CSCI 3104 Fall 2021
Instructors: Profs. Grochow and Waggoner

Problem Set 4

Due Date .... September 28, 2021
Name .... **John Blackburn**
Student ID **Jobl2177**
Collaborators .... **none**

Contents

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)

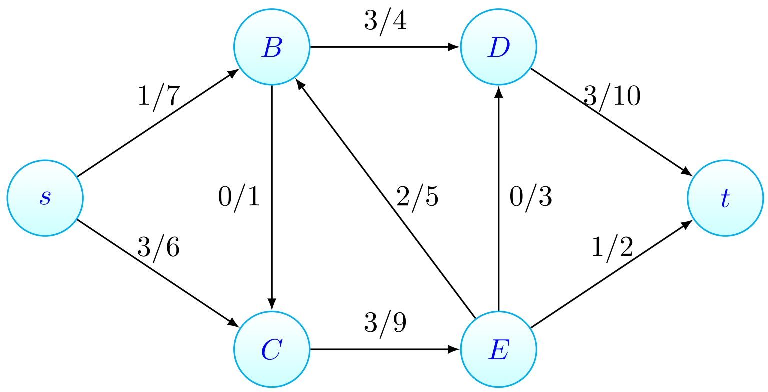
Problem 1.

- 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.

Agreed (John Blackburn). □

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 $B \to D$? Justify using 1-2 sentences.

Answer. You can add a max additional flow of 1 across the edge from $B \to D$. The edge has a capacity of 4 and already has 3 flowing through it. Therefore, only 1 can be added before you reach the edge capacity. □

(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 max that B can push backwards to E given the current flow configuration is 2. When pushing flow backwards through an edge, you cannot go into a negative flow, meaning the flow can never go below 0. Therefore, since there is already 2 going in the positive direction the max amount of backwards flow that can be pushed through is 2. □

(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. $D \to E$ cannot push any flow backwards through the edge. The edge currently has no flow going in the forward direction, and since backwards flow cannot push the flow into negative numbers, the max flow from D to E is 0.

□

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

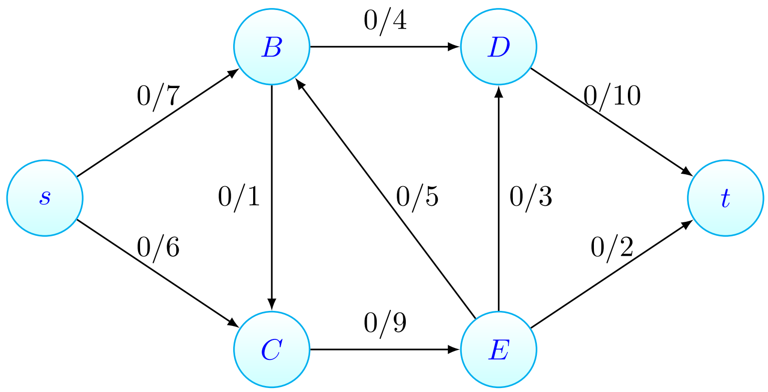
Answer. The additional flow that can be pushed along the path is 1. Along the path the minimum remaining flow available is the max amount of flow you can push along the whole path, and along edge $E \to T$ only 1 can be pushed through making 1 the max flow along said path. □

(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. A second flow augmenting path I found is $s \to C \to E \to D \to t$. And the max flow that can be pushed along this path is 3. $E \to D$ is the constraining edge with a max capacity of 3. Therefore, along the path the max amount of flow that can be pushed is 3. □

4 Standard 10- Network Flows: Ford-Fulkerson

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

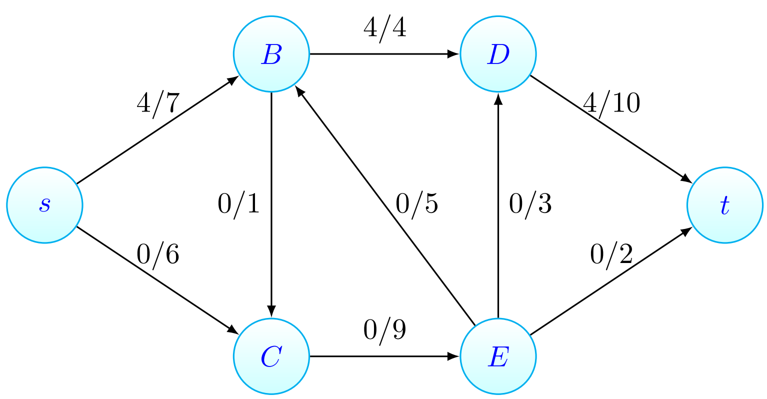


Do the following.

4.1 Problem 3(a)

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

Answer. You can push a total of 4 through the said path.

