CMSC 451: Lecture 14
Network Flow: Extensions and Applications

Extensions of Network Flow: Network flow is an important problem because it is useful in a wide variety of applications. We will discuss two useful extensions to the network flow problem. We will show that these problems can be reduced to network flow, and thus a single algorithm can be used to solve both of them. Many computational problems that would seem to have little to do with flow of fluids through networks can be expressed as one of these two extended versions.

Circulation with Demands: There are many problems that are similar to network flow in which, rather than transporting flow from a single source to a single sink, we have a collection of *supply nodes* that want to ship flow (or products or goods) and a collection of *demand nodes* that want to receive flow. Each supply node is associated with the amount of product it wishes to ship and each demand node is associated with the amount that it wishes to receive. The question that arises is whether there is some way to get the products from the supply nodes to the demand nodes, subject to the capacity constraints. This is a *decision problem* (or *feasibility problem*), meaning that it has a yes-no answer, as opposed to maximum flow, which is an *optimization problem*.

We can model both supply and demand nodes elegantly by associating a single numeric value with each node, called its *demand*. If $v \in V$ is a demand node, let $d(v)$ the amount of this demand. If v is a supply node, we model this by assigning it a negative demand, so that $-d(v)$ is its available supply. Intuitively, supplying x units of product is equivalent to demanding receipt of $-x$ units.[1] If v is neither a supply or demand node, we let $d(v) = 0$.

Suppose that we are given a directed graph $G = (V, E)$ in which each edge (u, v) is associated with a positive capacity $c(u, v)$ and each vertex v is associated with a supply/demand value $d(v)$. Let S denote the set of *supply nodes* ($d(v) < 0$), and let T denote the set of *demand nodes* ($d(v) > 0$). Note that vertices of S may have incoming edges and vertices of T may have outgoing edges. (For example, in Fig. 1(a), we show a network in which each node is each labeled with its demand and each edge with its capacity.) Recall that, given a flow f and a node v, $f^{\text{in}}(v)$ is the sum of flows along incoming edges to v and $f^{\text{out}}(v)$ is the sum of flows along outgoing edges from v.

Definition: Given a directed graph $G = (V, E)$ in which each edge $(u, v) \in E$ is associated with a positive capacity $c(u, v)$ and each vertex $v \in V$ is associated with a supply/demand value $d(v)$, a *circulation* in G is a function f that assigns a nonnegative real number to each edge that satisfies the following conditions:

Capacity constraints: For each $(u, v) \in E$, $0 \leq f(u, v) \leq c(u, v)$.

Supply/Demand constraints: For each $v \in V$, $f^{\text{in}}(v) - f^{\text{out}}(v) = d(v)$.

For example, in Fig. 1(b) we show a valid circulation for the network of part (a). Observe that demand constraints correspond to the flow-balance in the original max flow problem,

[1] I would not advise applying this in real life. I doubt that the IRS would appreciate it if you paid your \$100 tax bill by demanding that they send you −\$100 dollars.

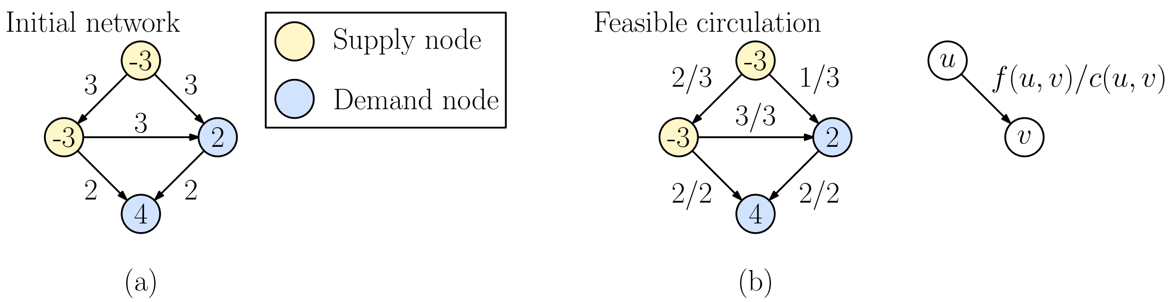

Fig. 1: (a) A circulation network and (b) a feasible circulation.

