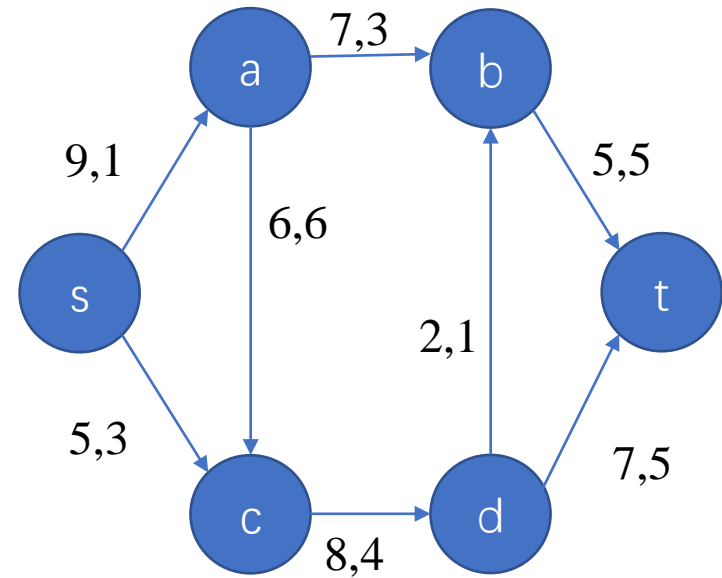
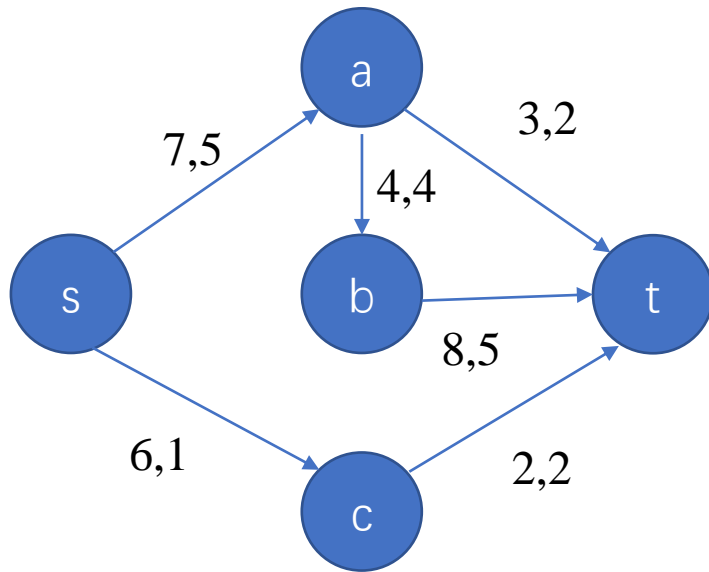
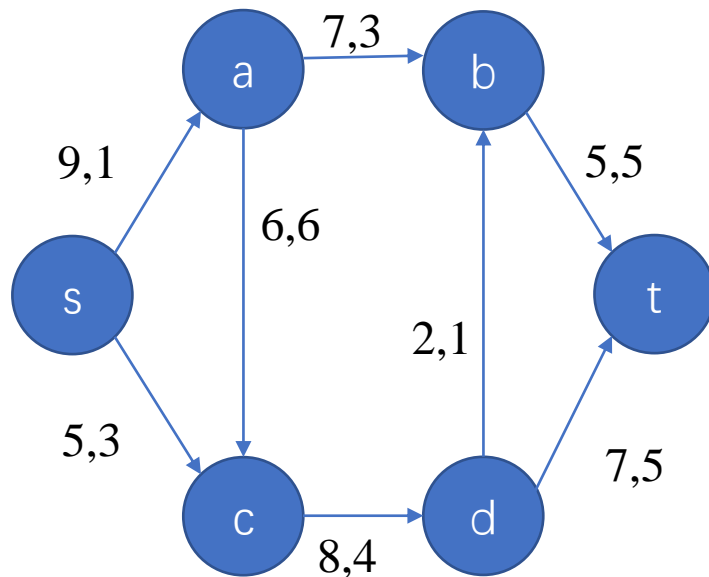
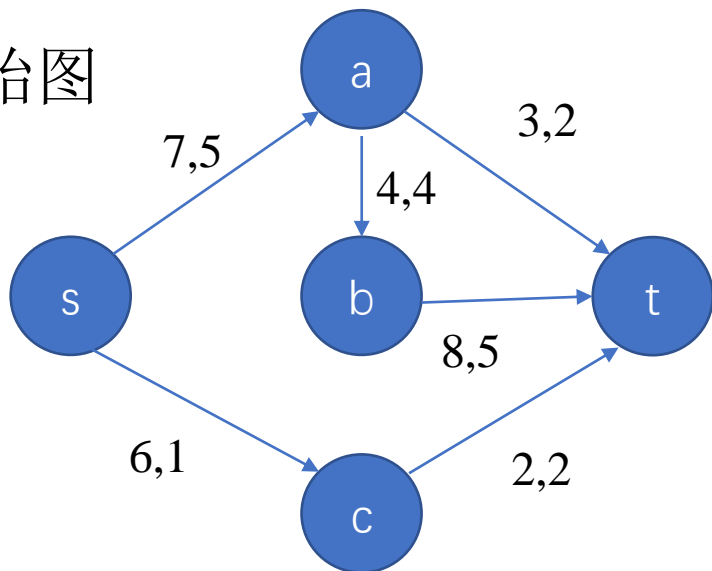


