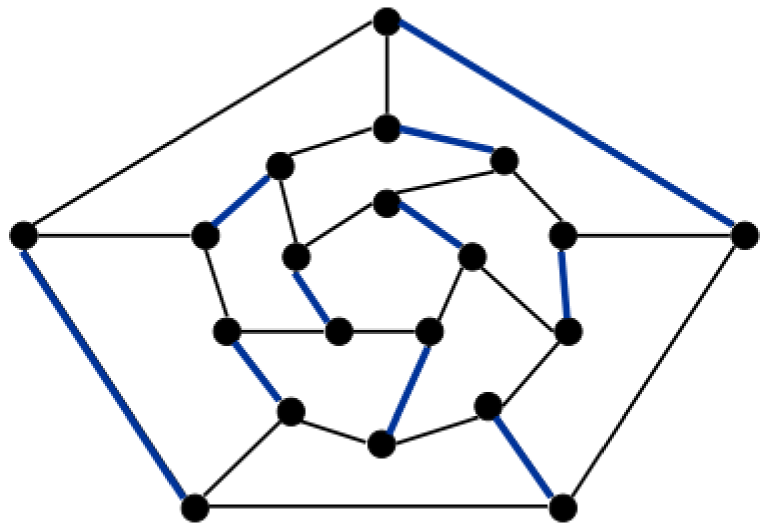
关于匹配的几个定义

匹配（Matching）

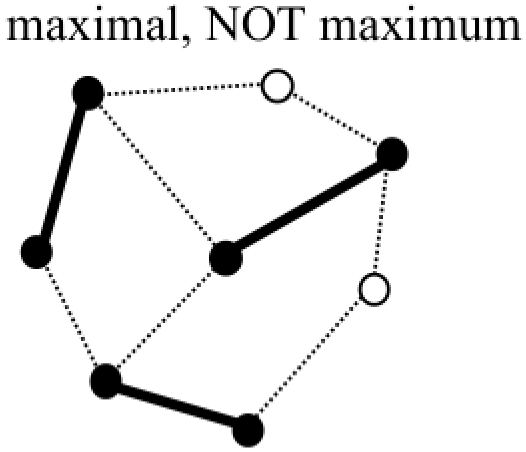
一个匹配是一个边的子集合$M \subseteq E$，且满足对所有顶点$v \in V$，M中至多有一条边与v相关联。也可以简单地说，一个匹配就是一个边的集合，其中任意两条边之间都没有公共顶点。

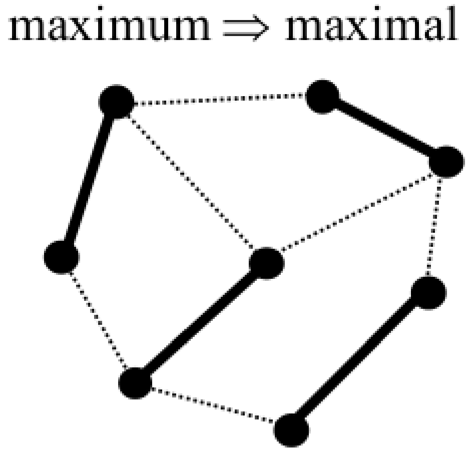
下面给出一个例子：



最大匹配（Maximum Matching）

简单地说，最大匹配是一个图的所有匹配中边数最多的那个匹配。给出一个例子：



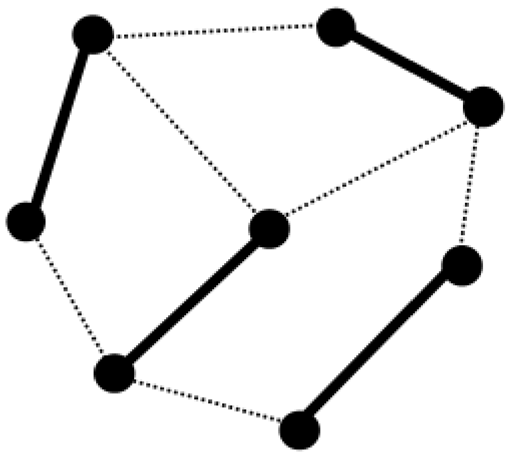


二分图（Bipartite Graph）

二分图是图论中的一种特殊模型，设$G=(V,E)$是一个无向图，如果顶点V可分割为两个互不相交的子集(A,B)，并且图中的每条边$(i，j)$所关联的两个顶点i和j分别属于这两个不同的顶点集，则称图G为一个二分图。

