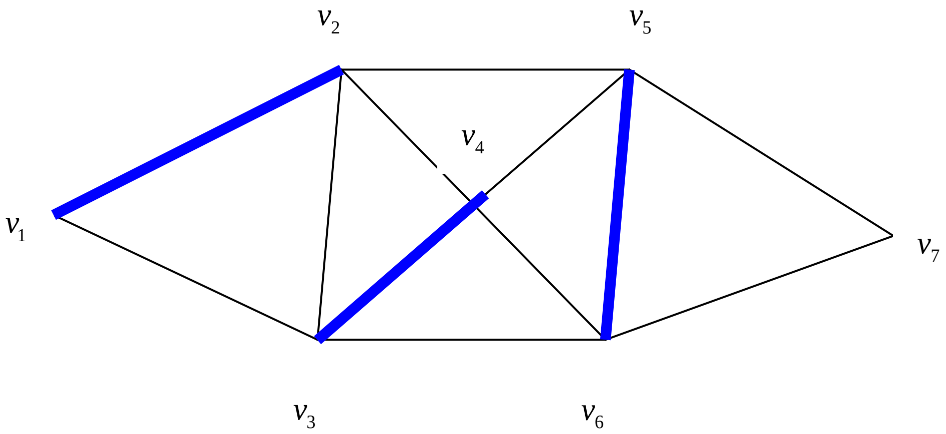
第六章 组合优化问题的有效算法

6.3 最优匹配的有效算法－1

一、匹配问题

图$G=(V,E)$的一个匹配M是边的一个子集，使得M中任何两条边没有公共端点。

例如：



$\{[v_1,v_2],[v_3,v_4],[v_5,v_6]\}$是一个匹配。

注：G的匹配不可能有多于$\lfloor |V|/2 \rfloor$条边。

匹配问题：找出G的最大匹配M。当匹配中的边数为$\lfloor |V|/2 \rfloor$时，称这个匹配是完美匹配。

设M是图G的一个匹配，M中的边称为**匹配**边，不在M中的边称为**自由边**。

若$[u,v]$是匹配边，则称u是v的**配偶**（关联匹配边的点）。

不关联匹配边的点称为**未盖点**，否则称为**已盖点**。

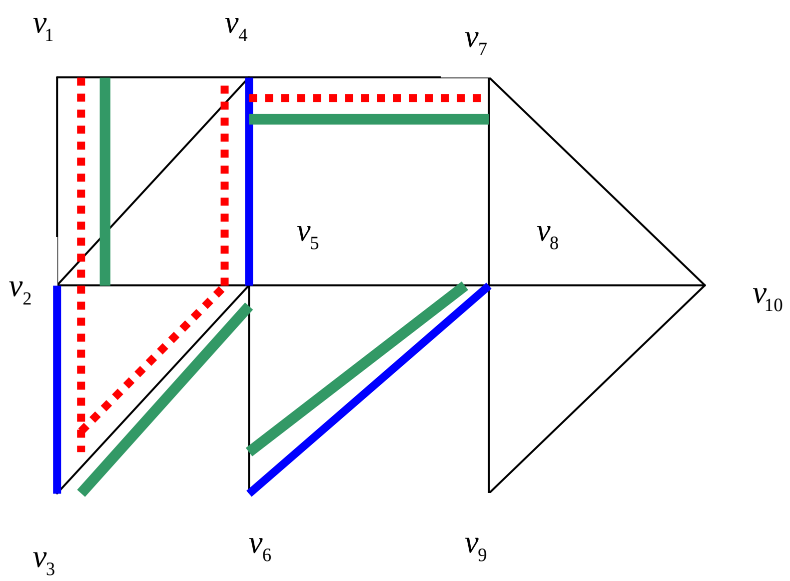
交错路：G中的一条路$P=[u_1,u_2,\cdots,u_k]$，如果边$[u_1,u_2],[u_3,u_4],\cdots,[u_{j-2},u_{j-1}],\cdots$都是自由边，而其余边为匹配边，则称$P$为交错路。