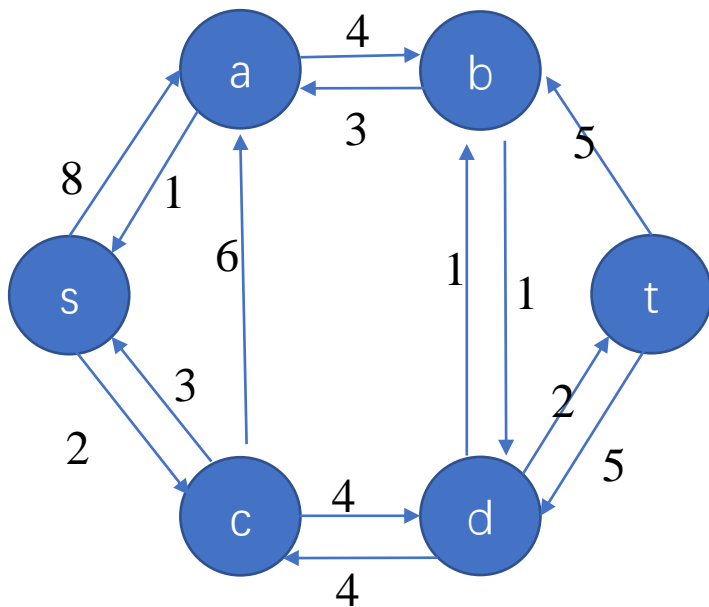
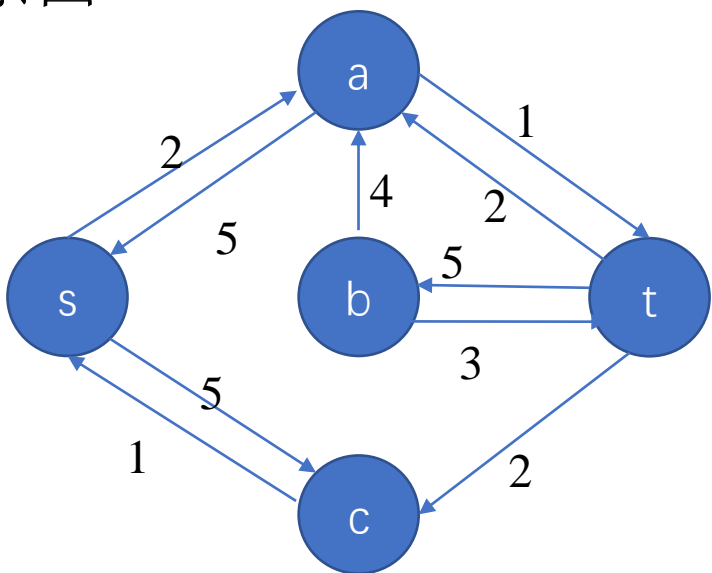
1. 给定如下两个带流的网络，画出对应的剩余图。



原始图



剩余图



2. 给定如下网络，采用Ford-Fulkerson算法按照增广路径1,2交替迭代，并更新剩余图。

增广路径1: $s \rightarrow a \rightarrow b \rightarrow t$, 增广路径2: $s \rightarrow b \rightarrow a \rightarrow t$

Ford-Fulkerson算法:

输入: 网络 (G, s, t, c)

输出: G 中的一个流

1. 初始化剩余图, 设 $R=G$

2. for 边 $(u, v) \in E$

3. $f(u, v) \leftarrow 0$

4. end for

5. While 在 R 中有一条增广路径 $p=s, \dots, t$

6. 设 Δ 为 p 的瓶颈容量

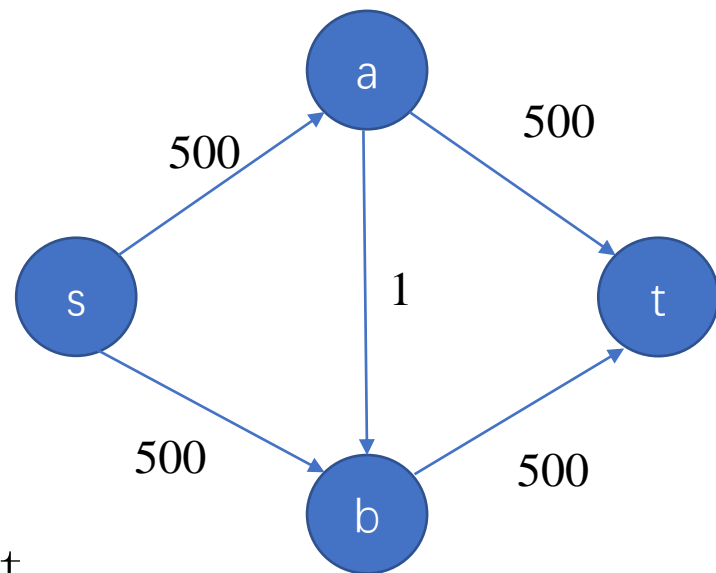
7. for p 中的每条边 (u, v)