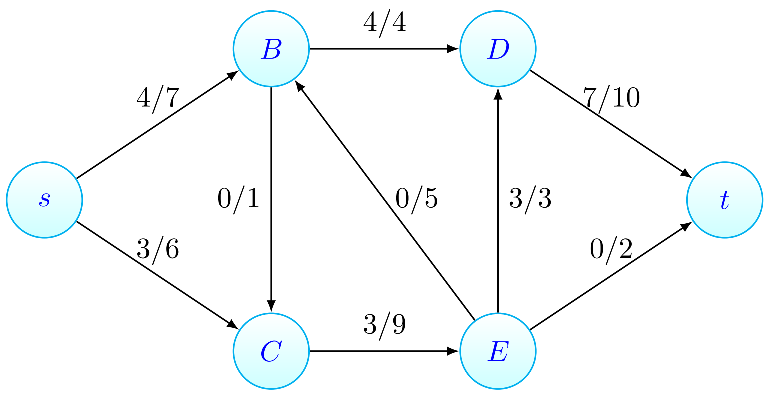


□

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. A new flow augmenting path that can be implemented is $s \to C \to E \to D \to t$. The max flow that can be pushed through this path is 3. The new updated flow network is below.

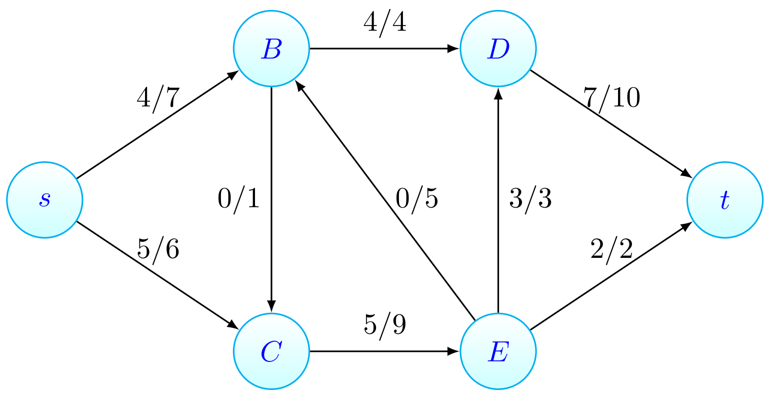


□

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. A new flow that can be added is the path $s \to C \to E \to t$. And the max flow that can be pushed through said path is 2. The updated flow network is below.



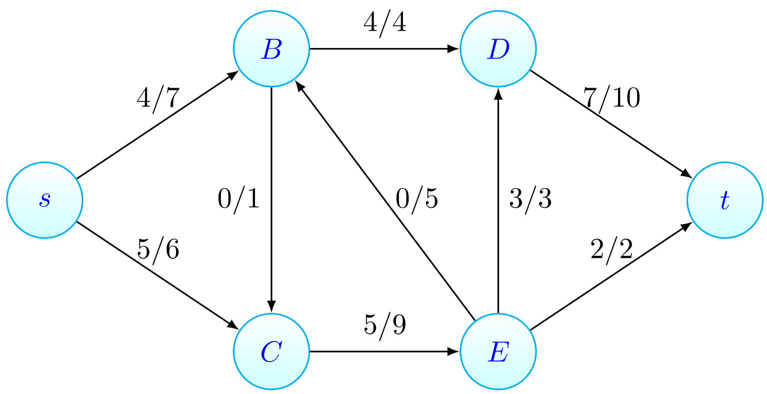
□

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. My complete flow network is below, with a max flow of 9. The minimum capacity cut, cuts along edges $B \to D$, $E \to D$, and $E \to t$. Set S contains nodes s, B, C, and E. Set T has nodes t, and D. The minimum cut I found is equal to 9. My max flow and min cut are both equal to 9, telling me that I executed ford-fulkerson correctly, because we have the theorem that states any max flow should be equivalent to the min cut.



□

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, $s' \to s$) in H. For each sink $t \in T$ of G, we add a directed edge (t, t') (that is, $t \to t'$) 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 $\text{val}(f) = \text{val}(f')$.

Proof. For every flow f on $\mathcal{N}$, we can know there exists a (corresponding) feasible flow f' on $\mathcal{N}'$ such that $\text{val}(f) = \text{val}(f')$ because we know our capacity function c' of H will not reduce the capacity anywhere within H. When constructing H from G all we are doing is adding edges from the sinks to a new single sink, keeping all the interior edges from the sources the same, and adding a single source that connects to each of the sources from G. The capacity of all the edges of G will remain the same as per the above definition, If (u, v) corresponds to an edge of G, then $c'(u, v) = c(u, v)$. So any flow f will not be changed in these edges. As for the source of the flow, it essentially just has to take one more step. Source s where flow f, starts within G will no longer be the starting point of the flow within flow f'. Our new flow starts at source s', and we know the capacity of any edge going from s' to s is

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

, as per defined above. Therefore we know that any flow f' won't be constrained in any way from the new source to the original source in G. That just leaves whether or not the edge from the old sink to the new sink will constrain flow f'. Our capacity function given for such edges is

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

. This tells us that any amount of flow that was able to be pushed to the old sink can also be pushed into our new single sink.

