1->3>4，2，5。

**最小可相交路径覆盖：**每一条路径经过的顶点可以相同。如果其最小路径覆盖数为2。即1->3->4，2->3>5。

特别的，每个点自己也可以称为是路径覆盖，只不过路径的长度是0。

## DAG的最小不相交路径覆盖

**算法：**把原图的每个点 $V$ 拆成 $V_x$ 和 $V_y$ 两个点，如果有一条有向边 $A \rightarrow B$ ，那么就加边 $A_x \rightarrow B_y$ 。这样就得到了一个二分图。那么最小路径覆盖=原图的结点数-新图的最大匹配数。

**证明：**一开始每个点都是独立的为一条路径，总共有 $n$ 条不相交路径。我们每次在二分图里找一条匹配边就相当于把两条路径合成了一条路径，也就相当于路径数减少了1。所以找到了几条匹配边，路径数就减少了多少。所以有最小路径覆盖=原图的结点数-新图的最大匹配数。

因为路径之间不能有公共点，所以加的边之间也不能有公共点，这就是匹配的定义。

**习题：**[POJ1422](#)

```
//
//  main.cpp
//  POJ1422最小不想交路径覆盖
//
//  Created by beMaster on 16/4/8.
//  Copyright © 2016年 beMaster. All rights reserved.
//

#include <iostream>
#include <stdio.h>
#include <string.h>
#include <vector>
using namespace std;
const int N = 200 + 10;
vector<int> g[N];
int cy[N];
bool vis[N];
bool dfs(int u){
    for(int i=0; i<g[u].size(); ++i){
        int v = g[u][i];
```

### 公告

昵称： justPassBy  
园龄： 4年11个月  
粉丝： 4  
关注： 3  
[+加关注](#)

<	2019年9			
日	一	二	三	
1	2	3	4	
8	9	10	11	
15	16	17	18	
22	23	24	25	
29	30	1	2	
6	7	8	9	

### 搜索

### 积分与排名

积分 - 45228

排名 - 14226

### 随笔分类

ACdream(1)

BestCoder(8)

codeforces(6)

data structure(14)

dp(56)

```
        if(vis[v]) continue;
        vis[v] = true;
        if(cy[v]==-1 || dfs(cy[v])){
            cy[v] = u;
            return true;
        }
    }
    return false;
}
int solve(int n){
    int ret = 0;
    memset(cy, -1, sizeof(cy));
    for(int i=1;i<=n;++i){
        memset(vis, 0, sizeof(vis));
        ret += dfs(i);
    }
    return n - ret;
}
int main(int argc, const char * argv[]) {
    int t,n,m;
    int u,v;
    scanf("%d",&t);
    while(t--){
        scanf("%d%d",&n,&m);
        for(int i=1;i<=n;++i)
            g[i].clear();
        for(int i=0;i<m;++i){
            scanf("%d%d",&u,&v);
            g[u].push_back(v);
        }

        int ans = solve(n);
        printf("%d\n",ans);
    }
    return 0;
}
```



DAG的最小可相交路径覆盖

**算法：**先用floyd求出原图的传递闭包，即如果a到b有路径，那么就加边a->b。然后就转化成了最小不相交路径覆盖问题。

**证明：**为了连通两个点，某条路径可能经过其它路径的中间点。比如1->3->4，2->4->5。但是如果两个点a和b是连通的，只不过中间需要经过其它的点，那么可以在这两个点之间加边，那么a就可以直达b，不必经过中点的，那么就转化成了最小不相交路径覆盖。

**题目：**[POJ2594](#)



```
//
//  main.cpp
//  POJ2594最小可相交路径覆盖
//
//  Created by beMaster on 16/4/8.
//  Copyright © 2016年 beMaster. All rights reserved.
//

#include <iostream>
#include <stdio.h>
#include <string.h>
#include <vector>
using namespace std;
const int N = 500 + 10;
bool dis[N][N];
bool vis[N];
int cy[N];
void floyd(int n){
    for(int i=1;i<=n;++i)
        for(int j=1;j<=n;++j)
            for(int k=1;k<=n;++k)
                if(dis[i][k] && dis[k][j])//传递可达性
                    dis[i][j] = true;
}
```

dp-概率dp(6)

dp-树形dp(6)

dp-状态dp(7)

game theory(4)

graph(44)

graph-生成树(3)

graph-最短路(10)

hdu多校(1)

hustOj

Java-I/O(3)

javaweb(1)

linear algebra(4)

math(18)

search(3)

string(15)

