3-13 Flow

Jun Ma

majun@nju.edu.cn

December 22, 2020

TC 26.1

Show that splitting an edge in a flow network yields an equivalent network. More formally, suppose that flow network G contains edge (u, v), and we create a new flow network G' by creating a new vertex x and replacing (u, v) by new edges (u, x) and (x, v) with $c(u, x) = c(x, v) = c(u, v)$. Show that a maximum flow in G' has the same value as a maximum flow in G.

TC 26.1

Let F and F' denote a maximum flow of G and G' accordingly.
Let f and f' denote any flow of G and G' accordingly.

TC 26.1

Let F and F' denote a maximum flow of G and G' accordingly.
Let f and f' denote any flow of G and G' accordingly.

$$\boxed{|F| = |F'|}$$

TC 26.1

Let F and F' denote a maximum flow of G and G' accordingly.
Let f and f' denote any flow of G and G' accordingly.

$$\boxed{|F| = |F'|}$$

$$\Updownarrow$$

$$\boxed{|F| \geq |F'| \text{ and } |F| \leq |F'|}$$

TC 26.1

Let F and F' denote a maximum flow of G and G' accordingly.
Let f and f' denote any flow of G and G' accordingly.

$$\boxed{|F| = |F'|}$$

$$\Updownarrow$$

$$\boxed{|F| \geq |F'| \text{ and } |F| \leq |F'|}$$

$$\Updownarrow$$

$$\boxed{|F| \geq |f'| \text{ for all } f' \text{ and } |f| \leq |F'| \text{ for all } f}$$

$|F| \geq |f'|$ for all f'

$|F| \geq |f'|$ for all f'

$$\boxed{|F| \geq |f'| \text{ for all } f'}$$

$$\Uparrow$$

$$\boxed{\text{for all } f', \text{ we could obtain a flow } f \text{ in } G, \text{ s.t. } |f| = |f'|}$$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- Let f' be a flow in G'. Obviously, $f'(u,x) = f'(x,v)$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- Let f' be a flow in G'. Obviously, $f'(u, x) = f'(x, v)$
- We could obtain a **flow?** f of G by:
  - For every edge $(a, b) \in G.E, (a, b) \neq (u, v)$, let $f(a, b) = f'(a, b)$
  - For the edge $(u, v) \in G.E$, let $f(u, v) = f'(u, x) = f'(x, v)$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- Let f' be a flow in G'. Obviously, $f'(u,x) = f'(x,v)$
- We could obtain a **flow?** f of G by:
  - For every edge $(a,b) \in G.E, (a,b) \neq (u,v)$, let $f(a,b) = f'(a,b)$
  - For the edge $(u,v) \in G.E$, let $f(u,v) = f'(u,x) = f'(x,v)$

Is f really a flow?

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- Let f' be a flow in G'. Obviously, $f'(u,x) = f'(x,v)$
- We could obtain a **flow?** f of G by:
  - For every edge $(a,b) \in G.E, (a,b) \neq (u,v)$, let $f(a,b) = f'(a,b)$
  - For the edge $(u,v) \in G.E$, let $f(u,v) = f'(u,x) = f'(x,v)$

Is f really a flow?

We are now ready to define flows more formally. Let $G = (V, E)$ be a flow network with a capacity function c. Let s be the source of the network, and let t be the sink. A ***flow*** in G is a real-valued function $f : V \times V \to \mathbb{R}$ that satisfies the following two properties:

Capacity constraint: For all $u, v \in V$, we require $0 \leq f(u,v) \leq c(u,v)$.

Flow conservation: For all $u \in V - \{s,t\}$, we require

$$\sum_{v \in V} f(v,u) = \sum_{v \in V} f(u,v) .$$

When $(u,v) \notin E$, there can be no flow from u to v, and $f(u,v) = 0$.