8. $f(u, v) \leftarrow f(u, v) + \Delta$

9. end for

10. 更新剩余图 R

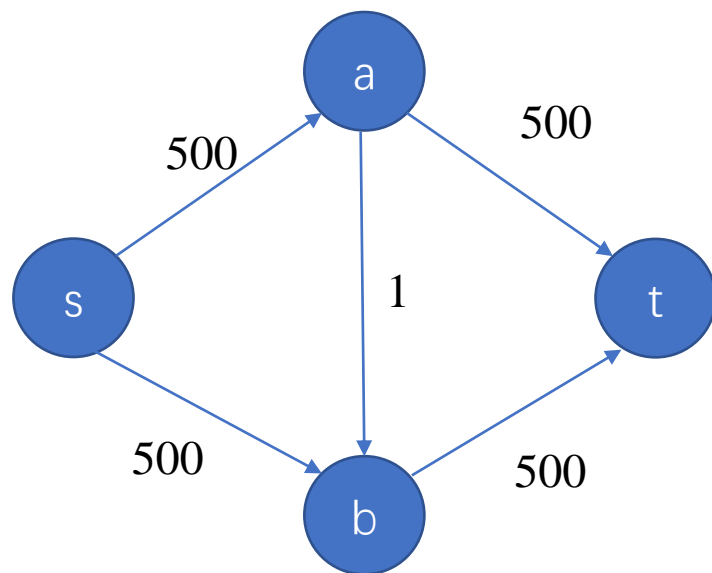
11. End while



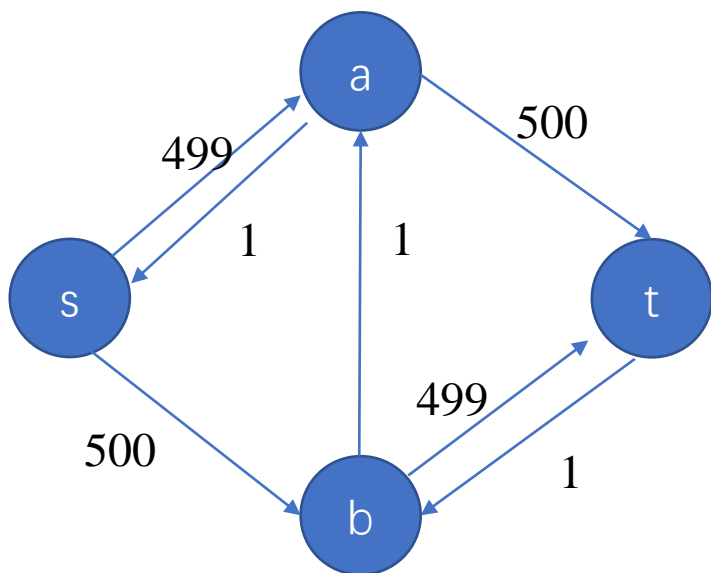
2. 给定如下网络，采用Ford-Fulkerson算法按照增广路径1,2交替迭代，并更新剩余图。