since if a vertex is not in S or T, then $d(v) = 0$ and we have $f^{\text{in}}(v) = f^{\text{out}}(v)$. Also it is easy to see that the total demand must equal the total supply, otherwise we have no chance of finding a feasible circulation. That is, we require that

$$\sum_{v \in V} d(v) \ = \ 0 \qquad \text{or equivalently} \qquad -\sum_{v \in S} d(v) \ = \ \sum_{v \in T} d(v).$$

Define $D = \sum_{v \in T} d(v)$ denote the *total demand*. (Note that this is equal to the negation of the total supply, $\sum_{v \in S} d(v)$.)

Observe that the max-flow problem is closely related. Suppose that we are given a (standard) s-t network, and we want to know whether it has a flow of value D. This is equivalent to a circulation network where we set $d(s) = -D$, $d(t) = D$ and $d(u) = 0$, for all other vertices.

Reducing Circulation to Max-Flow: Rather than devise an algorithm for the circulation problem, we will show that we can reduce any instance G of the circulation problem into an equivalent network flow problem. The input to our reduction is a network $G = (V, E)$. For each vertex v, let $d(v)$ denote the demand value and for each edge (u, v), let $c(u, v)$ denote its capacity. Intuitively, the idea is push flow into all the supply vertices and extract it from all the demand vertices (see Fig. 2).

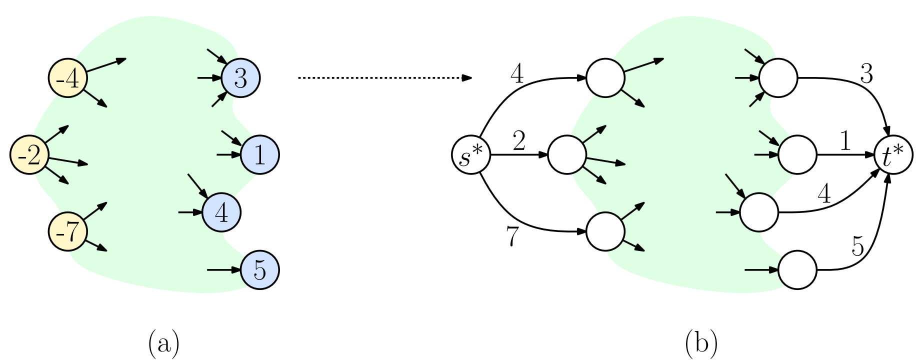

Fig. 2: Reducing the circulation problem (a) to a standard s-t flow network (b).

Here is the formal reduction. Oserve first that we may assume that sum of supplies equals total demand (since if not, we can simply answer "no" immediately.) Otherwise:

- Create a new network $G' = (V', E')$ that has all the same vertices and edges as G (that is, $V' \leftarrow V$ and $E' \leftarrow E$)
- Add to V' a *super-source* s^* and a *super-sink* t^*
- For each supply node $v \in S$, we add a new edge (s^*, v) of capacity $-d(v)$
- For each demand node $u \in T$, we add a new adge (u, t^*) of capacity $d(v)$

Applying this to the circulation network shown in Fig. 1, yields the s-t network is shown in Fig. 3(b). We then invoke any max-flow algorithm on G'. Recalling that D denotes the total demand, we check whether the value of the maximum flow equals D. If so, we answer "yes," G has a feasible circulation, and otherwise we answer "no," G does not have a feasible circulation. (For example, in Fig. 3(c), there exists a flow of value $D = 6$, implying that the original network has a circulation.)

