第九次作业

1. 假定除了边的容量外，流网络还有结点容量。即对于每个结点 u，有一个极限值 l(u)，请说明如何将带有结点容量的流网络 G=(V, E)转化为等价的没有结点容量的流网络 G’=(V’,E’)，使得最大流的取值一样。图 G’里有多少个结点和多少条边?

可以将图 G.中的每个点 u 都拆分为两个点 u1， u2，其中进入 u1 的流量就是进入 u 的流量，从 u2 出的流量就是从 u 出的流量，在 u1，u2 之间添加一条边，这条边的容量就是 u 结点的容量。因此图 G’中有 2|V|个结点和|V|+|E|条边。

通过这种构造，G 中所有的容量都对应到 G’中所有的容量，流经 G 中 u 的流量等价于流经 G’中(u1,u2)的流量，所有图 G 中的所有流都等价于图 G’中的所有流，因此该构造是可行的。

2. 说明在流网络 G=(V, E)中，如何使用一个最多包含|E|条增广路径的序列来找到一个最大流。

将题目的描述转化为：如何找到一个边集 E’’使得|E’’|小于等于|E|，而且可以通过 E’’找到 G 的最大流。

算法描述：

（1） 构造 G’=(V, E’)，其中 E’={G 中所有的增广路径的并集}。

（2） 使用 Ford-Fullkerson 算法找到 G’的最大流

（3） 找到最大流对应的最小割

（4） E’’={最小割的边}，| E’’|≤|E|，通过 E’’可以确定 s 到 t 的增广路径

算法正确性：

即证 G’的最大流与 G 的最大流相同。

假设 G’的最大流与 G 的最大流不相同，则存在$e \in G$，$e \notin G'$，由于$e \notin G'$，所以 e 不在 G 的任意一条增广路径上，因此 e 不能是 s 到 t 的流的一部分，与假设矛盾。

因此 G’的最大流与 G 的最大流相同。

3. 证明：如果总是释放高度最高的溢出结点，则可以使用推送-重贴标签算法在 $O(V^3)$时间内完成。

构造一个优先级队列 Q，Q 以点的高度来判断优先级，使用双向链表来构造 Q，其中链表头指向高度最高的点的链表，下一个指向高度减 1 的点的链表，因此该优先级队列添加元素和删除元素只需要线性时间，使用这个优先级队列实现 PUSH-RELABEL 算法：

```
PUSH-RELABEL-PRIORITY-QUEUE(G, s)
    INITIALIZE-PREFLOW(G, s)
    let q be a new empty priority queue
    for v ∈ G.Adj[s]
        PUSH(q, v)
    while q.head != NIL
        DISCHARGE(q.head)
        POP(q)
```

在执行 DISCHARGE 操作之前必然执行过一次 PUSH 操作，而一个点进行 PUSH 操作之后除非进行 RELABEL，否则不可能再次进行 PUSH，因此在|V|次 DISCHARGE 操作并且没有

RELABEL 操作之后，就没有点可以进行 PUSH 操作，因此要么算法结束，要么有 RELABEL 操作来进行。所以 DISCHARGE 操作执行的次数为$O(V^3)$，因此算法的运行时间为$O(V^3)$。

4. Hopcroft-Karp 二分匹配算法

a. 证明：如果 M 是图 G 的一个匹配，P 是一条关于 M 的增广路径，则对称差$\mathbf{M \oplus P}$也是一个匹配并且$\mathbf{|M \oplus P| = |M| + 1}$，另外证明：如果$\mathbf{P_1, P_2, \cdots, P_k}$为关于 M 的结点分离的增广路径，则对称差$\mathbf{M \oplus (P_1 \cup P_2 \cup \cdots \cup\ P_k)}$是一个基数为$\mathbf{|M| + k}$的匹配。

证明:

由于简单路径 P 的起点是从一个未匹配的点出发，所以 P 的第一条边一定不在 M 里，同样的，由于 P 的重点也是未匹配的点，所以 P 的最后一条边也一定不在 M 里。由于在 M 的边和不在 M 的边是交替的，所以在 M 的边要比不在 M 里的边少一条。即如果有 k 条边在 M 里，则有 k+1 条边不在 M 里，所以有:

$$|\mathrm{M \oplus P}| = |M| + |P| - 2k = |M| + 2k + 1 - 2k = |M| + 1$$

如果$\mathrm{P_1, P_2, \cdots, P}_k$为关于 M 的结点分离的增广路径，设$\mathrm{k}_i$是$\mathrm{P}_i$在 M 的边的数量，那么$|\mathrm{P}_i| = 2k_i + 1$，因此:

$$|\mathrm{M \oplus (P_1 \cup P_2 \cup \cdots \cup\ P}_k)| = |\mathrm{M}| + \sum_{i=1}^{k}(|P_i| - 2k_i) = |\mathrm{M}| + \mathrm{k}$$

要证明对称差$\mathrm{M \oplus P}$是一个匹配，即证对于任意顶点 u，不存在两个关联边 e，e'都在$\mathrm{M \oplus P}$里。假设存在 u 的两个关联边 e，e'都在$\mathrm{M \oplus P}$里，那么着两条边不可能都在 M 里，因为 M 是一个匹配，也不可能都在 P 里，因为 P 是简单路径，所以不妨令$\mathrm{e \in M, e' \in M - P}$。然而如果$\mathrm{e \in M}$，那么$\mathrm{e \in P}$，所以$\mathrm{e \notin M \oplus P}$，这与 e 在$\mathrm{M \oplus P}$里矛盾，所以不存在两个关联边 e，e'都在$\mathrm{M \oplus P}$里。因此对称差$\mathrm{M \oplus P}$是一个匹配，同理可证$\mathrm{M \oplus (P_1 \cup P_2 \cup \cdots \cup\ P}_k)$也是一个匹配。

b. 给定图 G 的两个匹配 M 和 M*，证明图$\mathbf{G' = (V, M \oplus M*)}$中的每个结点的度数最多为 2. 同时证明 G'是由一些不相交的简单路径或环路组成。另外，是说明每条这样的简单路径或环中的每条边交替属于 M 和 M*。证明：如果$\mathbf{|M| \le |M*|}$，则$\mathbf{M \oplus M*}$至少包含$\mathbf{|M*| - |M|}$条关于 M 的结点不相交的增广路径。

证明:

（1）假设$\mathrm{G' = (V, M \oplus M*)}$中有一个结点的度数为 3，则至少有两条边来自于 M 或 M*，由于 M 或 M*都是一种匹配，所以不可能同时有两条边连接同一个顶点，所以$\mathrm{G' = (V, M \oplus M*)}$中的每个结点的度数最多为 2

（2）如果 G'是由相交的简单路径或环路组成，那么至少会有一个点的度数超过 2，由（1）可知这种情况是不存在的。

（3）在简单路径上相邻的两条边经过同一个点，因此这两条边不能出现在同一个匹配中，所以他们交替属于 M 和 M*

（4）由于简单路径或环中的每条边交替属于 M 和 M*，所以在环里分别属于 M 和 M* 的边的数量相等，在简单路径上属于 M 与 M*的边的数量最多相差 1 个，所以如果$|\mathrm{M}| \le |\mathrm{M*}|$，$\mathrm{M \oplus M*}$至少包含$|\mathrm{M*}| - |\mathrm{M}|$条关于 M 的结点不相交的增广路径。

c. 证明：如果路径 P 与路径$\mathbf{P_1, P_2, \cdots, P_k}$之间没有共同结点，则路径 P 有多于 l 条边。

证明:

因为 P 是相对于 M'的一条增广路径，所以 P 的第一条路径不在 M'中。因为 M 中的每个点都与 M'中的而每个点相关联，所以 P 的起点在 M 中没有被匹配到。P 中在 M'中的边都

在 M 中，P 中不在 M'的边不可能在 M 中。P 的最后一条边一定和 M 中的一个顶点相关联，而这个顶点在 M'中没有被匹配到。因为在 M'中没有匹配到的点在 M 中也没有匹配到，所以 P 是 M 的一条增广路径。因为$P_1, P_2, \cdots, P_k$是 M 的不相交增广路径的最大集合，而 P 不在改集合中，所以 P 一定有多于 l 条边。

d. 现在假定路径 P 和路径$\mathbf{P_1, P_2, \cdots, P_k}$存在共同结点。设 A 为边$(\mathbf{M \oplus M'}) \oplus \mathbf{P}$的集合。证明：$\mathbf{A = (P_1 \cup P_2 \cup \cdots \cup\ P_k) \oplus P}$并且$\mathbf{|A| \geq (k+1)l}$。同时证明：路径 P 包含的边多于 l 条。