So, overall we can see that for any new flow, our new flow network has the same capacity as our old flow network just with a few new edges and nodes. Therefore, any flow f that can be constructed in $\mathcal{N}$ can also be reconstructed in $\mathcal{N}'$ as f' such that $\mathrm{val}(f) = \mathrm{val}(f')$.

$\square$

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 $\text{val}(g) = \text{val}(g')$.

Proof. For every feasible flow g' on $\mathcal{N}'$, we can recover a feasible flow g on $\mathcal{N}$ such that $\text{val}(g) = \text{val}(g')$, by moving up the source and sink of the flow. $\mathcal{N}'$ has a single source s' and a single sink f', while $\mathcal{N}$ has multiple sinks and sources. In order to recover a flow g on $\mathcal{N}$ such that $\text{val}(g) = \text{val}(g')$, we first need to move the starting point up to a new source node without losing any flow capacity. Our source node in $\mathcal{N}'$ has the capacity function:

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

So we know that any single flow flowing through node s to any interior node v will constrain the edge from s' to s because that edge from s to interior node v can at most be the capacity of the edge from s' to s. Therefore if we move up the starting point to s we know that we can push the same amount of flow through.

We also know that the capacity of all interior edges is exactly the same in both flow networks as given per, (u, v) corresponds to an edge of G, then $c'(u, v) = c(u, v)$. So these edges should cause no issue reconstructing the flow since they will be exactly the same.

That just leaves the sink nodes. In order to get back to $\mathcal{N}$ we have to remove our single sink node and all edges leading to it. The capacity of said edges is given by:

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

The edges leading from the interior nodes to the old sinks is at most the capacity of the edges leading to the single sink, so we will be able to push the same amount of flow through and just stop it one step earlier without issue.

Overall, if we remove the single sink and source, our capacity of flow will remain the same. Therefore for every feasible flow g' on $\mathcal{N}'$, we can recover a feasible flow g on $\mathcal{N}$ such that $\text{val}(g) = \text{val}(g')$.

$\square$

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 customers 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. I will describe a network that will maximize the amount of entrees sold while respecting customer preferences. It will have a single source node and single sink node.

My source node represents the restaurant(R), it has 3 edges leading out of it each connecting to a node representing the type of food they serve and the capacity of each edge corresponds to the amount of food they have left for that specific entree. The first edge connects restaurant to Burgers(B), the capacity of the edge is 2 because there are two burgers left. The second edge connects restaurant to Crab cakes(CC), with an edge capacity of 3. The third edge is connecting restaurant to salmon(S), with a capacity of 1.

From here we build connections from the food type nodes to customer nodes. Here is where the input from the customer comes in, depending on what the customer lists as their preference we build connections between them and the food types they write that they will eat. So we have 3 more new nodes representing each customer(C1, C2, C3). If the customer lists salmon as a preference there is an edge connected between the salmon node and that customer node. Same thing for each food type and customer. Each edge between the food nodes and customer would have a capacity of 1, because each customer can only order a single entree.

Then from each of the customer nodes we make connections to a final sink node representing the total orders eaten by the customers(O). So there are 3 connections to the sink node each with a capacity of 1.

This completes my flow network.

□

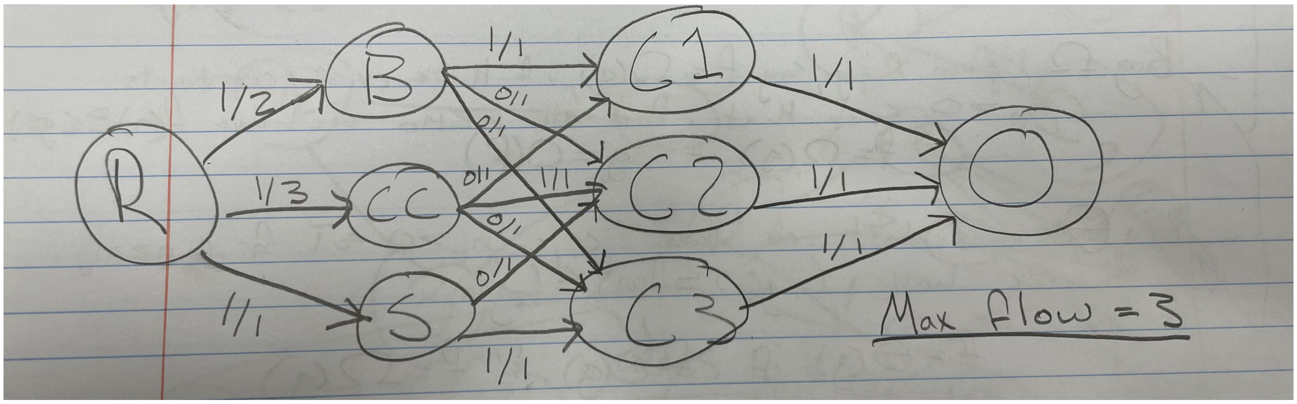
5.2.2 Problem 6(b)

(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. The max flow from my network is 3, meaning that each customer was able to get a single entree. Customer 1 got a Burger, Customer 2 got a crab cake, and Customer 3 got a salmon.



☐