增广路径1: $s-a-b-t$

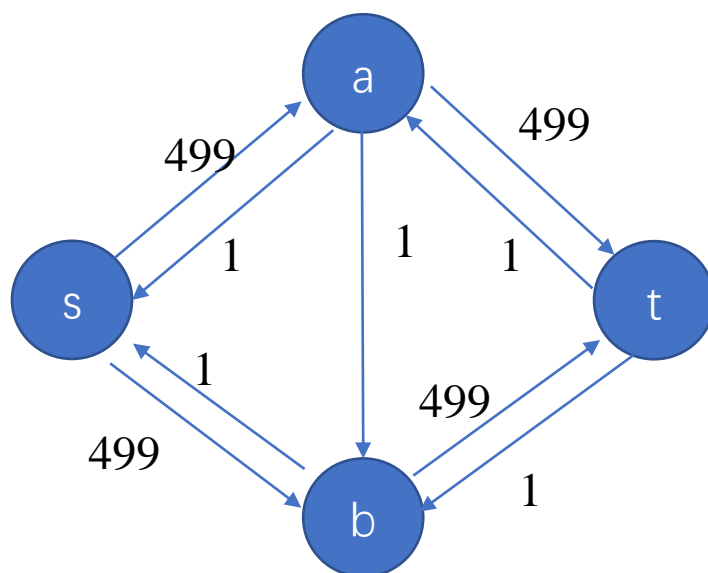
增广路径2: $s-b-a-t$



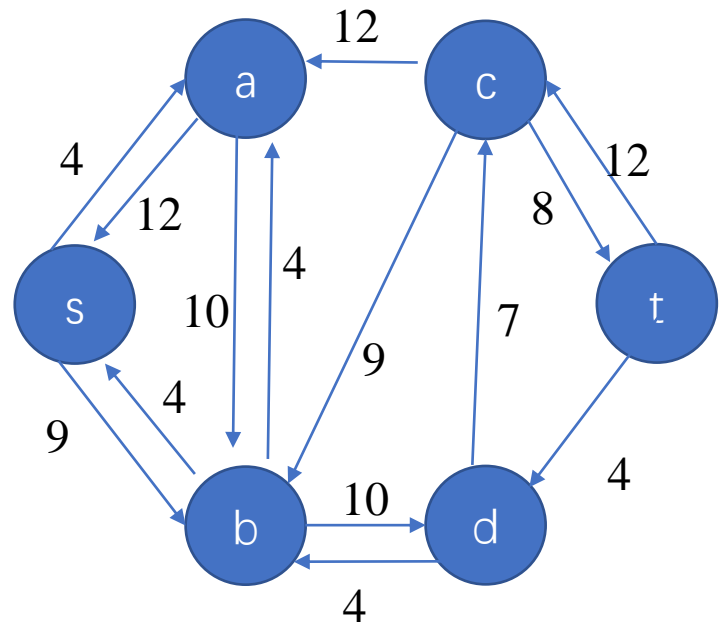
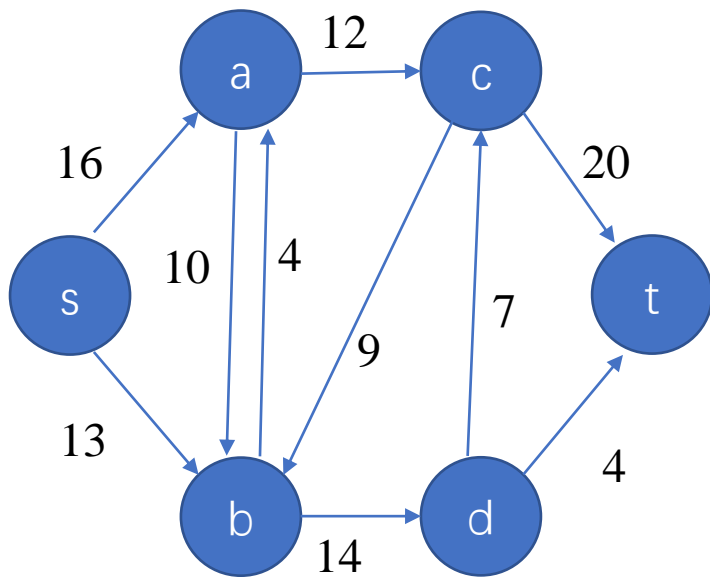
第一次迭代，路径 $s-a-b-t$



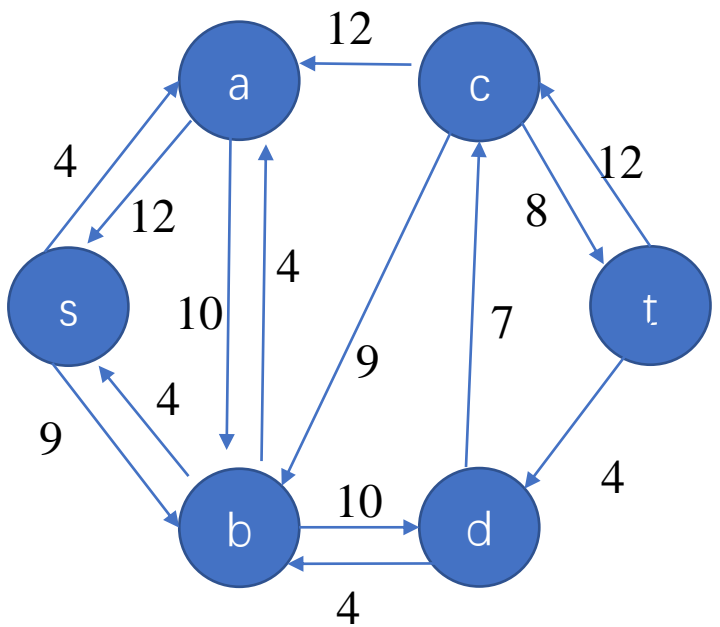
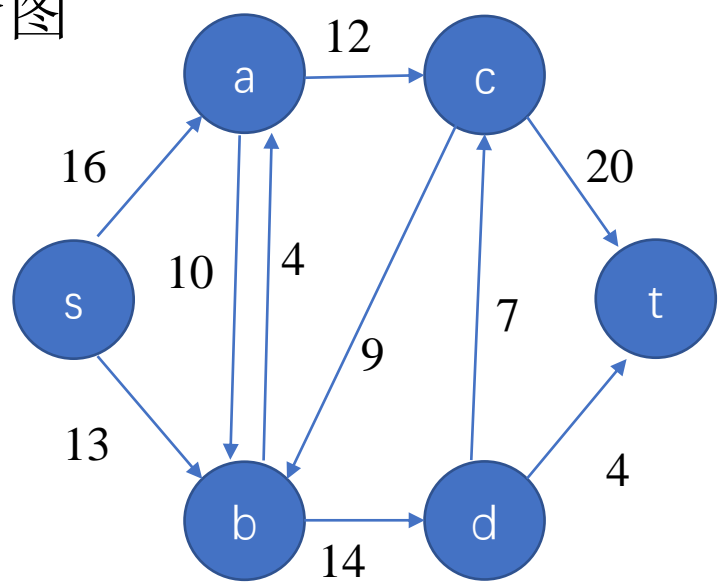
第二次迭代，路径 $s-b-a-t$