从交错路的起点（未盖点）起，所有奇数点称为**外点**，而偶数点称为**内点**。

增广路：一条交错路$P=[u_1,u_2,\cdots,u_k]$，若u_1和u_k都是未盖点，则称P为增广路。

引理 6.7 设$P=[u_1,u_2,\cdots,u_{2k}]$是图$G$中关于匹配$M$的增广路。则$M'=M\oplus P$是$|M|+1$条边的匹配。其中：$M\oplus P=(M-P)\cup(P-M)$。

如：



$M=\{[v_2,v_3],[v_4,v_5],[v_6,v_8]\}$（蓝色边）

$P=[v_1,v_2,v_3,v_5,v_4,v_7]$（红色边）

$M-P=\{[v_6,v_8]\}$——在 M 上但不在 P 上的边，即交错路以外的匹配边；

$P-M=\{[v_1,v_2],[v_3,v_5],[v_4,v_7]\}$——交错路上的非匹配边。

$M'=M\oplus P=\{[v_1,v_2],[v_3,v_5],[v_4,v_7],[v_6,v_8]\}$（绿色边）

证明：

（1）$M\oplus P=(M-P)\cup(P-M)$ 是一个匹配：即任何两条边无公共端点。用反证法。假设存在 $e,e'\in M\oplus P=(M-P)\cup(P-M)$，且有一个公共端点，则共有以下三种情况：

① $e,e'\in(M-P)$：因为 M 是一个匹配，所以不可能发生。

② $e,e'\in(P-M)$：因为 P 是一条交错路，$P-M$ 中的边是属于 $[u_{2j-1},u_{2j}]$ 的形式，所以 $P-M$ 中的任何两条边不可能有公共端点，所以不可能发生。

③ $e\in(M-P)$ 且 $e'\in(P-M)$：设 $e'=[u_{2j-1},u_{2j}]$，与 e 有公共端点，不妨设为 u_{2j}，但是 $e''=[u_{2j},u_{2j+1}]\in M$，所以 M 中的边 e 和 e'' 有公共端点，矛盾。

所以，$M\oplus P=(M-P)\cup(P-M)$ 是一个匹配。

（2）$|M'|=|M|+1$：因为 P 有 $2k-1$ 条边，其中 k 条自由边，即 $[u_1,u_2],[u_3,u_4],\cdots,[u_{2k-1},u_{2k}]$，而 $k-1$ 条边属于 M，所以 $M'=M\oplus P$ 有 $|M|+1$ 条匹配边。

