浅谈二分匹配

tiankonguse 2013/05/06

之前学过二分匹配，最近重新学习二分匹配后，做了一道二分匹配题后对二分匹配有所感想。

因为对于二分匹配的建图有了新的看法，于是想深入探讨一下怎么建图。

对于 poj 的 Antenna Placement 题（传送门 http://poj.org/problem?id=3020），由于是较基础的二分匹配，所以很多人随意的敲代码，随意的 ac 了。

可以先看看这个题的大概题意，这里不介绍了。

但是我在建图时遇到一个问题，我有两种建图的方法，我不知道应该选择那个。

由于二分图的边是点的关系，很容易想到用两个是'*'的坐标是否相邻来建立边的关系。

第一步是找到所有的点，对其标号，总共 n 个。
第二步是建立点与点的关系，这样就建了一个无向图了。
第三步是把每个点拆成两个点，建一个二分部。
第四步求出最大二分匹配 max，那么答案就是 n - max/2.

答案为什么是这个样子呢？

很多人都是很显然的说是点变为了原来的两倍，匹配的边也为了原来的两倍，因此要除以 2.

虽然对于这道题这样做 ac 了，但是理由却说的很牵强。

说一句话要有根有据的。

我们暂且不证明这，先来看看另一个建图的方法。

由于是坐标，而且只会和上下左右有关系，
所以我可以根据坐标的奇偶性来直接建二分图。

把(x+y)是奇数的点放上面，是偶数的点放下面，则上面的点之间一定没有关系，下面的也是。

所以可以直接建一个二分图,上面有 n1 个点，下面有 n2 个点。

此时求出的二分匹配是什么呢？

很容易理解，就是可以圈出的最多的点对 max，那么剩下的点只能单独一个存在了。
答案就是 n1 + n2 - max .

现在来看看为什么第一种建图的方法得到的答案是正确的。

第一个方法建的图上面 n 个点，下面 n 个点。
表面上看不出什么特点，但是仔细看的话可以看出上面的偶点只与下面的奇点有边，上面的奇点只与下面的偶边有关系。

那我们把点标记一下颜色，上面的奇点与下面的偶边为有黑色，反之为白色，你会发现黑色的点与白色的点之间没有边，于是可以分成两个二分部。

再仔细看的话，你会发现这两个着色的二分图其实是完全一样的。

那么答案就很容易理解了。

现在反过来猜想，如果点与点的关系不是基于上下左右改怎么处理呢？

如果斜着也有关系呢？

我们就不能用第二种方法建图了，斜着话，奇点与奇点是有关系的。

但是我们也已用第一种方法先建一个无向图还是很容易办到的，我们把建的图称为 G1。

此时我们的需求是尽量多的挑一些边，使得任意两个边之间的顶点不同。

这在图中称为最大边独立集。

那怎么求最大边独立集呢？

很多人会随口说二分匹配。

二分匹配虽然强大，但是并不是什么都可以用二分匹配的，使用二分匹配是有条件的。

那就是能够拆分成一个二分图。

有人说把一个点拆成两个点不就成二分图了吗？

其实并不是这样的。

如果这样能够解决的话，也就不存在那些所谓的 NP 问题了。

那我们来一步一步分析怎么解决最大边独立集吧。

假设我们选了一条边<a,b>,与顶点 a 有关联的边我们都不可以选了。
假设存在<a,c>，则<a,b>与<a,c>两条边有关系，关系是存在公共顶点，我们只能选择一个边。
我们把边变为顶点，顶点分解为边的关系，也就是边，则可以建一个图，我们称为 G2.

对于 G2,我们的需求是挑若干顶点，使得他们之间没有边。

这称为什么呢？

不就是最大独立集嘛!

那怎么求解最大独立集呢？

不要给我说二分匹配。

明说了吧，二分匹配解决不了这个问题。

那现在我再次对这个图进行分析。

我们的需求是挑若干顶点，使得他们之间没有边。

假设我们对这个图求反，也就是补图，那问题变成了什么呢？

挑若干顶点，使得任意两个顶点之间有边。

这又称为什么呢？

最大团吧！

那怎么求最大团呢？

求之前先了解什么是最大团吧！

waiting...

了解了最大团，你就知道：对于最大团，目前的算法是没有多项式的求解方法的。

你还能想用二分匹配做吗？

说到这是不是很失望？

当你查阅资料后你是不是更失望了？

需要解决这个问题用到的算法竟然有遗传算法、禁忌算法、模拟退火算法、神经网络。

简而言之是搜索，有效的搜索，剪枝的搜索，智能的搜索。

当然，你还会看到一点曙光，可以写个 dfs 来搜索，这个自己倒是可以写出来，虽然跑的可能慢了点。

好了，不多说了。

很多事自己多思考一下会得到很多意想不到的东西的。