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- **Capacity constraint:** Obviously, for every edge $(a, b) \in G.E$, $0 \leq f(a, b) \leq c(a, b)$

Given any f', obtain a flow f in G, s.t. $|f| = |f'|$

- **Capacity constraint:** Obviously, for every edge $(a, b) \in G.E$, $0 \leq f(a, b) \leq c(a, b)$
- For **Flow conservation**, we first prove that for all vertex $a \in G.V$

$$\sum_{b \in G.V} f(a, b) = \sum_{b \in G'.V} f'(a, b)$$

$$\sum_{b \in G.V} f(b, a) = \sum_{b \in G'.V} f'(b, a)$$

$$\sum_{b \in G.V} f(a,b) = \sum_{b \in G'.V} f'(a,b)$$

- **Case 1** ($a \neq u$): obviously

$\sum_{b\in G.V} f(a,b) = \sum_{b\in G'.V} f'(a,b)$

- **Case 1** $(a \neq u)$: obviously
- **Case 2** $(a = u)$:

$$\begin{aligned}\sum_{b\in G.V} f(a,b) &= \sum_{b\in G.V-\{v\}} f(u,b) + f(u,v)\\ &= \sum_{b\in G'.V-\{v,x\}} f'(u,b) + f'(u,x)\\ &= \sum_{b\in G'.V-\{v\}} f'(u,b)\\ &= \sum_{b\in G'.V} f'(u,b)\end{aligned}$$

$\sum_{b\in G.V} f(a,b) = \sum_{b\in G'.V} f'(a,b)$

- **Case 1** ($a \neq u$): obviously
- **Case 2** ($a = u$):

$$\begin{aligned}\sum_{b\in G.V} f(a,b) &= \sum_{b\in G.V-\{v\}} f(u,b) + f(u,v)\\ &= \sum_{b\in G'.V-\{v,x\}} f'(u,b) + f'(u,x)\\ &= \sum_{b\in G'.V-\{v\}} f'(u,b)\\ &= \sum_{b\in G'.V} f'(u,b)\end{aligned}$$

Similarly

$$\sum_{b\in G.V} f(b,a) = \sum_{b\in G'.V} f'(b,a)$$

Flow conservation

$$\boxed{\sum_{b\in G.V} f(a,b) = \sum_{b\in G'.V} f'(a,b) \text{ and } \sum_{b\in G.V} f(b,a) = \sum_{b\in G'.V} f'(b,a)}$$

$$\Downarrow$$

$$\boxed{\text{For all } a \in V - \{s,t\} \sum_{b\in G.V} f(b,a) = \sum_{b\in G.V} f(a,b)}$$

Flow value

$$\sum_{b\in G.V} f(a,b) = \sum_{b\in G'.V} f'(a,b) \text{ and } \sum_{b\in G.V} f(b,a) = \sum_{b\in G'.V} f'(b,a)$$

$$\Downarrow$$

$$\begin{aligned} |f| &= \sum_{b\in G.V} f(b,s) - \sum_{b\in G.V} f(s,b) \\ &= \sum_{b\in G'.V} f'(b,s) - \sum_{b\in G'.V} f'(s,b) \\ &= |f'| \end{aligned}$$

Flow value

$$\sum_{b \in G.V} f(a,b) = \sum_{b \in G'.V} f'(a,b) \text{ and } \sum_{b \in G.V} f(b,a) = \sum_{b \in G'.V} f'(b,a)$$

$$\Downarrow$$

$$\begin{aligned} |f| &= \sum_{b \in G.V} f(b,s) - \sum_{b \in G.V} f(s,b) \\ &= \sum_{b \in G'.V} f'(b,s) - \sum_{b \in G'.V} f'(s,b) \\ &= |f'| \end{aligned}$$

So, given any f' in G', we could obtain a flow f in G, s.t.

$$|f| = |f'|$$

Flow value

$$\sum_{b \in G.V} f(a,b) = \sum_{b \in G'.V} f'(a,b) \text{ and } \sum_{b \in G.V} f(b,a) = \sum_{b \in G'.V} f'(b,a)$$

$$\Downarrow$$

$$\begin{aligned} |f| &= \sum_{b \in G.V} f(b,s) - \sum_{b \in G.V} f(s,b) \\ &= \sum_{b \in G'.V} f'(b,s) - \sum_{b \in G'.V} f'(s,b) \\ &= |f'| \end{aligned}$$

So, given any f' in G', we could obtain a flow f in G, s.t. $|f| = |f'|$

Similarly, given any f in G, we could obtain a flow f' in G', s.t. $|f'| = |f|$

TC 26.1-2

Extend the flow properties and definitions to the multiple-source, multiple-sink problem. Show that any flow in a multiple-source, multiple-sink flow network corresponds to a flow of identical value in the single-source, single-sink network obtained by adding a supersource and a supersink, and vice versa.