结论：一条增广路可以用于增大匹配。

定理 6.5 G 的一个匹配 M 是最大匹配 $\Leftrightarrow$ G 中不存在关于 M 的增广路。

证明：

"$\Rightarrow$" 由上述引理可得。

"$\Leftarrow$" 反证法：假设 G 中不存在关于 M 的增广路，并且 M 不是 G 的最大匹配，则存在一个匹配 M' 使得 $|M'|>|M|$。

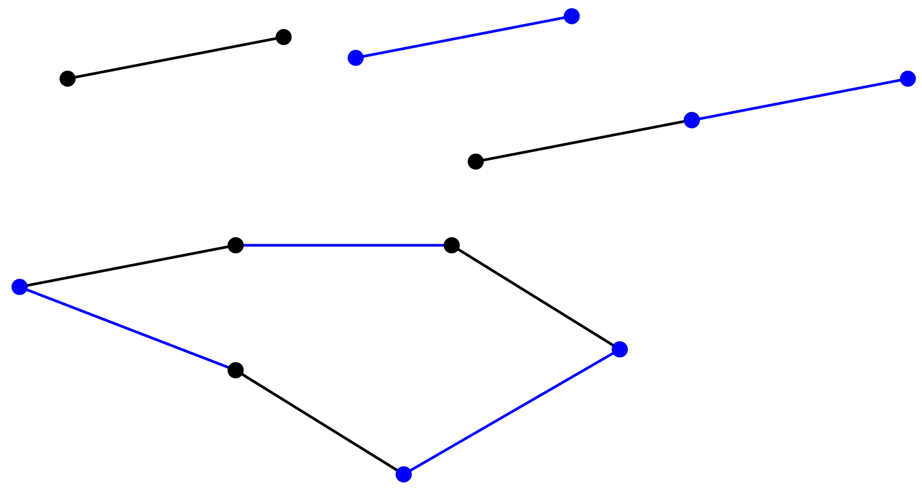
记 G 的子图 $G'=(V,M\oplus M')$

因为：匹配中的任何两条边无公共端点

所以：子图 $G'=(V,M\oplus M')$ 有特殊结构——每个节点的次 ≤ 2

如果一个节点的次为 2，则与之关联的两条边，一条边在 M，另一条边在 M' 中。

所以：G' 的连通分图或者是一条路，或者是偶数长的圈（在 M 中的边和在 M' 中的边各占一半）。



（1）对于偶数长的圈：由于 M 中的边和在 M' 中的边各占一半，但由于 $|M'|>|M|$，所以必存在一条路，使得在 M' 中的边数 $>$ 在 M 中的边数，所以它是一条关于 M 的增广路，矛盾。

（2）若不存在圈：由于 $|M'|>|M|$，所以必存在一条路，使得在 M' 中的边数 $>$ 在 M 中的边数，所以它是一条关于 M 的增广路，矛盾。

证毕

注：

- 这个定理刻画了最大匹配的特征，它与用增广路描述最大流的特征十分相似。因此，人们就试图模拟最大流算法设计匹配算法；
- 从任意一个匹配开始，寻找关于M的增广路，进行增广，重复这一过程，即可得到最大匹配；
- 目前关于匹配问题的所有算法都是基于这一思想，但是这些算法都相当复杂；
- 把这一思想用到特殊的图——二部图的匹配问题，比较简单，几乎就是模拟最大流的算法。

二、二部图的匹配算法

——模仿最大流的增广路算法，构造二部图的匹配的有效算法

因为：二部图的匹配问题是一般匹配问题的特殊情况

所以：可以反复应用寻找二部图关于当前匹配M的增广路P，及增广当前的匹配为$M \oplus P$的方法，求解二部图的匹配。

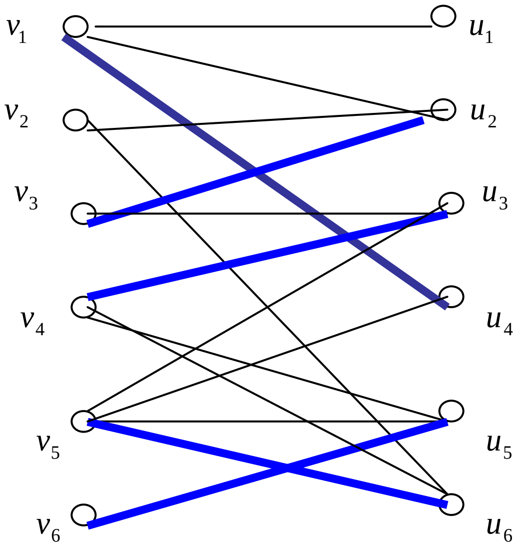
设M是二部图$B=(V,U,E)$的一个匹配，如何建立在B中搜索关于当前匹配M的增广路的有效方法？