UVA(4)

并查集(1)

操作系统(1)

差分约束系统(2)

分治(1)

几何(1)

技巧(4)

连通性(4)

树的点分治(3)

树链剖分(4)

数学(12)

```

}

bool dfs(int u, int n){
    for(int i=1;i<=n;++i){
        if(!vis[i] && dis[u][i]){
            vis[i] = true;
            if(cy[i]==-1 || dfs(cy[i], n)){
                cy[i] = u;
                return true;
            }
        }
    }
    return false;
}

int solve(int n){
    int cnt = 0;
    memset(cy,-1,sizeof(cy));
    for(int i=1;i<=n;++i){
        memset(vis,0,sizeof(vis));
        cnt += dfs(i, n);
    }
    return n - cnt;
}

int main(int argc, const char * argv[]) {
    int n,m;
    int a,b;
    while(scanf("%d%d",&n,&m),n+m){
        for(int i=1;i<=n;++i)
            for(int j=1;j<=n;++j)
                dis[i][j] = false;
        for(int i=1;i<=m;++i){
            scanf("%d%d",&a,&b);
            dis[a][b] = true;
        }
        floyd(n);
        int ans = solve(n);
        printf("%d\n",ans);
    }
    return 0;
}
```



参考：

**二分图大讲堂——彻底搞定最大匹配数（最小覆盖数）、最大独立数、最小路径覆盖、带权最优匹配**

分类： graph

好文要顶 关注我 收藏该文



 justPassBy

关注 - 3

粉丝 - 4

+加关注

« 上一篇： Java多线程使用场景  
» 下一篇： 缓存策略

posted @ 2016-04-08 20:50 justPassBy 阅读(8376) 评论(1) 编辑 收藏

评论列表

#1楼 2018-10-26 15:14 俺是小程

图挂了

支持(0) 反对(0)

搜索(1)

网络流(1)

随笔档案

2016年9月(1)

2016年5月(1)

2016年4月(8)

2016年3月(5)

2016年2月(5)

2015年12月(1)

2015年11月(3)

2015年10月(9)

2015年9月(19)

2015年8月(19)

2015年7月(16)

2015年6月(13)

2015年5月(20)

2015年4月(21)

2015年3月(12)

2015年2月(18)

2015年1月(1)

2014年11月(10)

2014年10月(9)

2014年9月(12)

最新评论

1. Re:有向无环图（DAG）的最小路径覆盖
- 图挂了

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)， [访问](#) [网站首页](#)。

- 【推荐】超50万C++/C#源码：大型实时仿真组态图形源码
- 【活动】阿里云910会员节多款云产品满减活动火热进行中
- 【推荐】新手上天翼云，数十款云产品、新一代主机0元体验
- 【推荐】零基础轻松玩转华为云产品，获赠礼加返百元大礼
- 【推荐】华为IoT平台开发者套餐9.9元起，购买即送免费课程

相关博文：

- 有向无环图（DAG）的最小路径覆盖
- 有向无环图（DAG）的最小路径覆盖
- 有向无环图（DAG）的最小路径覆盖
- HDU3861TheKing’sProblem最小路径覆盖（强连通分量缩点+二分图最大匹配）
- 二分图基础整理

最新 IT 新闻：

- 华为已获得50多份5G商用合同，5G基站发货超20万个
  - 获800亿日元投资后，JDI将建OLED工厂，但两年后才能量产
  - 搭载高通骁龙855移动平台的三星Galaxy A90 5G现已正式发布
  - 历经30多年的努力，科学家终于得到了另一种高温超导材料
  - 偿还30亿美元债务 退任CEO 贾跃亭宣布FF重大消息
- » 更多新闻...

2. Re:hdu1754(splay

初学splay树，有两个问题可以直接通过kth()方法进行更新，不调节其到r有，为什么kth()方法每开始寻找目标节点？是更

阅读排行榜

1. 有向无环图（DAG）(8376)
2. Java多线程使用场景
3. 阶梯博弈（历届试题4）
4. JDBC与反射(1846)
5. 插板法（排列组合）

评论排行榜

1. 有向无环图（DAG）(1)
2. hdu1754(splay)(1

推荐排行榜

1. 有向无环图（DAG）(5)
2. 集线器，交换机，路
3. J2EE如何生成验证码验证码(2)
4. Java多线程使用场景
5. 询问任意区间的minum,xor,or,and(1)