TC 26.1-2

$$S \subset V : \text{a set of sources}$$
$$T \subset V : \text{a set of targets}$$

We are now ready to define flows more formally. Let $G = (V, E)$ be a flow network with a capacity function c. Let s be the source of the network, and let t be the sink. A ***flow*** in G is a real-valued function $f : V \times V \to \mathbb{R}$ that satisfies the following two properties:

Capacity constraint: For all $u, v \in V$, we require $0 \le f(u,v) \le c(u,v)$.

Flow conservation: For all $u \in V - S - T$ we require

$$\sum_{v \in V} f(v,u) = \sum_{v \in V} f(u,v) .$$

When $(u,v) \notin E$, there can be no flow from u to v, and $f(u,v) = 0$.

TC 26.1-6

Professor Adam has two children who, unfortunately, dislike each other. The problem is so severe that not only do they refuse to walk to school together, but in fact each one refuses to walk on any block that the other child has stepped on that day. The children have no problem with their paths crossing at a corner. Fortunately both the professor's house and the school are on corners, but beyond that he is not sure if it is going to be possible to send both of his children to the same school. The professor has a map of his town. Show how to formulate the problem of determining whether both his children can go to the same school as a maximum-flow problem.

TC 26.1-6

- Build a flow network $G(V, E)$ as following

TC 26.1-6

- Build a flow network $G(V, E)$ as following
  - For each corner, create a vertex v in $G.V$

TC 26.1-6

- Build a flow network $G(V, E)$ as following
  - For each corner, create a vertex v in $G.V$
  - If there is an street between two corners, add two edges $v \to u$ and $u \to v$ to $G.E$

TC 26.1-6

- Build a flow network $G(V, E)$ as following
  - For each corner, create a vertex v in $G.V$
  - If there is an street between two corners, add two edges $v \to u$ and $u \to v$ to $G.E$
  - Let s be the vertex corresponding to the house corner, t be the vertex corresponding to the school corner.

TC 26.1-6

- Build a flow network $G(V, E)$ as following
  - For each corner, create a vertex v in $G.V$
  - If there is an street between two corners, add two edges $v \to u$ and $u \to v$ to $G.E$
  - Let s be the vertex corresponding to the house corner, t be the vertex corresponding to the school corner.
  - All edges have capacities of 1.

TC 26.1-6

- Build a flow network $G(V, E)$ as following
  - For each corner, create a vertex v in $G.V$
  - If there is an street between two corners, add two edges $v \to u$ and $u \to v$ to $G.E$
  - Let s be the vertex corresponding to the house corner, t be the vertex corresponding to the school corner.
  - All edges have capacities of 1.
- Check if there is a flow with value greater than 1.

TC 26.1-7

Suppose that, in addition to edge capacities, a flow network has ***vertex capacities***. That is each vertex v has a limit $l(v)$ on how much flow can pass though v. Show how to transform a flow network $G = (V, E)$ with vertex capacities into an equivalent flow network $G' = (V', E')$ without vertex capacities, such that a maximum flow in G' has the same value as a maximum flow in G. How many vertices and edges does G' have?

TC 26.1-7

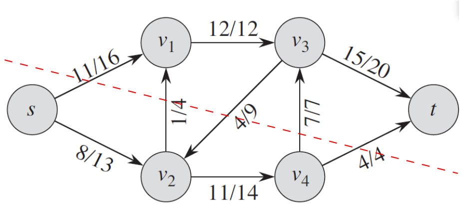
- For each vertex $v \in V$, divide it into a pair of two vertices v_1 and v_2 and add an edge (v_1, v_2) and set $c_{G'}(v_1, v_2) = l(v)$
- For each edge $(u, v) \in E$, replace it with an edge (u_2, v_1) with $c_{G'}(u_2, v_1) = c_G(u, v)$
- Let s_1 be the new source, and t_2 be the new target.

TC 26.2-2

In Figure 26.1(b), what is the flow across the cut $(\{s, v_2, v_4\}, \{v_1, v_3, t\})$? What is the capacity of this cut?

TC 26.2-2

In Figure 26.1(b), what is the flow across the cut $(\{s, v_2, v_4\}, \{v_1, v_3, t\})$? What is the capacity of this cut?



