

Problem Set 4

Due Date September 28, 2021
Name **Michael Ghattas**
Student ID **109200649**
Collaborators **Me, Myself, & I**

Contents

1 Instructions	1
2 Honor Code (Make Sure to Virtually Sign)	2
3 Standard 9- Network Flows: Terminology	3
4 Standard 10- Network Flows: Ford-Fulkerson	5
4.1 Problem 3(a)	5
4.2 Problem 3(b)	6
4.3 Problem 3(c)	7
4.4 Problem 3(d)	8
5 Standard 11- Network Flows: Reductions and Applications	9
5.1 Problem 4	9
5.1.1 Problem 5(a)	9
5.1.2 Problem 5(b)	11
5.2 Problem 6	12
5.2.1 Problem 6(a)	12
5.2.2 Problem 6(b)	14

1 Instructions

- The solutions **must be typed**, using proper mathematical notation. We cannot accept hand-written solutions. Here's a short intro to \LaTeX .
- You should submit your work through the **class Canvas page** only. Please submit one PDF file, compiled using this \LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).

- You are welcome and encouraged to collaborate with your classmates, as well as consult outside resources. You must **cite your sources in this document**. **Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.
- You **must** virtually sign the Honor Code (see Section 2). Failure to do so will result in your assignment not being graded.

2 Honor Code (Make Sure to Virtually Sign)

Problem 1. • My submission is in my own words and reflects my understanding of the material.

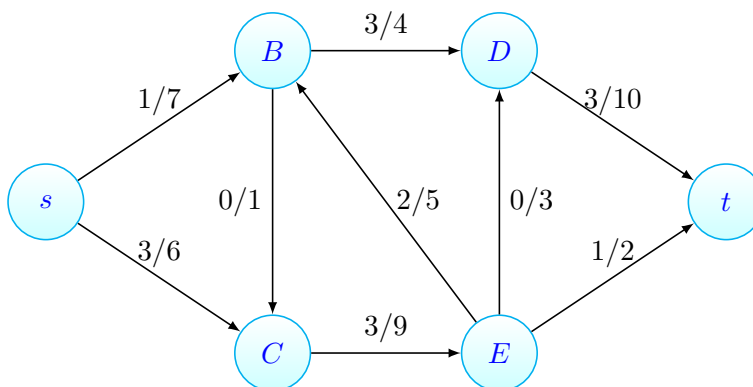
- Any collaborations and external sources have been clearly cited in this document.
- I have not posted to external services including, but not limited to Chegg, Reddit, StackExchange, etc.
- I have neither copied nor provided others solutions they can copy.

I agree to the above, Michael Ghattas.

□

3 Standard 9- Network Flows: Terminology

Problem 2. Consider the following flow network, with the following flow configuration f as indicated below.



Do the following.

- (a) Given the current flow configuration f , what is the maximum *additional* amount of flow that we can push across the edge (B, D) from $B \rightarrow D$? Justify using 1-2 sentences.

Answer:

The flow capacity, $c(B, D) = 4$, and the maximum *additional* amount of flow that we can push across the edge $E(B, D)$ from $B \rightarrow D = (\text{FlowCapacity} - \text{CurrentFlow}) = 4 - 3 = 1$. Thus, the total *additional* amount of flow we can push across the edge $E(B, D)$ from $B \rightarrow D = 1 - \text{Unit}$. \square

- (b) Given the current flow configuration f , what is the maximum amount of flow that B can push backwards to E ? Do **not** consider whether E can reroute that flow elsewhere; just whether B can push flow backwards. Justify using 1-2 sentences.

Answer:

The back-flow capacity of $E(B, E) = (\text{CurrentFlow})$ of $E(B, E) = 2$. Thus we can push $2 - \text{Units}$ back through $E(B, E)$. The maximum amount of flow that we can push back across the edge $E(B, E)$ from $B \rightarrow E = 2 - \text{Units}$. \square

- (c) Given the current flow configuration f , what is the maximum amount of flow that D can push backwards to E ? Do **not** consider whether D can reroute that flow elsewhere; just whether E can push flow backwards. Justify using 1-2 sentences.