增广路的搜索必须从一个未盖点出发，构造交错路。

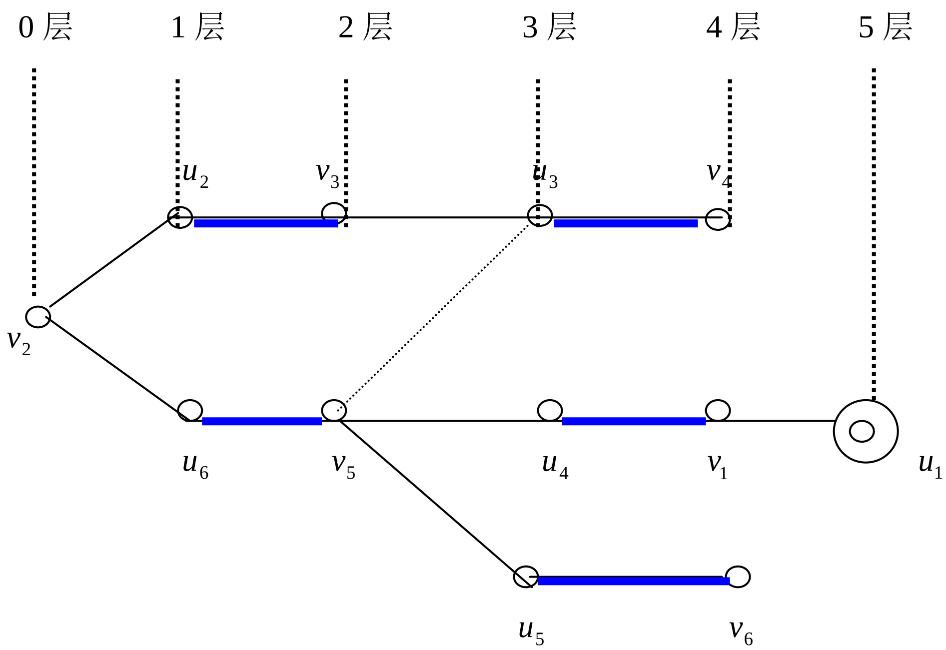
因为：一条增广路，它的一端必须在V里，而另一端在U里，且都是未盖点，

所以：我们可以从V中的一个未盖点v_i出发，用广探法同时搜索从v_i出发的所有交错路，找到U中的一个未盖点u_j，从而找到一条自未盖点v_i到未盖点u_j的一条增广路。

如下图，从V中的一个未盖点v_2出发，用广探法同时搜索从v_2出发的所有交错路，从v_2出发，考察所有与v_2相邻的节点，即u_2和u_6，



图(a)



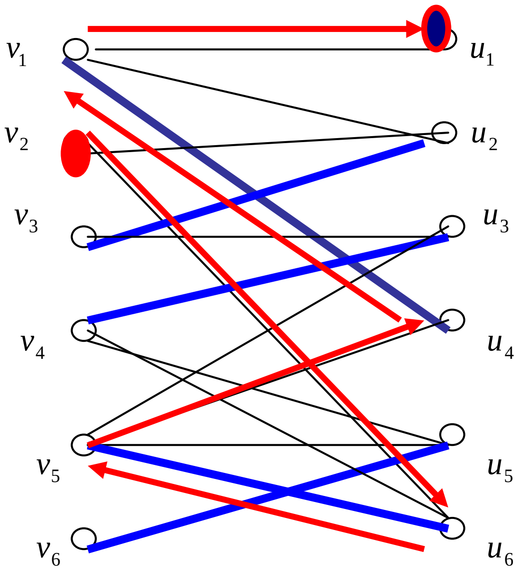
图(b)