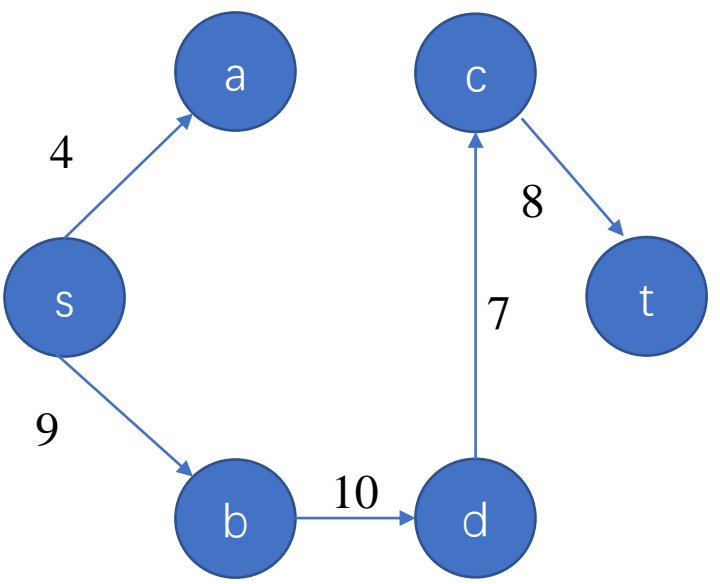
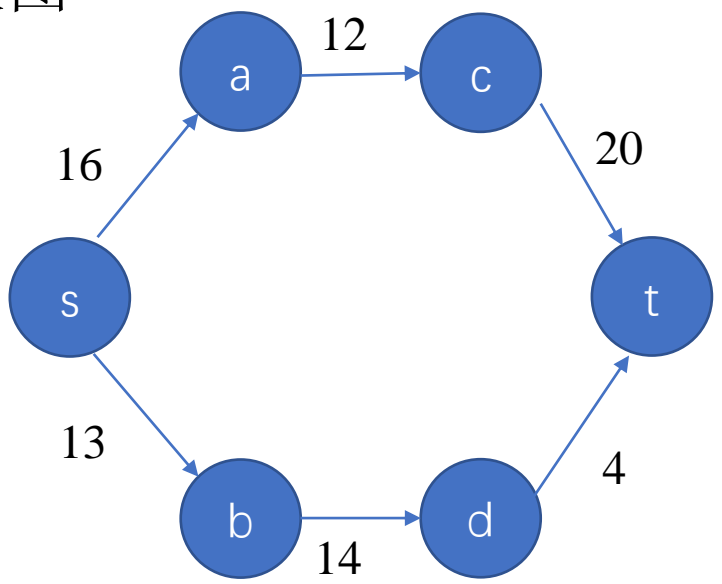
3. 给定如下剩余图，画出相应的层次图。



