

# AMATH 514 Assignment 5

Tyler Chen

**Note:** Note that I received a 26/30, however I have not made corrections

**Problem 4.5**

Let  $D = (V, A)$  be a directed graph, let  $s, t \in V$ . Let  $f : A \rightarrow \mathbb{R}_+$  be an  $s - t$  flow of value  $\beta$ . Show that there exists an  $s - t$  flow  $f' : A \rightarrow \mathbb{Z}_+$  of value  $\lceil \beta \rceil$  such that  $\lfloor f(x) \rfloor \leq f'(a) \leq \lceil f(a) \rceil$  for each  $a$ .

We provide an algorithm to find the flow  $f' : A \rightarrow \mathbb{Z}_+$ .

**Input:** directed graph  $D = (V, A)$  and  $s - t$  flow  $f : A \rightarrow \mathbb{R}_+$  of total value  $\beta$ .

**Output:**  $s - t$  flow  $f' : A \rightarrow \mathbb{R}_+$  with  $\lfloor f(x) \rfloor \leq f'(a) \leq \lceil f(a) \rceil$  for each  $a$  and total value  $\lceil \beta \rceil$ .

**Description:** Let  $\beta$  be the value of the  $s - t$  flow  $f$ . If  $\beta \in \mathbb{Z}$  terminate and return  $f : A \rightarrow \mathbb{R}_+$ .

Otherwise, if  $\beta \notin \mathbb{Z}$ , define an auxiliary graph  $D_f = (V, A_f)$ , where  $A_f$  is constructed by:

$$\text{if } f(a) \notin \mathbb{Z}, \text{ then } a, a^{-1} \in A_f$$

Suppose there is no  $s - t$  path in  $D_f$ . Let  $U = \{u \in V : \exists s - u \text{ path}\}$ . By our construction of  $D_f$  this means  $f(\delta^{\text{out}}(U)) = \sum_{u \in \delta^{\text{out}}(U)} f(u)$  is integer as for each  $u \in \delta^{\text{out}}(U)$ ,  $f(u) \in \mathbb{Z}$ . The flow of this cut is exactly the value of the flow  $f : A \rightarrow \mathbb{R}_+$ .

Since our flow is not integer, there must be a  $s - t$  path in  $D_f$ .

Define  $c : A_f \rightarrow \mathbb{R}_+$  by,

$$c(a) = \begin{cases} \lceil f(a) \rceil - f(a) & a \in A \\ f(a^{-1}) - \lfloor f(a^{-1}) \rfloor & a^{-1} \in A \end{cases}$$

Since every  $a \in A_f$  is either an arc in  $A$  or the inverse of an arc in  $A$  the above description is well defined.

We now adjust the value of  $f$  by as much as possible while still keeping  $f$  between  $\lfloor f(a) \rfloor$  and  $\lceil f(a) \rceil$ . To this end define,

$$\alpha = \min_{a \in P} \{c(a)\}$$

Now define  $f' : A \rightarrow \mathbb{R}_+$  by,

$$f'(a) = \begin{cases} f(a) + \alpha & a \in P \\ f(a) & a \notin P \end{cases}$$

Then clearly  $f' : A \rightarrow \mathbb{R}_+$  is still a flow. By our choice of  $\alpha$  we have,  $\lfloor f(a) \rfloor \leq f'(a) \leq \lceil f(a) \rceil$  for all  $a \in A$ . Moreover, for at least one  $a \in A$  we have  $f'(a) = \lfloor f(a) \rfloor$  or  $f'(a) = \lceil f(a) \rceil$ . In particular this means that there are strictly fewer non-integer arc values under  $f'$  than  $f$ .

Now repeat this algorithm with  $D$  and  $f'$  as inputs.

Since there is at least one more integer weight edge in the flow  $f'$  than in  $f$ , the algorithm will eventually terminate.

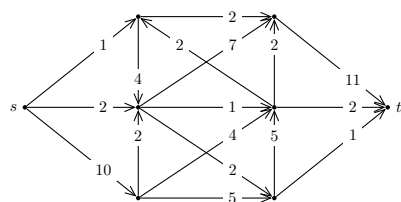
At each step the flow on an edge  $a$  never became lower than  $\lfloor f(a) \rfloor$  and never became greater than  $\lceil f(a) \rceil$ . This means that the final output of the algorithm,  $f' : A \rightarrow \mathbb{R}_+$  will satisfy,

$$\lfloor f(a) \rfloor \leq f'(a) \leq \lceil f(a) \rceil$$

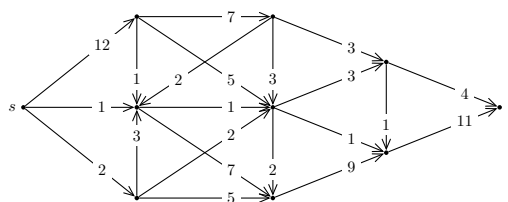
and since the algorithm only halts when the value of the input flow is integer and the flow value in each step is strictly increasing the final flow is of value  $\lceil \beta \rceil$ .  $\square$

