091M4041H - Assignment 5

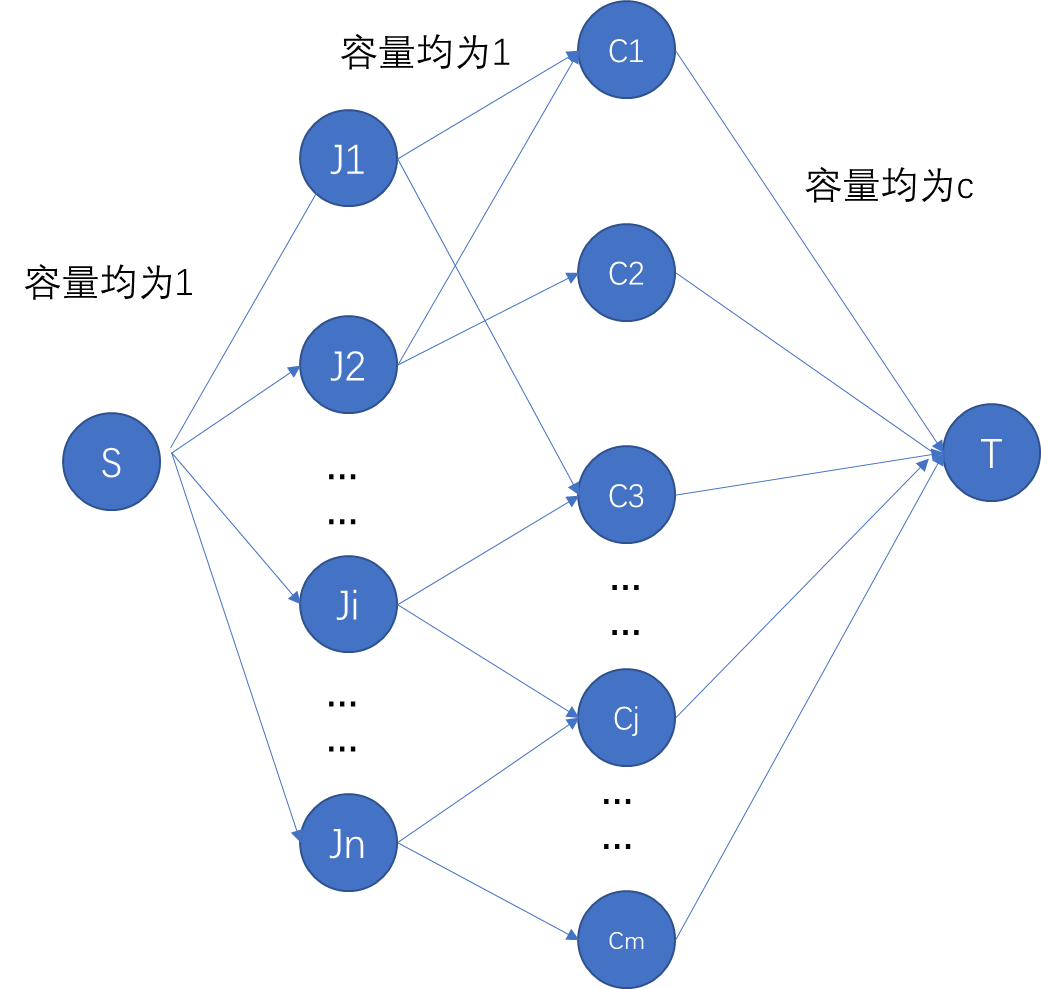
胡鹏飞 计控学院 201828007329009

1. Load balance

You have some different computers and jobs. For each job, it can only be done on one of two specified computers. The load of a computer is the number of jobs which have been done on the computer. Give the number of jobs and two computer ID for each job. You task is to minimize the max load.

* 1. **Describe in natural language**

假设有n个job以及m台computer，根据题意，每个job与两台computer之间构建连边，容量为1；有一个源点s与每个job都构建一个连边，容量也为1；所有computer都汇于一点，每条连边的容量为c，c为computer中单独一台的最大容量。如下图G所示



其中J1、J2……Jn表示job，C1、C2……Cm表示computer，S为源点，T为汇点，c为max load。

由题意可知c的范围为[1,n]，所以需要进行log n次最大流算法。

* 1. **Pseudo-code**

**MinMaxload(**G**)**

b**=**1**,**e**=**n

**while** b**<**e

c**=(**b**+**e**)/**2

**if** exist a flow f’**,**satisfy max\_flow**(**s**,**t**)==**n

e**=**c

**else**

b**=**c**+**1

**end** **if**

**end** **while**

**return** b

* 1. **Prove the correctness**

首先，对于节点J，流入的值为1，根据网络流特点，流出的值也为1，所以每个job最后只会在一台computer上工作，所以采用最大流算法，如果汇点T可以输入n，那么说明此时max load，也就是c偏大，继续缩小，直到汇点T的输入不能达到n，说明此时有一部分由于超过c而无法流到T，也就是找到了最小的c，即max load，算法无误。

* 1. **Analyse the complexity**

共需要log n次网络流，节点个数为n+m+2，连边条数为n+2n+m=3n+m，所以每次网络流算法时间复杂度为O[(n+m+2)(3n+m)]所以时间复杂度为：