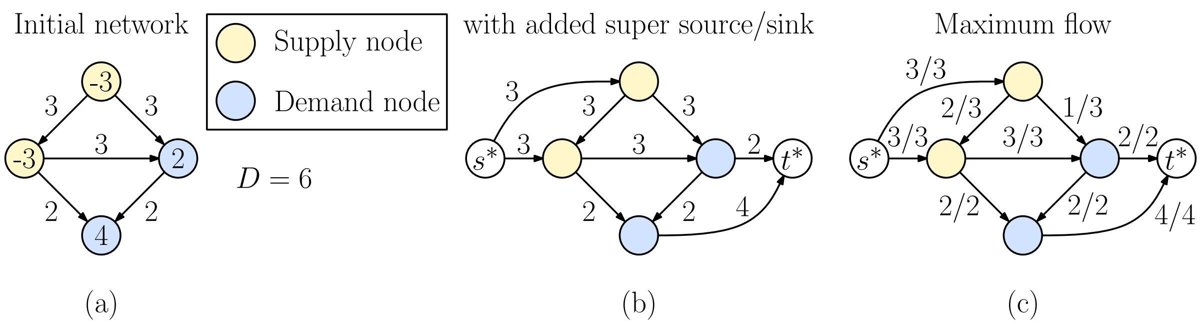

Fig. 3: Reducing the circulation problem to network flow.

We prove correctness below, but intuitively, the newly created edges from s^* will be responsible for providing the necessary supply for the supply vertices of S and newly created edges into t^* are responsible for draining off the excess demand from the vertices of T. Suppose that we now compute the maximum flow in G' (by whichever maximum flow algorithm you like).

Lemma: There is a feasible circulation in G if and only if G' has an s^*-t^* flow of value D.

Proof: ($\Rightarrow$) Suppose that there is a feasible circulation f in G. The value of this circulation (the net flow coming out of all supply nodes) is clearly D. We can create a flow f' of value D in G', by saturating all the edges coming out of s^* and all the edges coming into t^*. We claim that this is a valid flow for G'. Clearly it satisfies all the capacity constraints. To see that it satisfies the flow balance constraints observe that for each vertex $v \in V$, we have one of three cases:

- ($v \in S$) The flow into v from s^* matches the supply coming out of v from the circulation.
- ($v \in T$) The flow out of v to t^* matches the demand coming into v from the circulation.
- ($v \in V \setminus (S \cup T)$) We have $d(v) = 0$, which means that it satisfies flow conservation by the supply/demand constraints.

(⇐) Conversely, suppose that we have a flow f' of value D in G'. It must be that each edge leaving s^* and each edge entering t^* is saturated. Therefore, by the flow conservation of f', all the supply nodes and all the demand nodes have achieve their desired supply/demand constraints. All the other nodes satisfy their supply/demand constraints because by the flow conservation of f' the incoming flow equals the outgoing flow. Therefore, the resulting flow is a circulation for G.

It is not hard to see to that the reduction can be performed in $O(n + m)$ time by a simply analysis of the network's structure. Thus, the overall running time is dominated by the time to compute the network flow (which is $O(nm)$ according to the current state of the art).

Circulations with Upper and Lower Capacity Bounds: Sometimes, in addition to having a certain maximum flow value, we would also like to impose minimum capacity constraints. That is, given a network $G = (V, E)$, for each edge $(u, v) \in E$ we would like to specify two constraints $\ell(u, v)$ and $c(u, v)$, where $0 \leq \ell(u, v) \leq c(u, v)$. A circulation function f must satisfy the same demand constraints as before, but must also satisfy both the upper and lower flow bounds:

(New) Capacity Constraints: For each $(u, v) \in E$, $\ell(u, v) \leq f(u, v) \leq c(u, v)$.

Demand Constraints: For vertex $v \in V$, $f^{\text{in}}(v) - f^{\text{out}}(v) = d(v)$.

Henceforth, we will use the term *upper flow bound* in place of *capacity* (since it doesn't make sense to talk about a lower bound as a capacity constraint). An example of such a network is shown in Fig. 4(a), and a valid circulation is shown in Fig. 4(b).

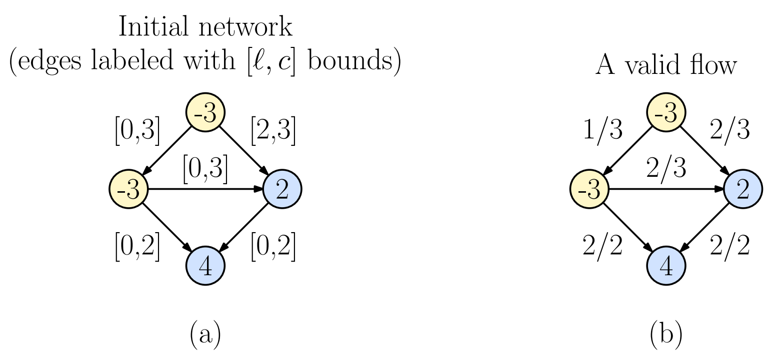

Fig. 4: (a) A network with both upper and lower flow bounds and (b) a valid circulation.