Answer:

The back-flow capacity of $E(D, E) = (\text{CurrentFlow})$ of $E(D, E) = 0$. Thus we can push $0 - \text{Units}$ back through $E(D, E)$. The maximum amount of flow that we can push back across the edge $E(D, E)$ from $D \rightarrow E = 0 - \text{Units}$. \square

- (d) How much *additional* flow can be pushed along the flow-augmenting path $s \rightarrow B \rightarrow E \rightarrow t$? Do not include the current flow along these edges. Justify using 1-2 sentences.

Answer:

- The flow capacity, $c(s, B) = 7$, and the maximum *additional* amount of flow that we can push across the edge $E(s, B)$ from $s \rightarrow B = (FlowCapacity - CurrentFlow) = 7 - 1 = 6$. Thus, the *additional* amount of flow we can push across the edge $E(s, B)$ from $s \rightarrow B = 6 - Units$.
- The back-flow capacity of $E(B, E) = (CurrentFlow)$ of $E(B, E) = 2$. Thus, we can push $2 - Units$ back through $E(B, E)$. The maximum *additional* amount of flow that we can push back across the edge $E(B, E)$ from $B \rightarrow E = 2 - Units$.
- The flow capacity, $c(E, t) = 2$, and the maximum *additional* amount of flow that we can push across the edge $E(E, t)$ from $E \rightarrow t = (FlowCapacity - CurrentFlow) = 2 - 1 = 1$. Thus, the *additional* amount of flow we can push across the edge $E(E, t)$ from $E \rightarrow t = 1 - Unit$.

Since the *additional* amount of flow we can push from $E \rightarrow t = 1 - Unit$, the total *additional* amount of flow we can push across the *flow - augmenting* path $s \rightarrow B \rightarrow E \rightarrow t = 1 - Unit$. \square

- (e) Find a second flow-augmenting path and indicate the maximum amount of additional flow that can be pushed along the path. Assume that the flow-augmenting path from part (d) has **not** been applied. Justify using 1-2 sentences.

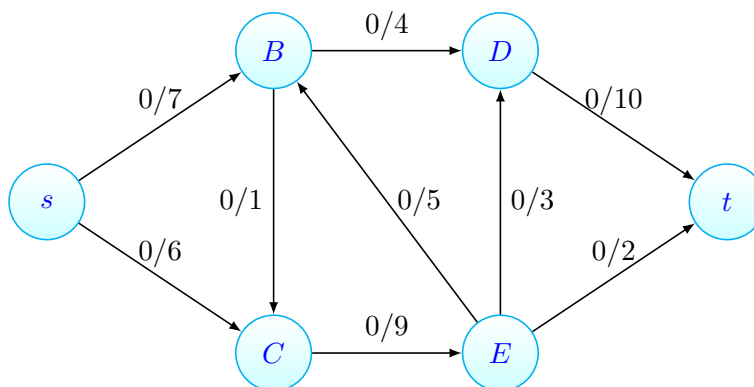
Answer:

- The flow capacity, $c(s, C) = 6$, and the maximum *additional* amount of flow that we can push across the edge $E(s, C)$ from $s \rightarrow C = (FlowCapacity - CurrentFlow) = 6 - 3 = 3$. Thus, the *additional* amount of flow we can push across the edge $E(s, C)$ from $s \rightarrow C = 3 - Units$.
- The flow capacity, $c(C, E) = 9$, and the maximum *additional* amount of flow that we can push across the edge $E(C, E)$ from $C \rightarrow E = (FlowCapacity - CurrentFlow) = 9 - 3 = 6$. Thus, the *additional* amount of flow we can push across the edge $E(C, E)$ from $C \rightarrow E = 6 - Units$.
- The flow capacity, $c(E, D) = 3$, and the maximum *additional* amount of flow that we can push across the edge $E(E, D)$ from $E \rightarrow D = (FlowCapacity - CurrentFlow) = 3 - 0 = 3$. Thus, the *additional* amount of flow we can push across the edge $E(E, D)$ from $E \rightarrow D = 3 - Units$.
- The flow capacity, $c(D, t) = 10$, and the maximum *additional* amount of flow that we can push across the edge $E(D, t)$ from $D \rightarrow t = (FlowCapacity - CurrentFlow) = 10 - 3 = 7$. Thus, the *additional* amount of flow we can push across the edge $E(D, t)$ from $D \rightarrow t = 7 - Units$.