**Problem 4.7**

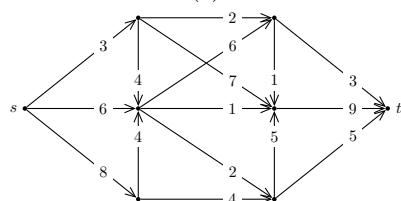
Determine with the maximum flow algorithm an  $s-t$  flow of maximum value and a  $s-t$  cut of minimum capacity in the following graphs:



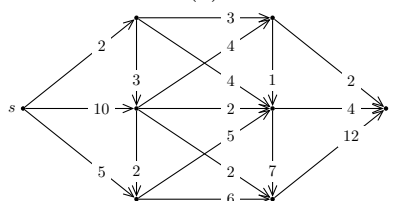
(a)



(b)

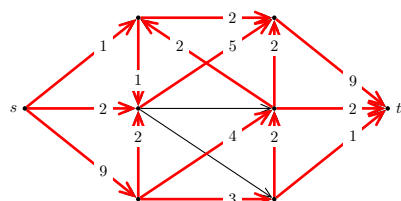


(c)

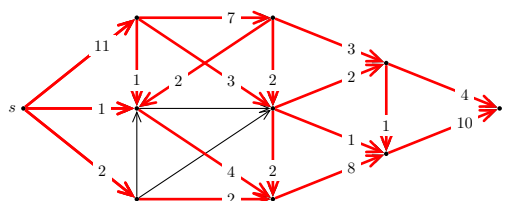


(d)

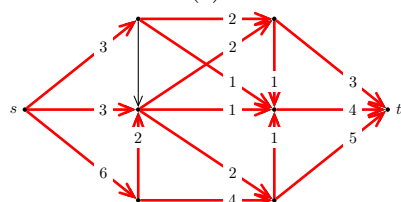
We compute max-flows of size: 12, 14, 12, and 14, respectively. The flows found are pictured below. Since the graphs aren't too large we first produced a flow by hand. The algorithm was then applied, and generally terminated in one or two steps since the initial flows were reasonably large.



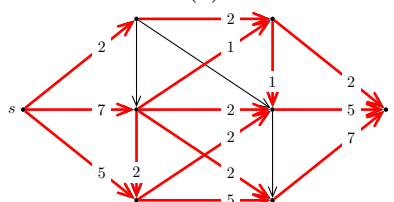
(a)



(b)



(c)



(d)

---

**Problem 4.9**

Describe the problem of finding a maximum-size matching in a bipartite graph as a maximum flow problem.

---

Let  $G = (V, E)$  be a bipartite where  $V = U \cup W$  and  $E \subseteq \{\{u, v\} : u \in U, v \in V\}$ .

Let  $V' = V \cup \{s, t\}$ . Let  $E' = \{(u, v) : \{u, v\} \in E\} \cup \{\{s, u\} : u \in U\} \cup \{\{v, t\} : v \in V\}$ .

Let  $c(e) : E' \rightarrow \mathbb{R}$  be defined as  $c(e) = 1$ .

We claim an integer max flow on  $G'$  gives a maximum-size matching on  $G$  by showing flows in  $G'$  are matchings in  $G$ .

Indeed, let  $u \in U$ . By construction the flow into  $u$  is at most one so the flow out of  $u$  can be at most one. Since we have restricted the flows to integers, there can be at most one edge out of  $u$  in the flow, and it must have value one or zero.

Similarly, let  $v \in V$ . By construction the flow out of  $v$  is at most one so the flow into  $v$  can be at most one. Since we have restricted the flow to integers, there can be at most one edge into  $v$  in the flow, and it must have value one or zero.

Then clearly taking all edges in  $G'$  between  $U$  and  $V$  with nonzero flow gives a matching in  $U$ .

Let  $M \subseteq E$  be a matching on  $G$ . Then we can define a flow by  $f(e) = 1$  if any vertex of  $e$  is in an edge of  $M$  and  $f(e) = 0$  if no vertex of  $e$  is in any edge of  $M$ . Obviously for all  $u \in U$  and  $v \in V$  the flow in is equal to the flow out (there is either one edge in and one edge out, or no edges in and no edges out).

Note that the weight of the flow is equal to the size of the matching. This proves the result.  $\square$