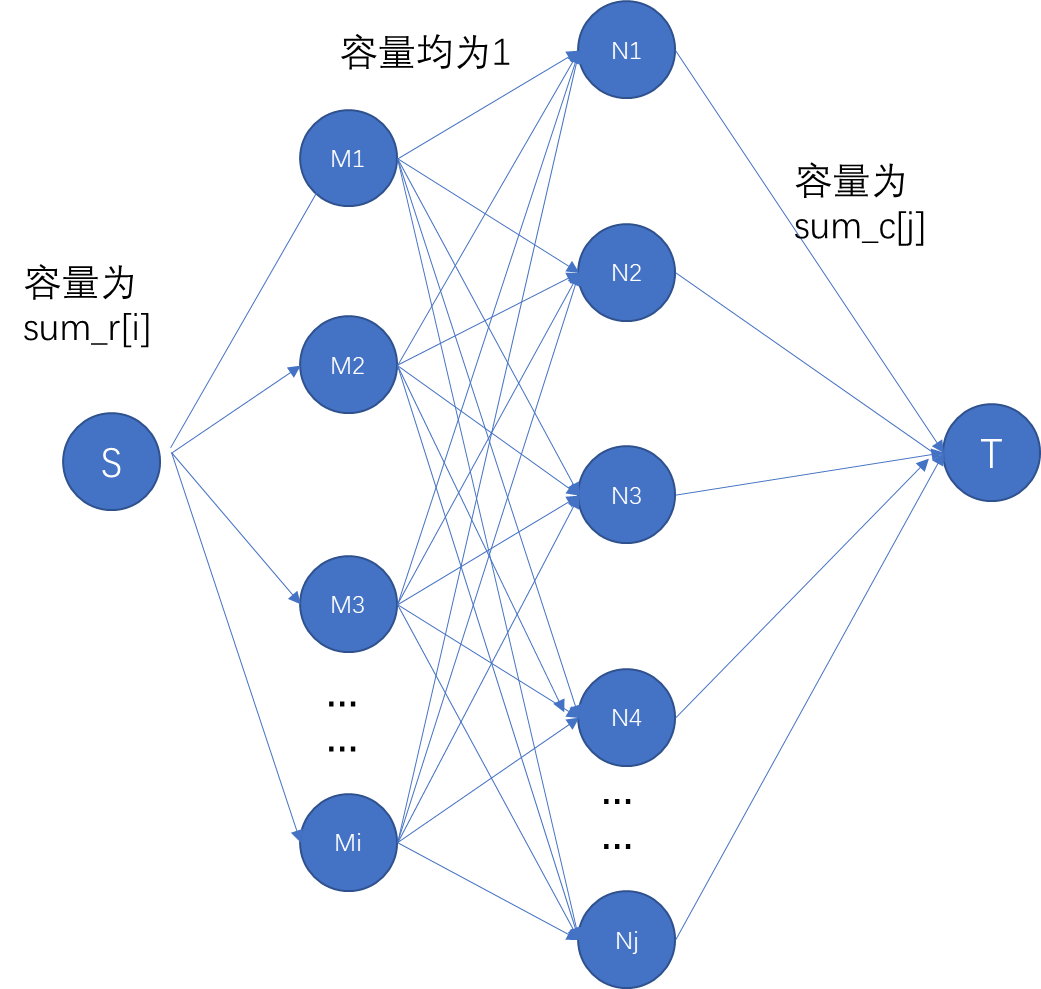


1. Matrix

For a matrix filled with 0 and 1, you know the sum of every row and column. You are asked to give such a matrix which satisfys the conditions.

* 1. **Describe in natural language**

假设矩阵有m行n列，每一行构造一个节点，有m个，每一列也构造一个节点，有n个，这m个节点与n个节点都互连，每条连边容量为1；有一个源点s与所有行节点都构建一个连边，容量为改行对应的和；所有列节点都汇于一点，每条连边的容量为该列的和。如下图G所示



跑最大流算法，如果最大流的结果等于各行的总和（各列的总和），则问题有解。

* 1. **Pseudo-code**

**Matrix(**m,n,sum\_r[m],sum\_c[n]**)**

constructing a graph G

running max\_flow Algorithm **in** G

**if** max\_flow**(**s**,**t**)** **==** sum of sum\_r **or** sum of sum\_c

**return** the final flow graph

**else**

**return** **False**

**end** **if**

* 1. **Prove the correctness**

源点到Mi的容量为第i行总和，保证了各行总流量不会超过sum\_r[i]，Nj到汇点的容量为第j列总和，保证了各列总流量sum\_c[j]，Mi到Nj的容量全部设为1，意味着只能为0或1，与矩阵每个位置只能填充0或1对应。因此，若最后最大流等于所有行或所有列总和，那么矩阵有解，返回求得的flow graph即可，算法无误。

* 1. **Analyse the complexity**

节点个数为n+m+2，连边条数为mn+m+n，所以所以时间复杂度为



4 Network Cost

For a network, there is one source and one sink. Every edge is directed and has two value c and a. c means the maximum flow of the edge. a is a coefficient number which means that if the flow of the edge is x, the cost is ax2.

Design an algorithm to get the Minimum Cost Maximum Flow.

**4.1 Describe in natural language**

如果每条边的费用固定，那么这个问题就是标准的最小割最大流问题，现在又多了一个系数a用来计算费用，且费用与此条边的流量有关，cost=ax2，但是可以将这个问题转化，可以将此处容量为n,费用为an2的一条边转换为n条边，每条边的容量为1，费用分别为a,3a,5a……(2n-1)a，再跑标准算法即可。

**4.2 Pseudo-code**

**MinCost\_MaxFlow(**G**):**

constructing a graph G' From G

running Min\_Cost\_Max\_Flow Algorithm **in** G’

**return** Mincost

**4.3 Prove the correctness**

证明图G与图G’等价即可，图G中一条容量为n，费用为an2的边转换为图G’中容量为1，费用分别为a,3a,5a……(2n-1)a的n条边，总容量仍为1\*n=n,费用仍为a+3a+5a+……+(2n-1)a= an2,所以两图等价，算法无误。

* 1. **Analyse the complexity**

寻找最短路径算法复杂度为O(VlogV),所以总时间复杂度为：