We will reduce this problem to a standard circulation problem (with just the usual upper capacity bounds). To help motivate our reduction, suppose (for conceptual purposes) that we generate an initial (probably invalid) "pseudo-circulation" f_0 that exactly satisfies all the lower flow bounds. In particular, we let $f_0(u, v) = \ell(u, v)$. This circulation may be (in fact will likely be) invalid because f_0 need not satisfy the demand constraints. (Recall these also guarantee flow conservation.) Here's the trick. We will now modify the supply/demand values to compensate for this shortcoming. Why will this work? Since the lower-bound constraints are all satisfied, it will be possible to apply a standard circulation algorithm (without lower flow bounds) to solve the problem.

Let's carry out this plan. Recall our "pseudo-flow" f_0. For each $v \in V$, let $f_0^{\text{in}}(v)$ and $f_0^{\text{out}}(v)$ denote the flow in and flow out of vertex v. Let $\chi(v)$ denote the *excess flow* coming into v in f_0, that is

$$\chi(v) \;=\; f_0^{\text{in}}(v) - f_0^{\text{out}}(v) \;=\; \sum_{(u,v)\in E} \ell(u,v) - \sum_{(v,w)\in E} \ell(v,w).$$

Note that this may be negative, which means that we have a flow deficit. We will adjust the supply and demand values so that computing any valid circulation f_1 for the adjusted values and combining this with f_0, we will obtain a flow that satisfies all the requirements.

How do we adjust the supply/demand values? For each v, we want to cancel out the excess $\chi(v)$ coming in and generate a net flow of $d(v)$ units coming in. That is, we want f_1 to satisfy

$$f_1^{\text{in}}(v) - f_1^{\text{out}}(v) \;=\; d(v) - \chi(v).$$

To achieve this we can change the demand at vertex v to be $d(v) - \chi(v)$. We'll call this $d'(v)$.

But what about the edge capacities? Recall that we had an upper capacity of $c(u,v)$, but we are essentially "forcing" $\ell(u,v)$ units of flow. So, the remaining capacity is $c(u,v) - \ell(u,v)$. We'll call this $c'(u,v)$.

We are now ready to put the pieces together. Given the network $G = (V,E)$ as input (with vertex demands $d(v)$ and lower and upper flow bounds $\ell(u,v)$ and $c(u,v)$), we do the following (see Fig. 5):

1. Create an initial pseudo-circulation f_0 by setting $f(u,v) = \ell(u,v)$.
2. Create a new network $G' = (V', E')$ that has all the same vertices and edges as G (that is, $V' \leftarrow V$ and $E' \leftarrow E$):
   (a) For each $(u,v) \in E'$, set its adjusted capacity to $c'(u,v) \leftarrow c(u,v) - \ell(u,v)$.
   (b) For each $v \in V'$, set its adjusted demand to $d'(v) + \leftarrow d(v) - \chi(v)$, where, $\chi(v) = f_0^{\text{in}}(v) - f_0^{\text{out}}(v)$.
3. Compute a (standard) circulation f_1 in G'. If it does not exist, return "no circulation!" Otherwise, return the final flow $f = f_0 + f_1$ (see Fig. 6(c)).