Since the smallest *additional* amount of flow we can push between any 2 vertices through any edge $E(G)$ across $s \rightarrow C \rightarrow E \rightarrow D \rightarrow t$ is (3), the total *additional* amount of flow we can push across the *flow - augmenting* path $s \rightarrow C \rightarrow E \rightarrow D \rightarrow t = 3 - Units$. Thus our new *flow - augmenting* path $(f') = s \rightarrow C \rightarrow E \rightarrow D \rightarrow t$. \square

4 Standard 10- Network Flows: Ford-Fulkerson

Problem 3. Consider the following flow network, with no initial flow along the graph.



Do the following.

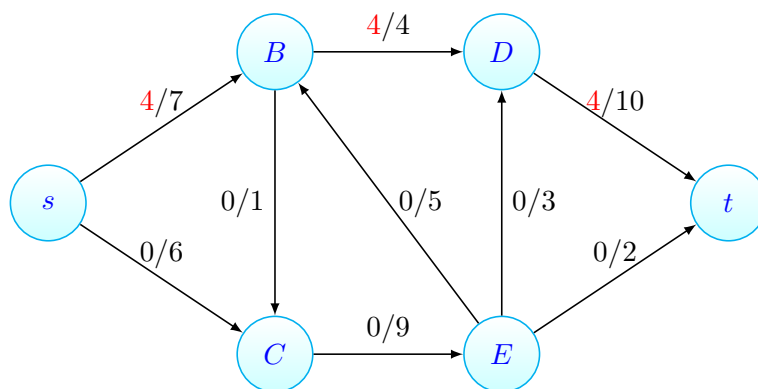
4.1 Problem 3(a)

- (a) Consider the flow-augmenting path $s \rightarrow B \rightarrow D \rightarrow t$. Push as much flow through the flow-augmenting path and draw the updated flow network below.

Answer:

Using *flow – augmenting path* $(f_1) := s \rightarrow B \rightarrow D \rightarrow t$.

Note the minimum available flow capacity across $(f_1) := s \rightarrow B \rightarrow D \rightarrow t$ is the edge $[B, D] = 4$, thus we were able to push a maximum of *additional 4 – Units*.



□

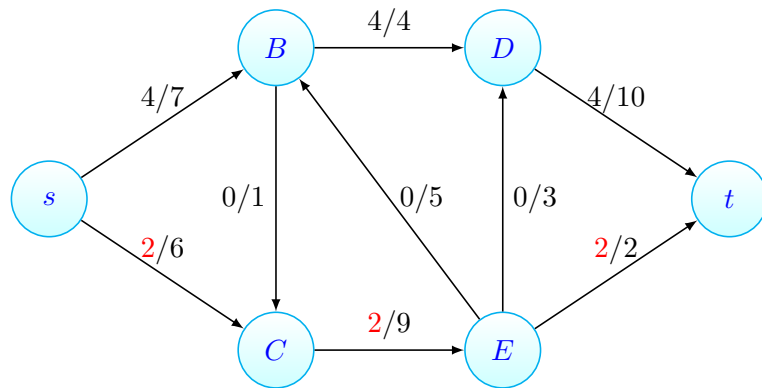
4.2 Problem 3(b)

- (b) Find a flow-augmenting path using the updated flow configuration from part (a). Then do the following:
 (i) clearly identify both the flow-augmenting path and the maximum amount of flow that can be pushed through said path; and then (ii) push as much flow through the flow-augmenting path and draw the updated flow network below.