剩余图



层次图



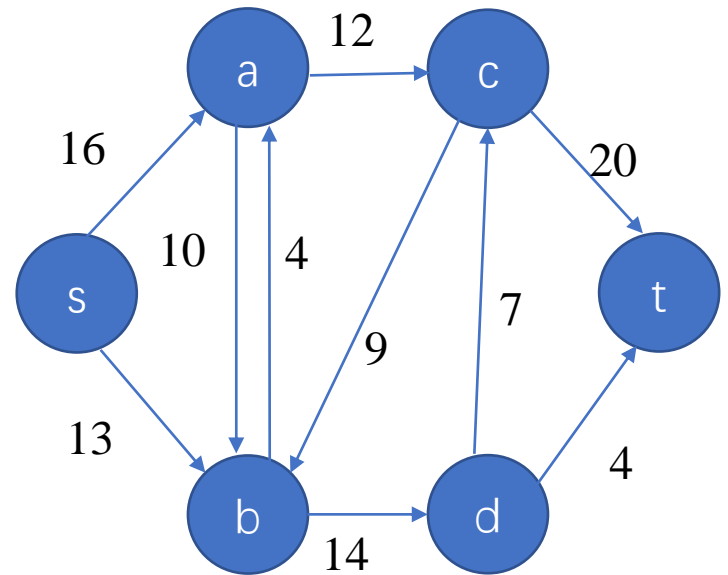
4. 给定如下网络，采用最小路径长度增值法（MPLA）计算最大流。

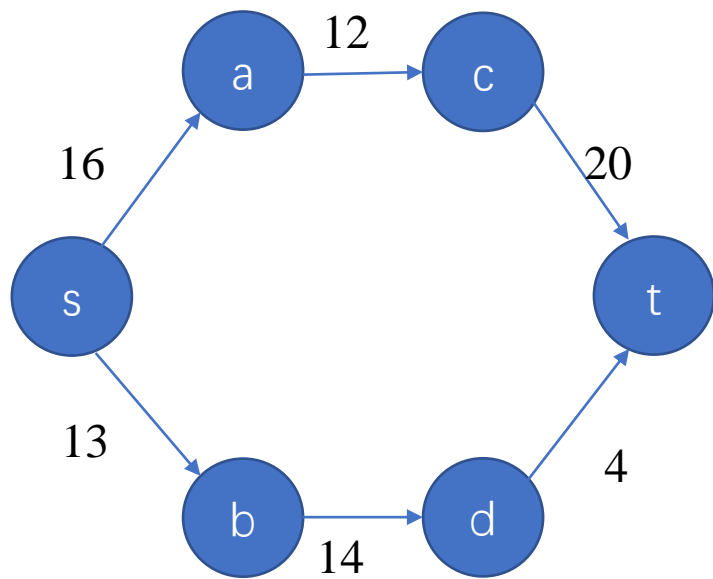
MPLA算法：

输入：网络 (G, s, t, c)

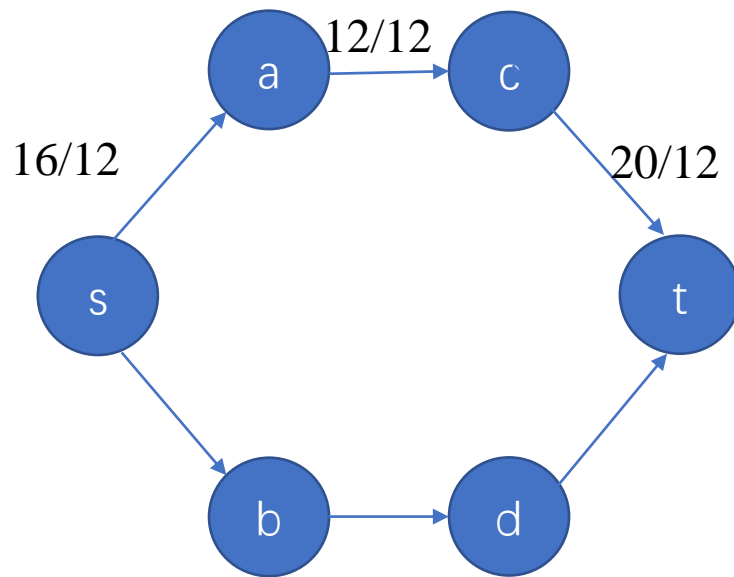
输出：G中的最大流

1. for 每条边 $(u, v) \in E$
2. $f(u, v) \leftarrow 0$
3. end for
4. 初始化剩余图，设 $R=G$
5. 查找R的层次图L
6. while t 为 L中的顶点
7. while t 在L中能从s到达
8. 设 p 为L中从s到t的一条路径
9. 设 Δ 为 p 的瓶颈容量
10. 用 Δ 增值当前流 f
11. 沿着路径 p 更新L和R
12. end while
13. 用剩余图R计算新的层次图L
14. end while