注：因为在检查v_5之前，已经从v_3到达了节点u_3，所以边$[v_5,u_3]$可以去掉，这样做只是在求交错路的过程中去掉了一些多余的增广路。

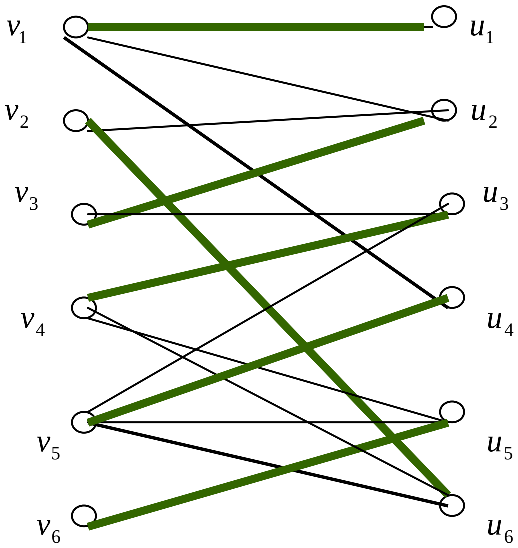
搜索得到了一条增广路：

$$[v_2,u_6],[u_6,v_5],[v_5,u_4],[u_4,v_1],[v_1,u_1]$$

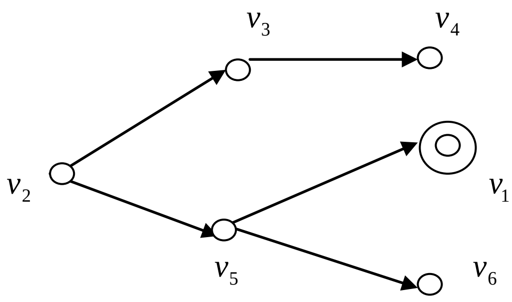
下图红线所示：



将 M 增广，得到新的匹配：



搜索增广路的过程，实际上是一种广探法。这个广探法有一种特殊的结构：它对于奇数层节点（v_i 为 0 层，奇数层节点即为U 中的节点）的搜索非常简单，这是因为奇数层节点的下一个节点就是它的配偶。——可以略掉对奇数层节点的搜索，而直接从外点到新的外点搜索。



图(c)