Answer:

Using *flow – augmenting path* $(f_2) := s \rightarrow C \rightarrow E \rightarrow t$.

Note the minimum available flow capacity across $(f_2) := s \rightarrow C \rightarrow E \rightarrow t$ is the edge $[\{E, t\} = 2]$, thus we were able to push a maximum of additional **2 – Units.**



□

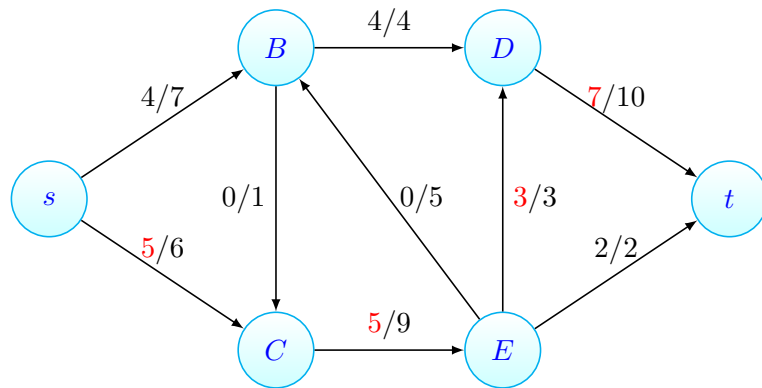
4.3 Problem 3(c)

- (c) Find a flow-augmenting path using the updated flow configuration from part (b). Then do the following:
 (i) clearly identify both the flow-augmenting path and the maximum amount of flow that can be pushed through said path; and then (ii) push as much flow through the flow-augmenting path and draw the updated flow network below.

Answer:

Using *flow – augmenting path* $(f_3) := s \rightarrow C \rightarrow E \rightarrow D \rightarrow t$.

Note the minimum available flow capacity across $(f_3) := s \rightarrow C \rightarrow E \rightarrow D \rightarrow t$ is the edge $[\{E, D\} = 3]$, thus we were able to push a maximum of *additional 3 – Units*.



□

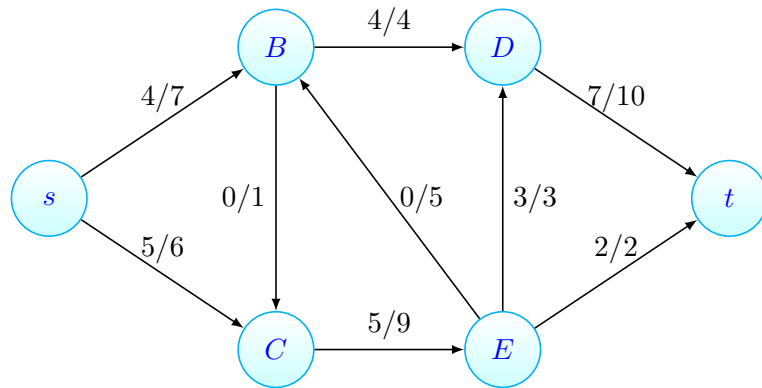
4.4 Problem 3(d)

- (d) Using the flow configuration from part ((c)), finish executing the Ford–Fulkerson algorithm. Include the following here: (i) your flow network, reflecting the maximum-valued flow configuration you found, and (ii) the corresponding minimum capacity cut. There may be multiple minimum capacity cuts, but you should identify the one corresponding to your maximum-valued flow configuration. Then (iii) finally, compare the value of your flow to the capacity of the cut.

Note: You do **not** need to include the remaining steps of the Ford–Fulkerson algorithm. We will not check these steps when grading.

Answer:

Looking back at our *flow – augmenting* paths (f_1) , (f_2) , & $(f_3) := (f^*)$, we were able to push a total maximum flow of $(4 + 2 + 3) = 9 - \text{Units}$.