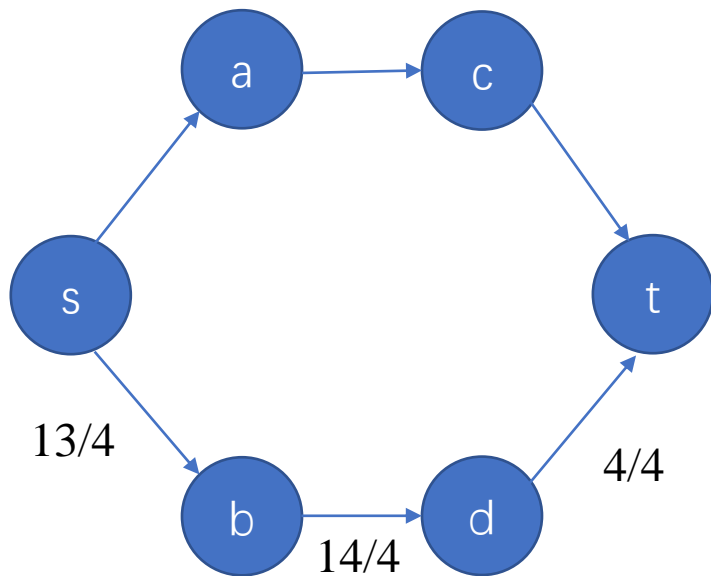




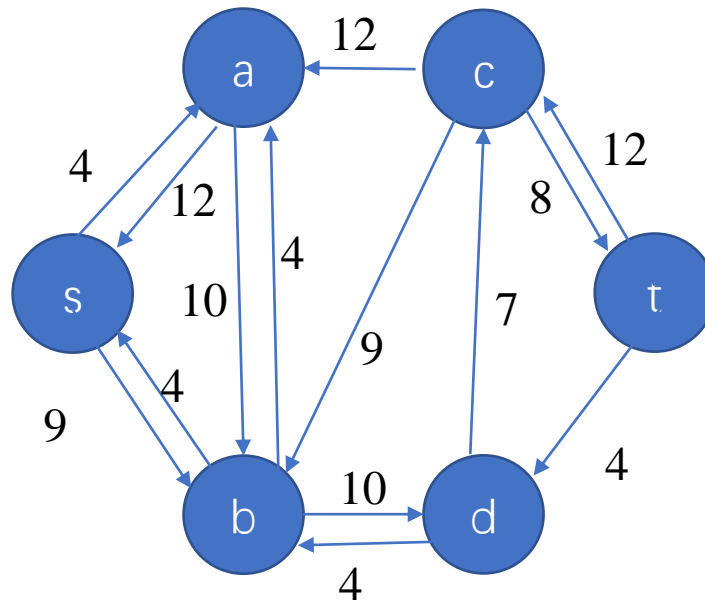
第一层次图



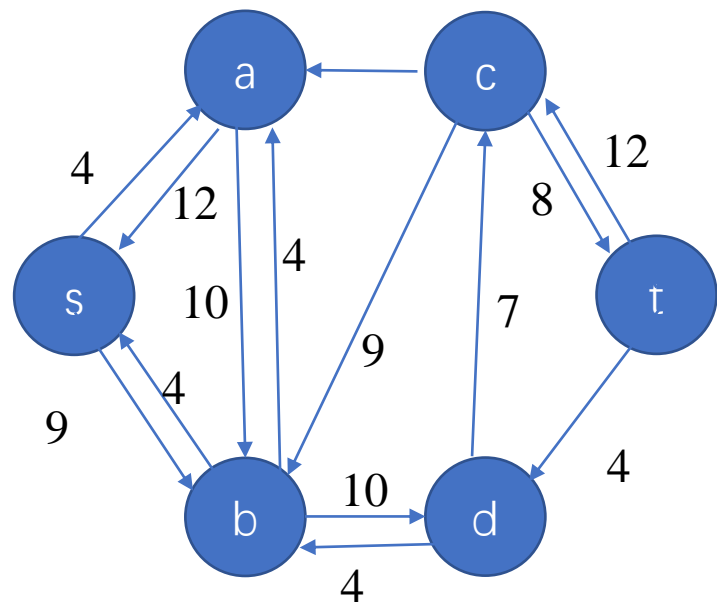
增值s, a, c, t



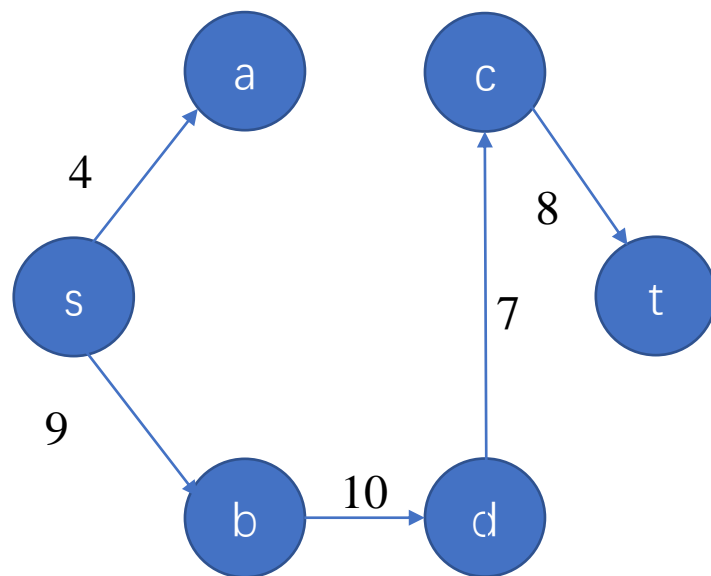
增值s, b, d, t



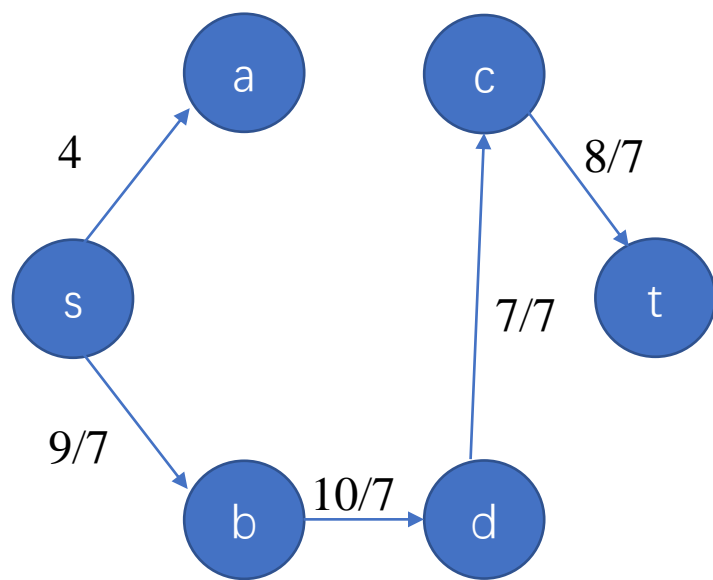