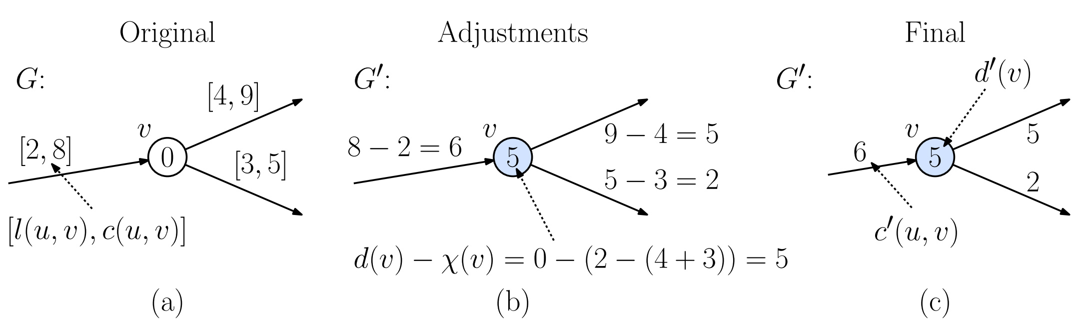

Fig. 5: Eliminating lower bound constraints. The net pseudo-flow into v is $\chi(v) = 3-(4+3) = -4$.

To establish the correctness of our reduction, we prove below that its output $f_0 + f_1$ is a valid circulation for G (with lower flow bounds) if and only if f_1 is a valid circulation circulation for G'.

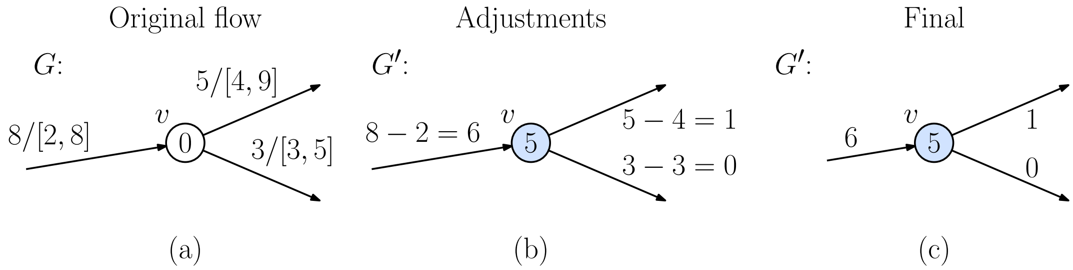

Fig. 6: Adjusting flows between original network and transformed network. Observe that v satisfies the demand requirements, since $6 - (1 + 0) = 5 = d'(v)$.

Lemma: The network G (with both lower and upper flow bounds) has a feasible circulation if and only if G' (with only upper capacity bounds) has a feasible circulation.

Proof: (Sketch. See Kleinberg and Tardos for a formal proof.) Intuitively, if G' has a feasible circulation f' then the circulation $f(u, v) = f'(u, v) + \ell(u, v)$ can be shown to be a valid circulation for G and it satisfies the lower flow bounds. Conversely, if G has a feasible circulation (satisfying both the upper and lower flow bounds), then let $f'(u, v) = f(u, v) - \ell(u, v)$. As above, it can be shown that f' is a valid circulation for G'. (Think of f' as f_1 and f as $f_0 + f_1$.)

Note that (as in the original circulation problem) we have not presented a new algorithm. Instead, we have shown how to *reduce* the current problem (circulation with lower and upper flow bounds) to a problem we have already solved (circulation with only upper bounds). Again, the running time will be the sum of the time to perform the reduction, which is easily seen to be $O(n + m)$ plus the time to compute the circulation, which as we have seen reduces to the time to compute a maximum flow, which according to the current best technology is $O(nm)$ time.

Application: Survey Design: To demonstrate the usefulness of circulations with lower flow bounds, let us consider an application problem that arises in the area of data mining. A company sells k different products, and it maintains a database which stores which customers have bought which products recently. We want to send a survey to a subset of n customers. We will tailor each survey so it is appropriate for the particular customer it is sent to. Here are some guidelines that we want to satisfy:

- The survey sent to a customer will ask questions only about the products this customer has purchased.
- We want to get as much information as possible, but do not want to annoy the customer by asking too many questions. (Otherwise, they will simply not respond.) Based on our knowledge of how many products customer i has purchased, and easily they are annoyed, our marketing people have come up with two bounds $0 \leq c_i^- \leq c_i^+$. We will ask the ith customer about at least c_i^- products they bought, but (to avoid annoying them) at most c_i^+ products.
- Again, our marketing people know that we want more information about some products (e.g., new releases) and less about others. To get a balanced amount of information

about each product, for the jth product we have two bounds $0 \le p_j^- \le p_j^+$, and we will ask at least p_j^- customers about this product and at most p_j^+ customers.