Minimum-Capacity cut of $(f^*) = c(X, Y)$:

- $X = \{s, B, C, E\}$
 - $[s \rightarrow B]$ We can still push 3 – Units of additional positive flow
 - $[s \rightarrow C]$ We can still push 1 – Unit of additional positive flow
 - $[C \rightarrow E]$ We can still push 4 – Units of additional positive flow
- $Y = \{D, t\}$

$$\begin{aligned}
 c(X, Y) &= c(B, D) + c(E, D) + c(E, t) \\
 &= 4 + 3 + 2 \\
 &= 9 - \text{Units}
 \end{aligned}$$

Therefore, we can conclude that for $(f^*) \rightarrow \text{MaximumFlow} = \text{MinimumCut}$. □

5 Standard 11- Network Flows: Reductions and Applications

5.1 Problem 4

Problem 4. In this problem, we reduce the Maximum Flow problem where multiple sources and sinks are allowed to the One-Source, One-Sink Maximum Flow problem. The reduction is as follows. Let $\mathcal{N}(G, c, S, T)$ be our flow network with multiple sources and multiple sinks. We construct a new flow network $\mathcal{N}'(H, c', S', T')$, as follows.

- $S' = \{s'\}$ is the set containing our one source.
- $T' = \{t'\}$ is the set containing our one sink.
- We now construct H by starting with G (including precisely the vertices and edges of G) and adding s' and t' . For each source $s \in S$ of G , we add a directed edge (s', s) (that is, $s' \rightarrow s$) in H . For each sink $t \in T$ of G , we add a directed edge (t, t') (that is, $t \rightarrow t'$) in H .
- We construct the capacity function c' of H as follows.
 - If (u, v) corresponds to an edge of G , then $c'(u, v) = c(u, v)$.
 - If $s \in S$ is a source of G , then $c'(s', s)$ is the amount of flow that we can push from s in G . That is:

$$c(s', s) = \sum_{(s, v) \in E(G)} c(s, v).$$

- If $t \in T$ is a sink of G , then $c'(t, t')$ is the maximum amount of flow that t can receive along its incoming edges in G . That is:

$$c'(t, t') = \sum_{(v, t) \in E(G)} c(v, t).$$

Do the following. [**Hint:** Before attempting either part (a) or part (b), we highly recommend doing the following scratch work first. Construct your own flow network with multiple sources and multiple sinks. Then go through the above construction carefully to obtain a new flow network with one source and one sink. Trying to construct your own examples is extremely beneficial when working to understand a new construction.]

5.1.1 Problem 5(a)

- (a) Show that, for every feasible flow f on \mathcal{N} , there exists a (corresponding) feasible flow f' on \mathcal{N}' such that $\text{val}(f) = \text{val}(f')$.

Answer:

- **Considering the information provided above, we can note that the difference between $\mathcal{N}(G, c, S, T)$ and $\mathcal{N}'(H, c', S', T')$ is the addition of (s') and (t') , along with the associated edges connecting (s') to all $(s \in S)$, and all $(t \in T)$ to (t') .**
- **If $\forall (f') \in \mathcal{N}'(H, c', S', T')$, $\forall s \in S$, (s') is connected to (s) , and $\forall t \in T$, (t') is connected to (t) , then (s') is connected to (t') through all the possible combinations of flow – augmenting paths $\in \mathcal{N}(G, c, S, T)$.**
- **Then $\forall (f) \in \mathcal{N}(G, c, S, T)$, there exists $(f') \in \mathcal{N}'(H, c', S', T')$ that is the same with the exceptions of (s') and (t') , along with the associated edges connecting (s') to all (s) , and all (t) to (t') .**

- Furthermore, for any $(f') \in \mathcal{N}'(H, c', S', T')$, the single edge in any (f') connecting (s') to each (s) holds the maximum flow capacity (s) can distribute to its adjacent vertices, and is equal in value to $c(s', s)$. Similarly, the single edge in any (f') connecting each (t) to (t') holds the maximum capacity that can be pushed from t -adjacent vertices to (t) , and is equal in value to $c(t, t')$.
- We can conclude, $\forall (f) \in \mathcal{N}(G, c, S, T)$, there exists $(f') \in \mathcal{N}'(H, c', S', T')$ such that $val(f) = val(f')$ as desired.