(b)

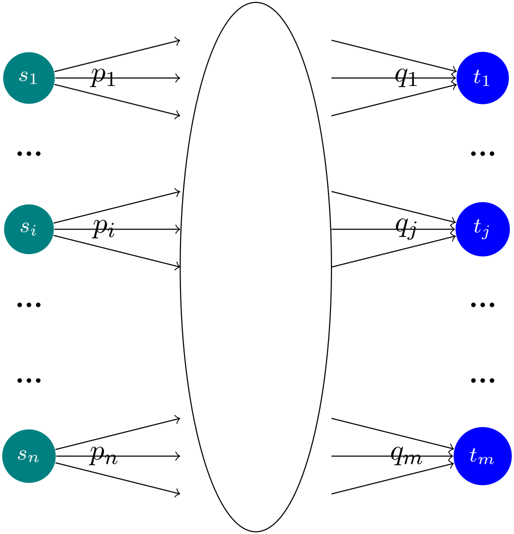
$$\text{flow:}(11 + 1 + 7 + 4) - 4 = 19$$

$$\text{capacity:}16 + 4 + 7 + 4 = 31$$

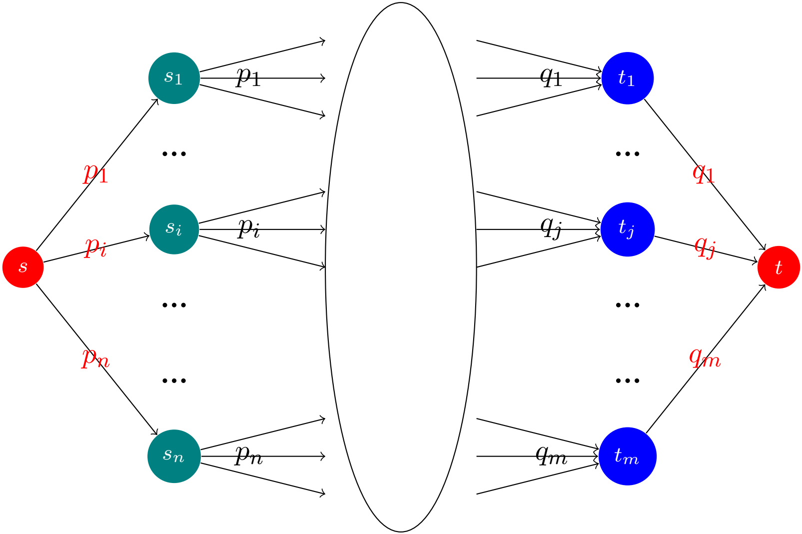
TC 26.2-6

Suppose that each source s_i in a flow network with multiple sources and sinks produces exactly p_i units of flow, so that $\sum_{v \in V} f(s_i, v) = p_i$. Suppose also that each sink t_j consumes exactly q_j units, so that $\sum_{v \in V} f(v, t_j) = q_j$, where $\sum_i p_i = \sum_j q_j$. Show how to convert the problem of finding a flow f that obeys these additional constraints into the problem of finding a maximum flow in a single-source, single-sink flow network.