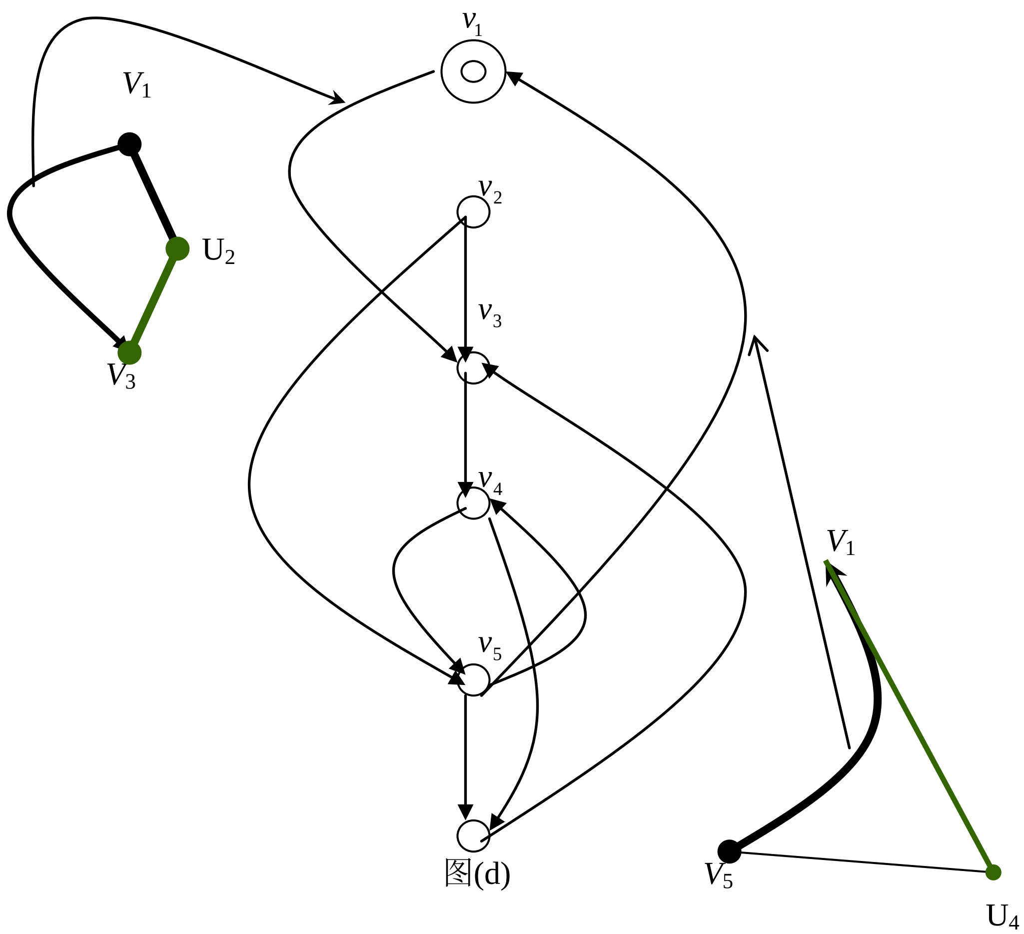
这种搜索相当于搜索一个辅助有向图(V,A)，其中：

$(v_1,v_2)\in A \Leftrightarrow$在一条增广路上$v_2$是$v_1$的下一个外点，即：$v_1$相邻于$v_2$的配偶。

因为：从V中节点出发的交错路，只有V中节点才可能成为外点

所以：辅助有向图的节点集为V。

对应于图(a)的辅助有向图为：



图(d)

容易看出，表示在图(c)上的增广路搜索，是从v_2出发，对辅助有向图(d)施行的广探法。

在我们的算法里用到了两个数组 mate（配偶）和 exposed（未盖点），另外还有一个搜索的数组 label。mate 数组有$|V|+|U|$个元素，它表示当前的匹配；对每一个节点$v \in V$，用 exposed[v]表示相邻于v的U中的一个未盖点，若没有这样的节点存在，则 exposed[v]＝0。

显然，若一个节点$v \in V$且在搜索中遇到$\text{exposed}[v] \neq 0$，则发现了一条增广路。为了找出这条增广路，需要一个递归的增广程序$\text{augment}(v)$，并进行增广。

二部图的最大匹配算法：

Input: 二部图 $B=(V,U,E)$。

Output: 用数组 mate 表示的 B 的最大匹配

begin

for all $v \in V \cup U$ do matel[v]:=0;（初始化）

stage:

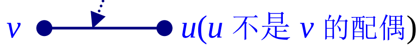
begin

for all $v \in V$ do exposed[v]:=0;

$A := \phi$;（开始构造辅助辅助有向图 (V,A)） u 没有配偶

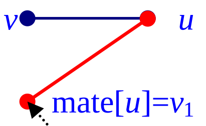
for all $[v,u] \in E$ do 不是匹配边

if mate$[u]=0$ then exposed$[v]:=u$;



else u 是 v 的未盖点

if mate$[u] \neq v$ then $A := A \cup (v, \text{mate}[u])$; u 有配偶但不是 v



$Q=\phi$ （Q 是 V 中的未盖点集合）

for all $v \in V$ do

if matel[v]=0 then $Q := Q \cup \{v\}$, label$[v]:=0$;

u 的配偶

v 相邻于 u 的配偶 mate[u], 所以 (v,mathe[u]) $\in A$

while $Q \neq \phi$ do

begin

令 v 是 Q 中的一个节点；

$Q = Q \setminus v$;

if exposed$[v] \neq 0$ then augment(v),go to stage;

exposed[v] 表示相邻于 v 的 U 中的一个未盖点

else

for all 未标号的节点 v' 且使得 $(v,v') \in A$ do

label$[v']=v, Q := Q \cup \{v'\}$;

找增广路，并进行增广

end

end

end

```
procedure augment(v)
  if label[v] = 0 then mate[v]:=exposed[v], mate[exposed[v]]:=v;
  else
    begin
      exposed[label[v]]:=mate[v];
      mate[v]:= exposed[v];
      mate[exposed[v] ]:=v;
      augment(label[v])
    end
```

定理 6.6 求二部图的上述算法是正确的，其计算复杂度为$O\left(\min\{|V|,|U|\}\cdot|E|\right)$。

证明：如果辅助有向图中不存在自V中未盖点到目的点的路，则算法终止。由辅助有向图的构造知，此时意味着不存在B中关于当前匹配的增广路，从而目前的匹配是最优匹配。正确性得证。