剩余图



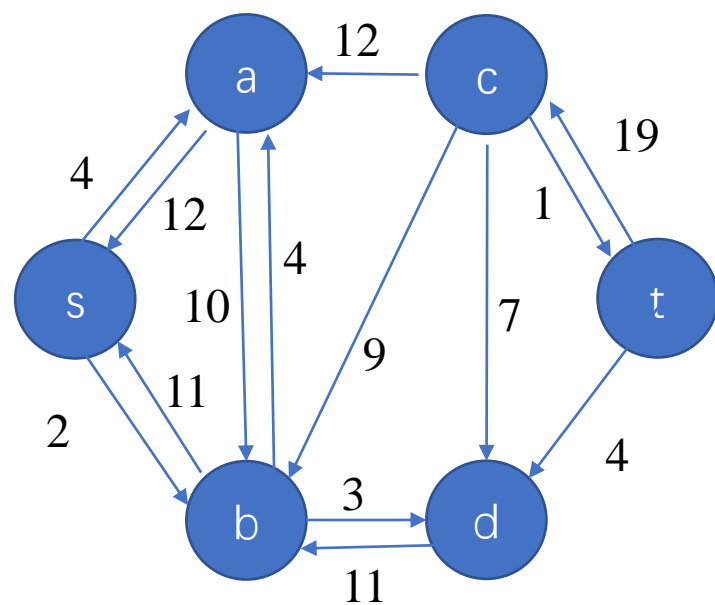
剩余图



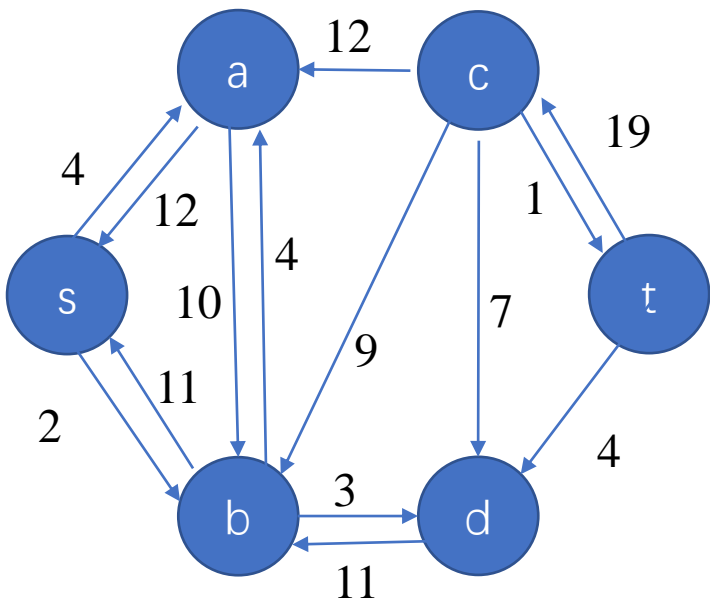
第二层次图



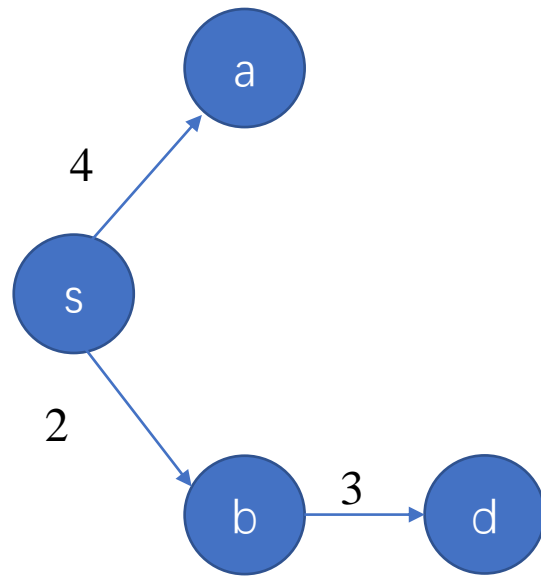
增值 s, b, d, c, t



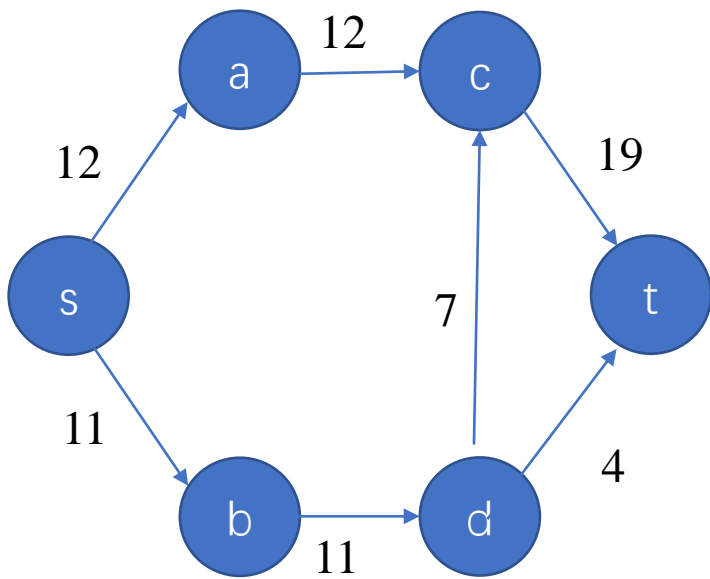
剩余图



剩余图



第三层次图



最后的流