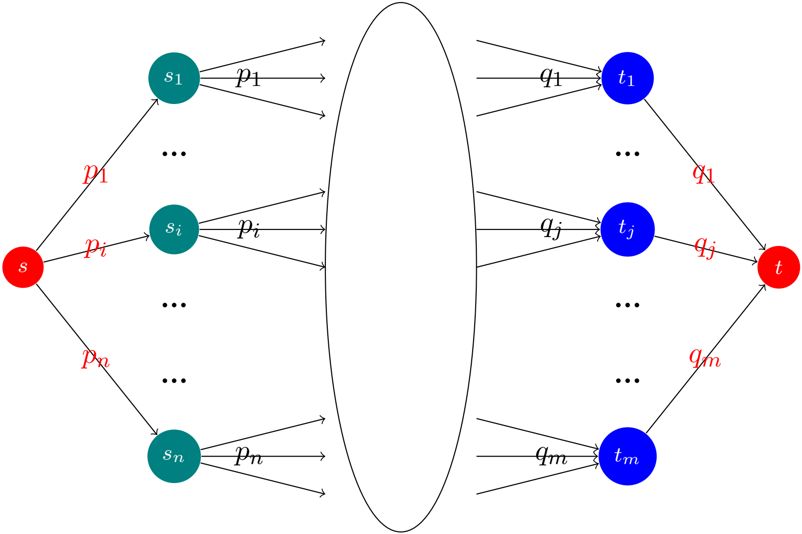
TC 26.2-6



TC 26.2-6



TC 26.2-6



check if the value of a maximum flow f equals $\sum_i p_i$

TC 26.2-8

Suppose that we redefine the residual network to disallow edges into s. Argue that the procedure FORD-FULKERSON still correctly computes a maximum flow.

```
FORD-FULKERSON(G, s, t)
for each edge (u, v) ∈ G.E
    (u, v).f = 0
while there exists a path p from s to t in the residual network G_f
    c_f(p) = min {c_f(u, v) : (u, v) is in p}
    for each edge (u, v) in p
        if (u, v) ∈ E
            (u, v).f = (u, v).f + c_f(p)
        else (v, u).f = (v, u).f − c_f(p)
```

TC 26.2-8

Suppose that we redefine the residual network to disallow edges into s. Argue that the procedure FORD-FULKERSON still correctly computes a maximum flow.

```
FORD-FULKERSON(G, s, t)
for each edge (u, v) ∈ G.E
    (u, v).f = 0
while there exists a path p from s to t in the residual network G_f
    c_f(p) = min {c_f(u, v) : (u, v) is in p}
    for each edge (u, v) in p
        if (u, v) ∈ E
            (u, v).f = (u, v).f + c_f(p)
        else (v, u).f = (v, u).f − c_f(p)
```

Removing edges into s would not affect any simple path from s to t in the residual network.

TC 26.2-10

Show how to find a maximum flow in a network $G = (V, E)$ by a sequence of at most $|E|$ augmenting paths. (*Hint:* Determine the paths *after* finding the maximum flow.)

TC 26.2-10

TC 26.2-10

- Run Ford-Fulkerson to G to obtain an initial maximum flow f;

TC 26.2-10

- Run FORD-FULKERSON to G to obtain an initial maximum flow f;
- Build a flow network G', s.t. $G'.V = G.V$ and $c(u, v) = f(u, v)$;

TC 26.2-10

- Run Ford-Fulkerson to G to obtain an initial maximum flow f;
- Build a flow network G', s.t. $G'.V = G.V$ and $c(u, v) = f(u, v)$;
- Run Ford-Fulkerson, with the **modification** that we remove an edge if its flow reaches its capacity.

TC 26.2-10

- Run FORD-FULKERSON to G to obtain an initial maximum flow f;
- Build a flow network G', s.t. $G'.V = G.V$ and $c(u, v) = f(u, v)$;
- Run FORD-FULKERSON, with the **modification** that we remove an edge if its flow reaches its capacity.
- The augmenting paths chosen in this modified version of FORD-FULKERSON are precisely the ones we want.

TC 26.2-10

- Run FORD-FULKERSON to G to obtain an initial maximum flow f;
- Build a flow network G', s.t. $G'.V = G.V$ and $c(u,v) = f(u,v)$;
- Run FORD-FULKERSON, with the **modification** that we remove an edge if its flow reaches its capacity.
- The augmenting paths chosen in this modified version of FORD-FULKERSON are precisely the ones we want.
- Every augmenting path produces at least one edge whose flow is equal to its capacity. Therefore, at most $|E|$ paths.

TC 26.2-12

Suppose that you are given a flow network G, and G has edges entering the source s. Let f be a flow in G in which one of the edges (v, s) entering the source has $f(v, s) = 1$. Prove that there must exist another flow f' with $f'(v, s) = 0$ such that $|f| = |f'|$. Give an $O(E)$-time algorithm to compute f', given f, and assuming that all edge capacities are integers.

TC 26.2-12

Part-1

Prove that there must exist another flow f' with $f'(v, s) = 0$ such that $|f| = |f'|$.

Proof.

TC 26.2-12

Part-1

Prove that there must exist another flow f' with $f'(v, s) = 0$ such that $|f| = |f'|$.

Proof.