□

5.1.2 Problem 5(b)

- (b) For every feasible flow g' on \mathcal{N}' , show how to recover a feasible flow g on \mathcal{N} such that $\text{val}(g) = \text{val}(g')$.

Answer:

Considering the provided information and part (a) above, we can note that the difference between $\mathcal{N}(G, c, S, T)$ and $\mathcal{N}'(H, c', S', T')$ is the addition of (s') and (t') , along with the associated edges connecting (s') to all $(s \in S)$, and all $(t \in T)$ to (t') .

Thus to reach our desired solution (g) by deconstructing (g') , we simply need to remove all the edges connecting (s') to (s) with exception to the edge connecting (s') to the *source* vertex of (g) , and similarly removing all the edges connecting $(t \rightarrow t')$ with exception to the edge connecting the *sink* vertex of (g) to (t') .

Finally, we will need to delete (s') and (t') along with their associated edges connecting (g') to the *source* and *sink* vertices of (g) . Thus, the remaining *flow augmenting* path (g) is recovered with the $\text{val}(g)$ preserved in the sum of flow in the edges connected to (t) . Accordingly, this will result in (g) , where $\text{val}(g) = \text{val}(g')$ as desired. \square

5.2 Problem 6

Problem 5. Suppose a restaurant has 3 customers, each of whom can order at most one entree. The restaurant is running low on inventory. It has the following in stock:

- Two burgers,
- Three servings of crab cakes, and
- One salmon filet.

Each customer submits their order, indicating only whether they will eat a given meal. **Of the meals a given customer is willing to eat, that customer does NOT indicate whether they prefer one meal to another.** The restaurant wishes to assign entrees in such a way that respects whether the customers will eat their entrees and maximizes the number of entrees sold.

Do the following.

5.2.1 Problem 6(a)

- (a) Carefully describe how to construct a flow network corresponding to the customers' order preferences. That is, we take as input the customers' order preferences, and you need to describe how to construct the corresponding flow network.