计算复杂度：

（1）迭代步数（阶段数）：

因为：B中任何一个匹配的边数不能超过$\min\{|V|,|U|\}$，且每次增广匹配边数增加 1。

所以：算法最多执行$\min\{|V|,|U|\}$个阶段数。

（2）每一步迭代（一个阶段）的计算复杂度：

因为:构造辅助有向图和计算数组 exposed 都需要考察B中的每一条边，

所以：所需要的计算复杂度为：$O(|E|)$

又在辅助有向图中求有向路，可以在$O(|A|)=O(|E|)$时间内完成。

所以，整个算法的计算复杂度为$O\left(\min\{|V|,|U|\}\cdot|E|\right)$。

三、二部图的匹配的最大流算法

——直接将二部图的匹配问题化成最大流问题，然后用最大流的有效算法求解，从而得到二部图的匹配问题的有效算法。

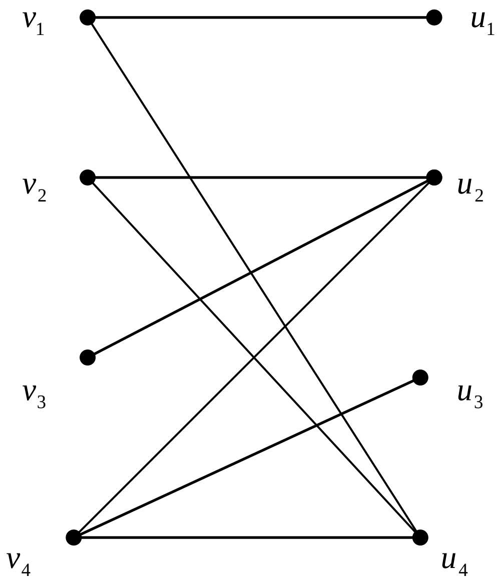
将二部图的匹配问题归结为简单网络上的最大流问题。给定一个二部图$B=(V,U,E)$，定义一个单位容量的网络$N(B)=(s,t,W,A)$，其中s,t是新增的两个特殊节点，$W=\{s,t\}\cup V\cup U$，且A由下述三种类型的弧组成：