- As $f(v, s) = 1$, there must be a cycle $C : v_0 \to v_1 \to v_2 \to \cdots \to v_{n-1} \to v_n$, where $v_0 = v_n = s, v_{n-1} = v$ and $f(v_i, v_{i+1} \geq 1)$ for all $i = \{0, \cdots, n-1\}$

TC 26.2-12

Part-1

Prove that there must exist another flow f' with $f'(v, s) = 0$ such that $|f| = |f'|$.

Proof.

- As $f(v, s) = 1$, there must be a cycle $C : v_0 \to v_1 \to v_2 \to \cdots \to v_{n-1} \to v_n$, where $v_0 = v_n = s, v_{n-1} = v$ and $f(v_i, v_{i+1} \geq 1)$ for all $i = \{0, \cdots, n-1\}$
- Decreasing $f(v_i, v_{i+1})$ by 1 for all $i = \{0, \cdots, n-1\}$ results a new flow f'.

TC 26.2-12

Part-1

Prove that there must exist another flow f' with $f'(v, s) = 0$ such that $|f| = |f'|$.

Proof.

- As $f(v, s) = 1$, there must be a cycle $C : v_0 \to v_1 \to v_2 \to \cdots \to v_{n-1} \to v_n$, where $v_0 = v_n = s, v_{n-1} = v$ and $f(v_i, v_{i+1} \geq 1)$ for all $i = \{0, \cdots, n-1\}$
- Decreasing $f(v_i, v_{i+1})$ by 1 for all $i = \{0, \cdots, n-1\}$ results a new flow f'.
- Obviously

$$
\begin{aligned}
|f'| &= \sum_{v \in V} f'(s, u) - \sum_{u \in V} f'(u, s) \\
&= \sum_{u \in V - \{v_1\}} f'(s, u) + f'(s, v_1) - \sum_{u \in V - \{v\}} f'(u, s) - f'(v, s) \\
&= \sum_{u \in V - \{v_1\}} f(s, u) + (f(s, v_1) - 1) - \sum_{u \in V - \{v\}} f(u, s) - (f(v, s) - 1) \\
&= \sum_{v \in V} f(s, u) - \sum_{u \in V} f(u, s) = |f|
\end{aligned}
$$

TC 26.2-12

Part-2

Give an $O(E)$-time algorithm to compute f', given f.

TC 26.2-12

Part-2

Give an $O(E)$-time algorithm to compute f', given f.

- Build a directed graph G', where $(x, y) \in G'.E$ if and only if $f(y, x) > 0$ and $G'.V$ is the set of vertices of all edges in $G'.E$.

TC 26.2-12

Part-2

Give an $O(E)$-time algorithm to compute f', given f.

- Build a directed graph G', where $(x, y) \in G'.E$ if and only if $f(y, x) > 0$ and $G'.V$ is the set of vertices of all edges in $G'.E$.
- Start from v, run DFS to find a path P to s in G'.

TC 26.2-12

Part-2

Give an $O(E)$-time algorithm to compute f', given f.

- Build a directed graph G', where $(x, y) \in G'.E$ if and only if $f(y, x) > 0$ and $G'.V$ is the set of vertices of all edges in $G'.E$.
- Start from v, run DFS to find a path P to s in G'.
- For each edge $(x, y) \in P$, decrease $f(y, x)$ by 1; and finally, decrease $f(v, s)$ by one.

TC 26.2-13

Suppose that you wish to find, among all minimum cuts in a flow network G with integral capacities, one that contains the smallest number of edges. Show how to modify the capacities of G to create a new flow network G' in which any minimum cut in G' is a minimum cut with the smallest number of edges in G.

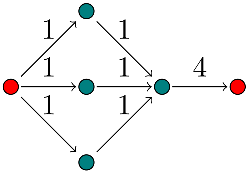
TC 26.2-13

Suppose that you wish to find, among all minimum cuts in a flow network G with integral capacities, one that contains the smallest number of edges. Show how to modify the capacities of G to create a new flow network G' in which any minimum cut in G' is a minimum cut with the smallest number of edges in G.