证明：

在$M \oplus M'$中的边要么在 M 中，要么在 M'中，所以 M'中匹配的边一定在$P_1 \cup P_2 \cup \cdots \cup\ P_k$里。M 中的边要想成为 M'中的匹配边需要满足在 M 和$P_1 \cup P_2 \cup \cdots \cup\ P_k$中，所以$M \oplus M' = (P_1 \cup P_2 \cup \cdots \cup\ P_k)$，所以$A = (P_1 \cup P_2 \cup \cdots \cup\ P_k) \oplus P$

假设 P 中的边 e 也是P_i中的边，，因为 P 是 M'的增广路径，所以 e 要么属于 M'，要么属于 E-M'，如果 e 属于 M'，因为 P 也是 M 的增广路径，所以 e 也属于 M，因此 e 不能属于P_i。如果 e 属于 E-M'，那么 e 属于 E-M。而因为 e 是P_i的一条边，所以 e 属于 M，这也是一个矛盾，所以不存在这样的 e，所以 P 和P_i是结点不相交的。

因为 P 的边要么在 M'中，要么在 E-M'中，且与P_i不相交，P 还是 M 的增广路径，所以|P|>l，因为 A 中每条边都不相交，所以有$|A| \geq (k+1)l$

e. 证明：如果关于 M 的一条最短路径有 l 条边，则最大匹配的规模至多为$\mathbf{|M| + |V|/(l+1)}$

证明：

假设存在 M*匹配规模大于$|M| + |V|/(l+1)$，由 b 可知 M 中就有超过$|V|/(l+1)$的不相交路径，每条路径都有超过 l 条边，也就是(l+1)条边，所以这些路径上一共有超过$|V| \times (l+1)/(l+1)$，也就是|V|个不同的顶点。这与一共只有|V|个顶点矛盾，所以最大匹配的规模至多为$|M| + |V|/(l+1)$。

f. 证明：Hopcroft-Karp 二分匹配算法中 repeat 循环的迭代次数至多为$\mathbf{2\sqrt{V}}$

证明：

假设 M*是 G 的最大匹配，有$|M| \leq |M*|$，因此根据 b 的结论， $M \oplus M*$至少包含$|M*| - |M|$条关于 M 的结点不相交的增广路径。根据 c 的结论，这些路径也是 M 的增广路径。在$\sqrt{|V|}$次迭代后，每条路径的长度为$\sqrt{|V|}$，这样的路径最多有$\sqrt{|V|}$条，所以$|M*| - |M| \leq \sqrt{|V|}$。因此 repeat 循环继续进行$\sqrt{|V|}$ 次，所以 repeat 循环的迭代次数至多为$2\sqrt{V}$。

g. 给出一个 O(E)时间复杂度的算法，可以找到一个关于给定匹配 M 的结点不相交最短增广路径$\mathbf{P_1, P_2, \cdots, P_k}$的最大集合，证明：算法 HOPCROFT-KARP 的总运行时间为$\mathbf{O(\sqrt{V}E)}$

算法：

对于 L 集中每个没有能匹配的顶点，运行一次 BFS，找到到达 R 集中未匹配顶点的最短路径的长度。但在运行 BFS 的时候需要保证访问的边在 M 和 E-M 之间交替。到达 R 中顶点的时候我们就找到了一条长度为 k 的最短增广路径。

在查找时，如果长度超过了 k，那一定不是最短增广路径，算法可以直接停止查找。在寻找不相交路径时，从找到的最短增广路径在 R 集上的点开始执行 DFS 算法，记录查找过程中的上一个顶点，在运行 DFS 的时候同样也要在 M 和 E-M 之间交替，找到这样一个路径需要$O(E)$的时间，由 f 的讨论可知总的运行时间为$O(\sqrt{V}E)$。