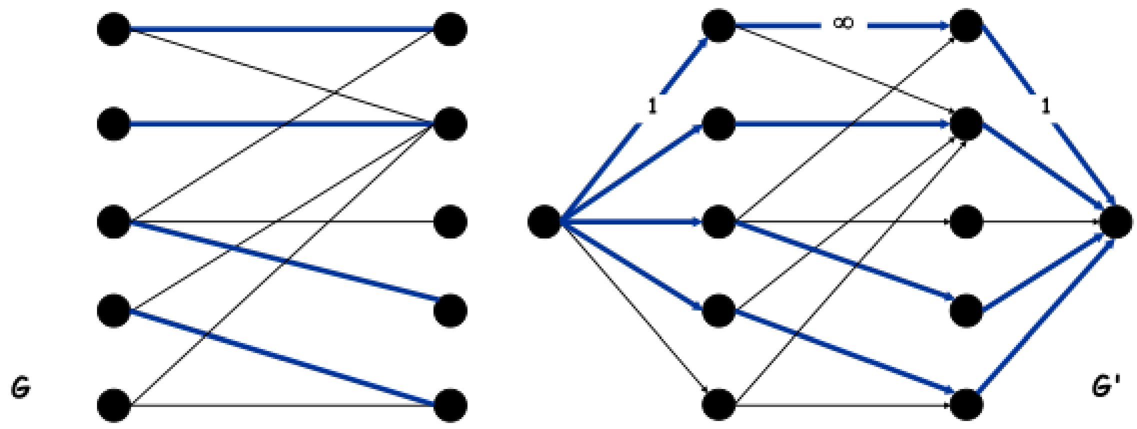
完全匹配（Perfect Matching）

简单地说，当一个图的某个匹配中所有的顶点都是匹配点，那么这个匹配就是完美匹配。同样给出一个例子：



在二分图中寻找最大匹配

从本质上来说，二分图匹配其实是最大流的一种特殊情况。是解决这个问题的关键技巧在于建立一个流网络，其中流对应于匹配，如下图所示。

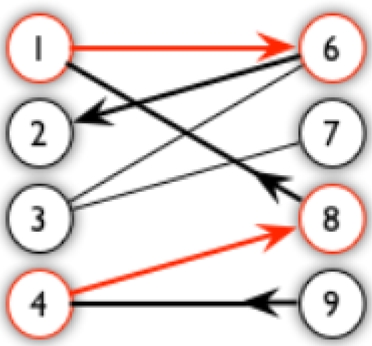


可以看出，图中添加了源点s和汇点t，它们是不属于V的新顶点。令已有边的容量为无穷大，且令s和t分别连接二分图，并设置其容量为1。这时，我们通过Ford-Fulkerson方法计算得到的最大流就等于最大二分匹配。

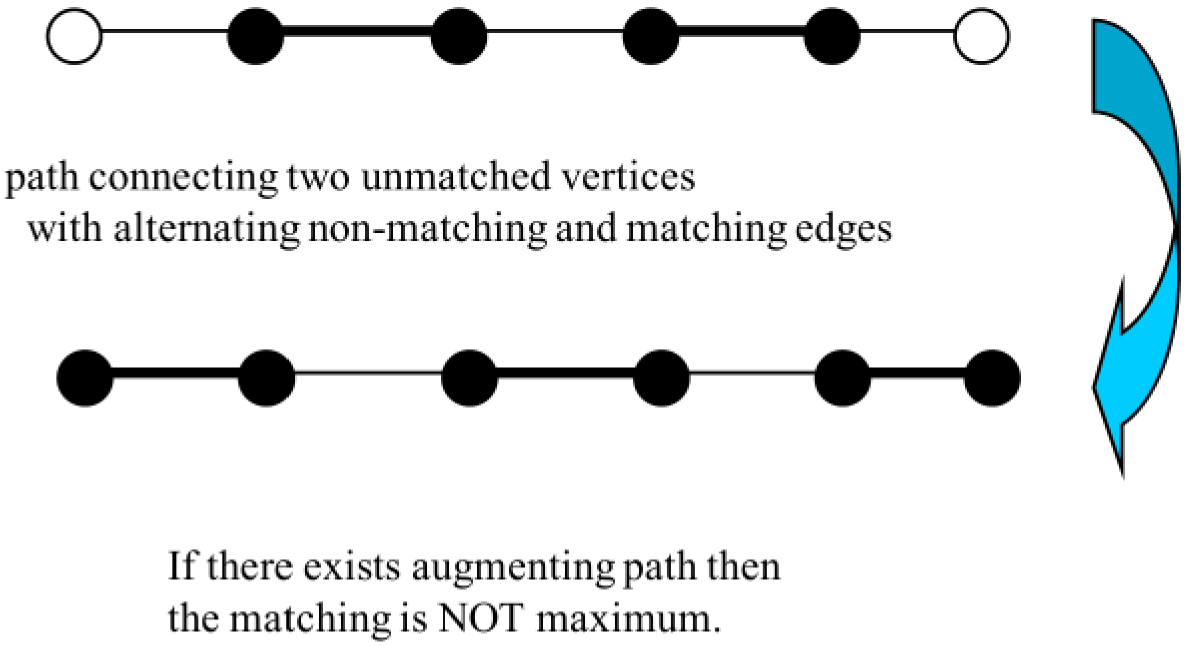
在一般图中寻找最大匹配

在一般图中，我们使用**增广路径（Agumenting path）**来寻找最大匹配。如果一条路径的首尾是非匹配点，路径中除此之外（如果有）其他的点均是匹配点，那么这条路径就是一条**增广路径**。

如下图所示，我们从非匹配点9出发，经过匹配点4、8、1、6，最后在非匹配点2停止。所以，9->4->8->1->6->2 就是一条增广路径。



由于增广路径的首尾是非匹配点，那么增广路径的首尾边必为非匹配边。由于增广路径中匹配边与非匹配边一次交替，所以非匹配边的数目比匹配边多一条。我们可以利用这个特性来改进匹配，只要将匹配边与非匹配边互换即可，如下图所示。



所以，只要不断地迭代这个过程，直至找不到增广路径为止，就可以找到一般图的最大匹配。

> 补充：若要寻找带权一般图上的最大匹配，则在上面算法的基础上加个权重和判断即可。