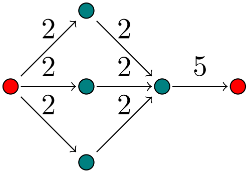
$$\boxed{c(u,v) = c(u,v) + 1}$$

Is it OK? ✗

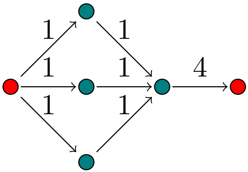
TC 26.2-13



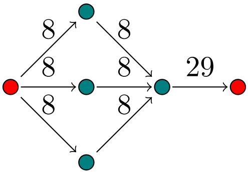
$$\Downarrow c(u, v) = c(u, v) + 1$$



TC 26.2-13

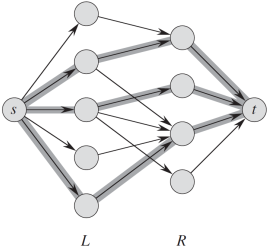


$$\Downarrow c(u,v) = c(u,v) * |E| + 1$$



TC 26.3-3

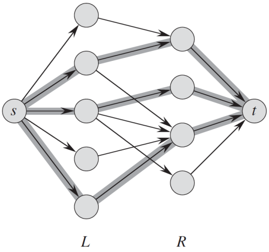
Let $G = (V, E)$ be a bipartite graph with vertex partition $V = L \cup R$, and let G' be its corresponding flow network. Give a good upper bound on the length of any augmenting path found in G' during the execution of FORD-FULKERSON.



(c)

TC 26.3-3

Let $G = (V, E)$ be a bipartite graph with vertex partition $V = L \cup R$, and let G' be its corresponding flow network. Give a good upper bound on the length of any augmenting path found in G' during the execution of FORD-FULKERSON.

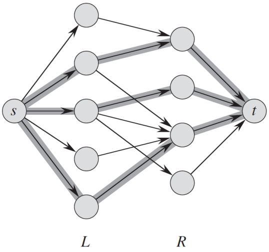


(c)

Is that 3?

TC 26.3-3

Let $G = (V, E)$ be a bipartite graph with vertex partition $V = L \cup R$, and let G' be its corresponding flow network. Give a good upper bound on the length of any augmenting path found in G' during the execution of FORD-FULKERSON.

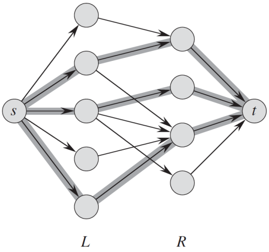


(c)

Is that 3? ✗

TC 26.3-3

Let $G = (V, E)$ be a bipartite graph with vertex partition $V = L \cup R$, and let G' be its corresponding flow network. Give a good upper bound on the length of any augmenting path found in G' during the execution of FORD-FULKERSON.

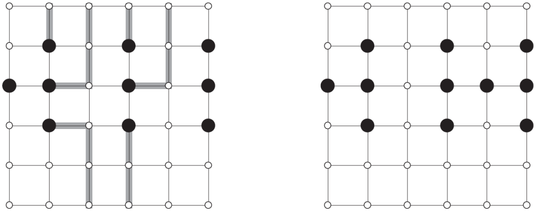


(c)

Is that 3? ✗

It is $2\min\{|L|, |R|\} - 1 + 2 = 2\min\{|L|, |R|\} + 1$

TC Problem 26-1(Escape problem)



(a) (b)

An $n \times n$ ***grid*** is an undirected graph consisting of n rows and n columns of vertices, as shown in Figure 26.11. We denote the vertex in the ith row and the jth column by (i, j). All vertices in a grid have exactly four neighbors, except for the boundary vertices, which are the points (i, j) for which $i = 1, i = n, j = 1$, or $j = n$.

Given $m \leq n^2$ starting points $(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$ in the grid, the ***escape problem*** is to determine whether or not there are m vertex-disjoint paths from the starting points to any m different points on the boundary.

TC Problem 26-1(Escape problem)

- As each **vertex** can be used only once, each vertex has a capacity of **one**.
- **Split** each vertex v into a pair of vertices v_1 and v_2 and add an edge (v_1, v_2) with capacity one
- Given a vertex v, for each of its neighbor u, add an edge (v_2, u_1) with capacity one.
- Create a supersource s and add an edge (s, v_1) with capacity one for each vertex$v$ corresponding to a starting point
- Create a supersink t and add an edge (v_2, t) with capacity one for each vertex v corresponding to a boundary point
- Finally, check if the value of a maxflow is no less than m


Thank
You!