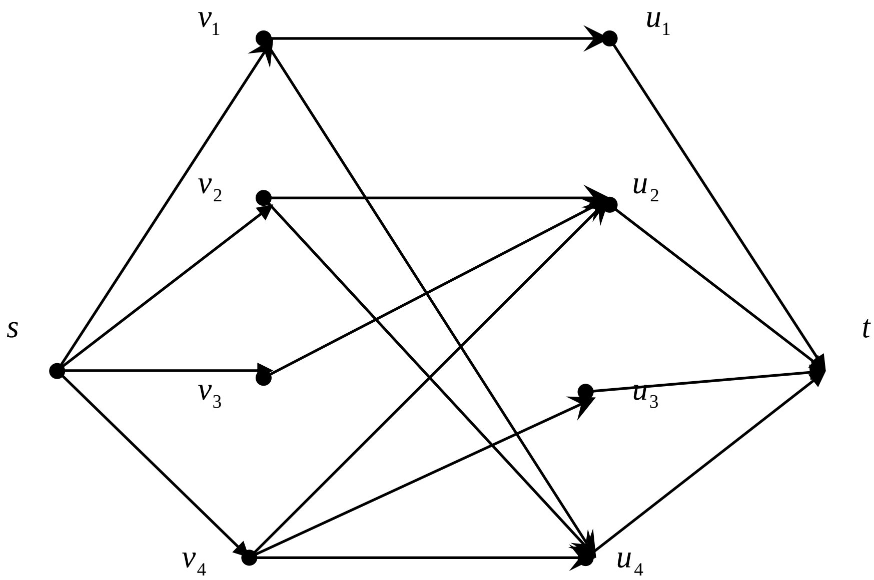
（1）$\forall v\in V,(s,v)$弧；

（2）$\forall u\in U,(u,t)$弧；

（3）对$\forall v\in V,u\in U,(v,u)\in E$，弧$(v,u)$。

如：





注：对如何二部图，$N(B)$ 都是一个简单网络，这是因为 $N(B)$ 中每一个 V 中节点的入次为 1，而 U 中节点的出次为 1。

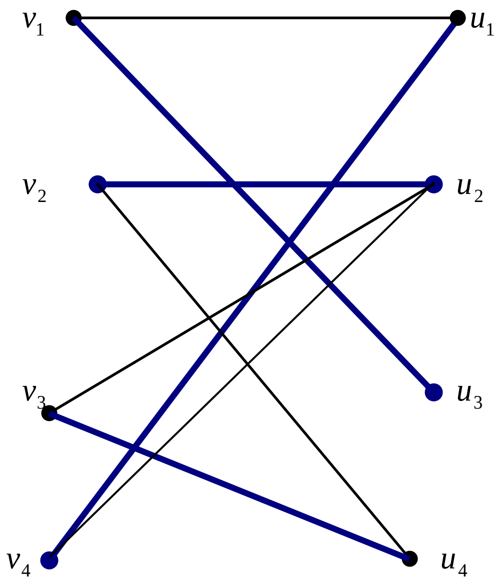
引理 6.7 二部图 $B=(V,U,E)$ 的最大匹配的边数为 $N(B)$ 最大的 $s-t$ 流值。

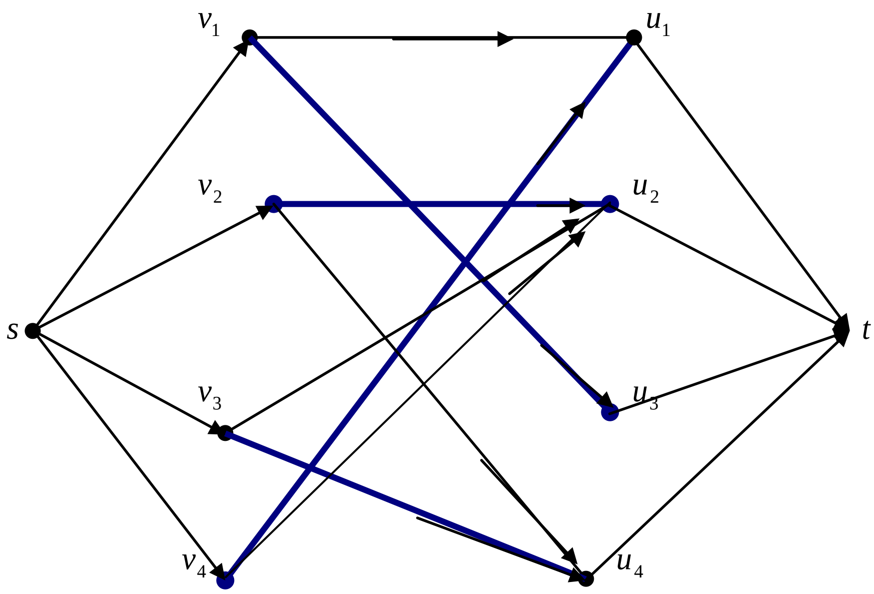
证明：1）给定 B 的任意一个匹配 M，构造 $N(B)$ 中的一个可行流如下：

对V中的每一个匹配点v，让一个单位流通过弧(s,v)，

对U中的每一个匹配点u，让一个单位流通过弧(u,t)，

对每一个匹配边$[v,u]\in M$，让一个单位流通过弧(v,u)。





显然，这样得到的流是可行流，且流值为$|M|$。

2）给定$N(B)$的一个最大流f，就存在一个等价的 0−1 整数流，从这个整

数流开始，我们构造一个匹配M，使得M是由所有使得$f(v,u)=1$的边组成。根据$N(B)$的构造，它一定是合理的匹配，这是因为若E中由一个公共端点v（或者u），则$N(B)$中对应的这两条弧中至少有一个其流值为 0，否则就要破坏弧(s,v)（或者(u,t)）的容量限制。

注：给定$N(B)$中一个整数流的最大流后，能找出B的一个最大匹配，并且时间复杂度为$O(|V|)$。

定理 6.7 解二部图$B=(V,U,E)$的最大匹配问题，可以在时间$O\left(|V|^{\frac{1}{2}}\cdot|E|\right)$内完成。

注：对于二部图的匹配问题，这个算法是目前已知的渐近最快的算法。