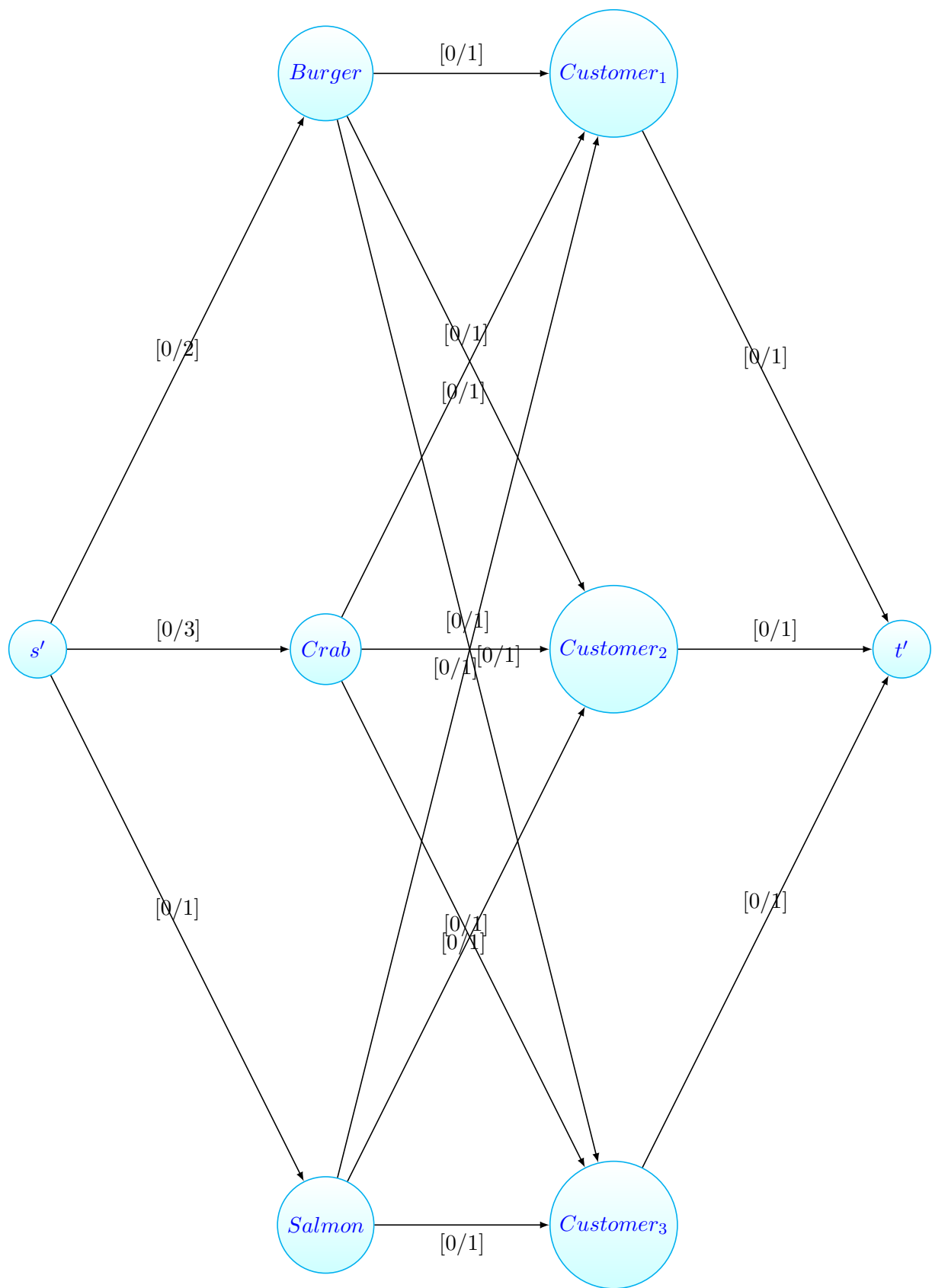
Note that you just have to give a construction. You are **not** being asked to prove anything about your construction.

Answer:

Our graph will be constructed as follows:

- The graph will hold 3 *source* vertices on the left, representing each of the available meals.
- Additionally, on the right, the graph has 3 *sink* vertices, representing each of the 3 customers.
- Connecting the *source* vertices to the *sink* vertices are identical edges with capacity = 1.
- Each edge represents the preference of the customer regarding the meal, with ($No = 0$) and ($Yes = 1$).

Please see graph below for reference:



□

5.2.2 Problem 6(b)

(b) Suppose that the three customers provide their orders:

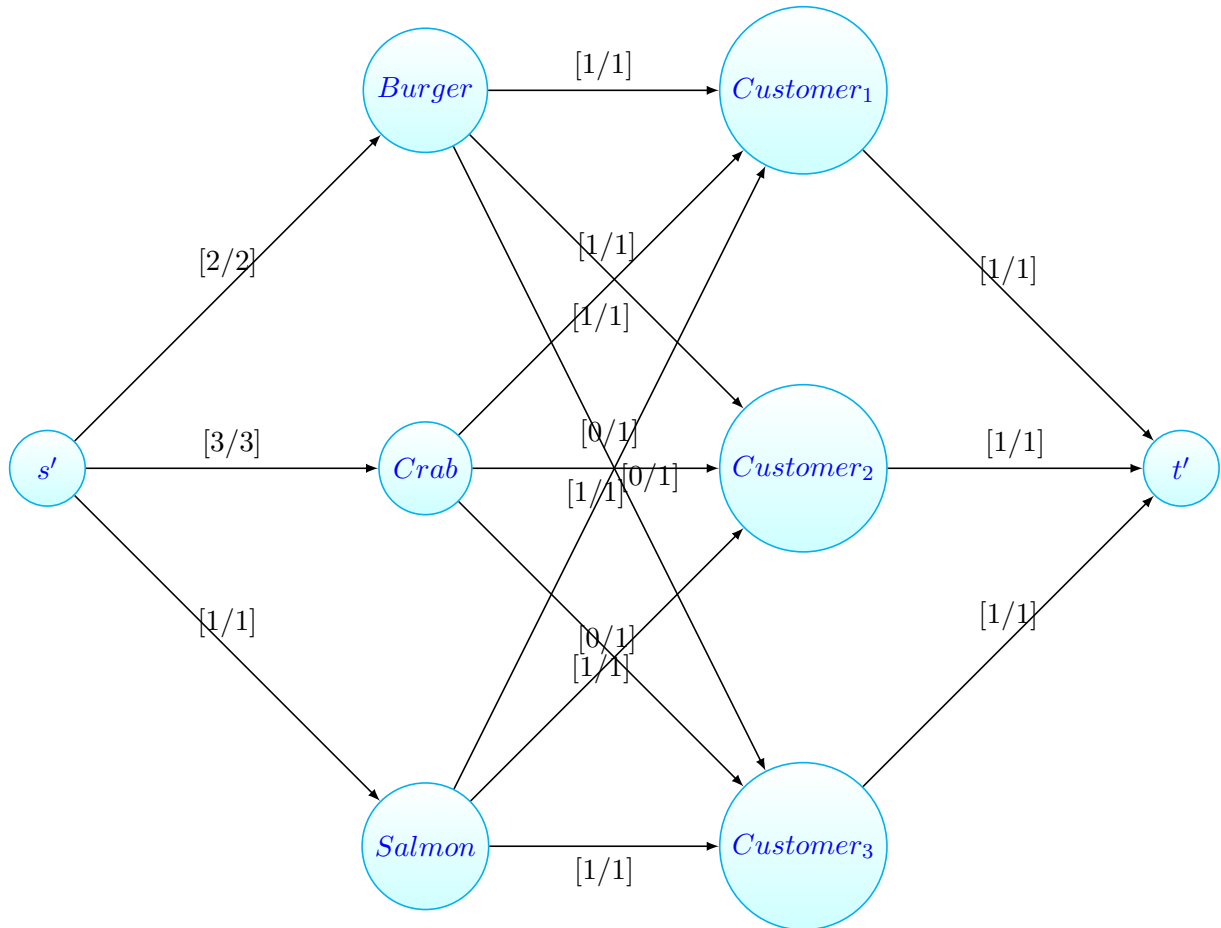
- Customer 1: Crab cakes, Burger. [Note: Customer 1 will not eat Salmon]
- Customer 2: Salmon, Burger, Crab cakes.
- Customer 3: Salmon, Crab cakes, Burger.

Using your construction from part (a), find a maximum-valued flow on your flow network and identify the corresponding allocation of entrees to customers. You may hand-draw your flow network, but the explanation must be typed.

Answer:

A few possible variations of *flow – augmenting* paths exist, and one of these variations is:

- $Customer_1$: [0 - Burger : 1 - Crab cake : 0 - Salmon] = 1 - Meal.
- $Customer_2$: [1 - Burger : 0 - Crab cake : 0 - Salmon] = 1 - Meal.
- $Customer_3$: [0 - Burger : 0 - Crab cake : 1 - Salmon] = 1 - Meal.



In conclusion; for any *flow – augmenting* path (f) $[\{D, t'\} = 1]$, $[\{E, t'\} = 1]$, $[\{F, t'\} = 1]$, **thus regardless of which possible variation we choose, the maximum-valued flow for $|f| = 1 + 1 + 1 = 3$. (i.e. One meal per customer)** \square