We can model this as a bipartite graph G, in which the customers form one of the parts of the network and products form the other part. There is an edge (i, j) if customer i has purchased product j. The flow through each customer node will reflect the number of products this customer is asked about. The flow through each product node will reflect the number of customers that are asked about this product.

This suggests the following network design. Given the bipartite graph G, we create a directed network as follows (see Fig. 7).

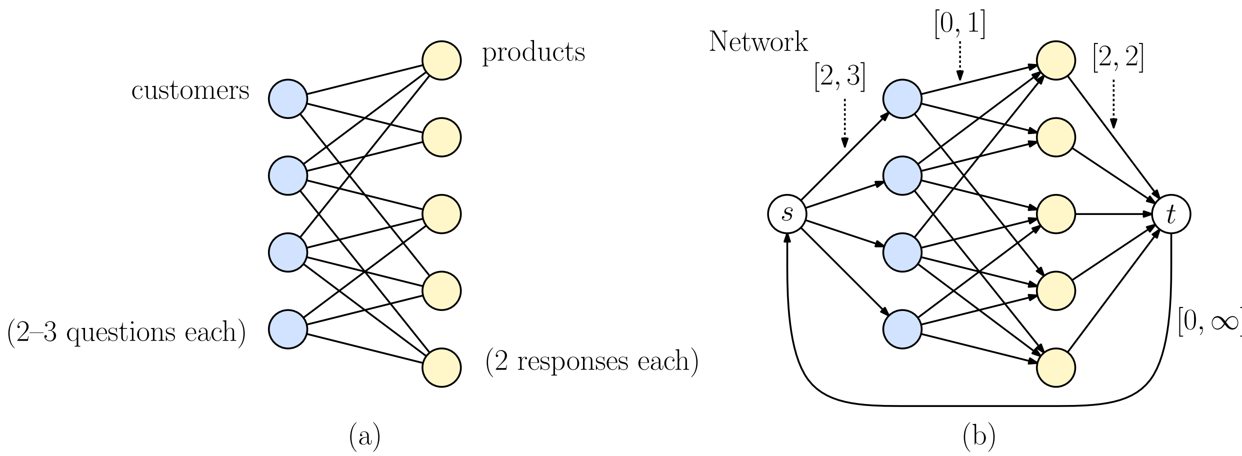

Fig. 7: Reducing the survey design problem to circulation with lower and upper flow bounds. (Assuming $[c_i^-, c_i^+] = [2, 3]$ for all customers and $[p_j^-, p_j^+] = [2, 2]$ for all products.)

- For each customer i who purchased product j we create a directed edge (i, j) with an upper flow bounds of 1, respectively. This models the requirement that customer i will be surveyed at most once about product j, and customers will be asked only about products they purchased.
- We create a source vertex s and connect it to all the customers, where the edge from s to customer i has lower and upper flow bounds of c_i^- and c_i^+, respectively. This models the requirement that customer i will be asked about at least c_i^- products and at most c_i^+.
- We create a sink vertex t, and create an edge from product j to t with lower and upper flow bounds of p_j^- and p_j^+. This models the requirement that there are at least p_j^- and at most p_j^+ customers will be asked about product j.
- We create an edge (s, t). Its lower bound is set to zero and its upper bound can be set to any very large value. This is needed for technical reasons, since we want a circulation.
- All node demands are set to 0.

The correctness of the reduction is established in the following lemma.

Lemma: There exists a valid circulation in G if and only there is a valid survey design.

Proof: ($\Rightarrow$) Suppose that G has a valid (integer-valued) circulation. We create a survey as follows. For each customer-product edge (i, j) that carries one unit of flow, customer i is surveyed about product j. We will show that this is a valid survey. By definition of the edges, customers are surveyed only about products they purchased. From our capacity constraints and the fact that demands are all zero, it follows that the total flow into each customer node is in the interval $[c_i^-, c_i^+]$, implying that this customer is asked about the proper range of products. Similarly, the total flow out of each product node is in the interval $[p_j^-, p_j^+]$, implying that this product is involved in the proper number of surveys. Therefore, this is a valid survey, as desired (see Fig. 8).

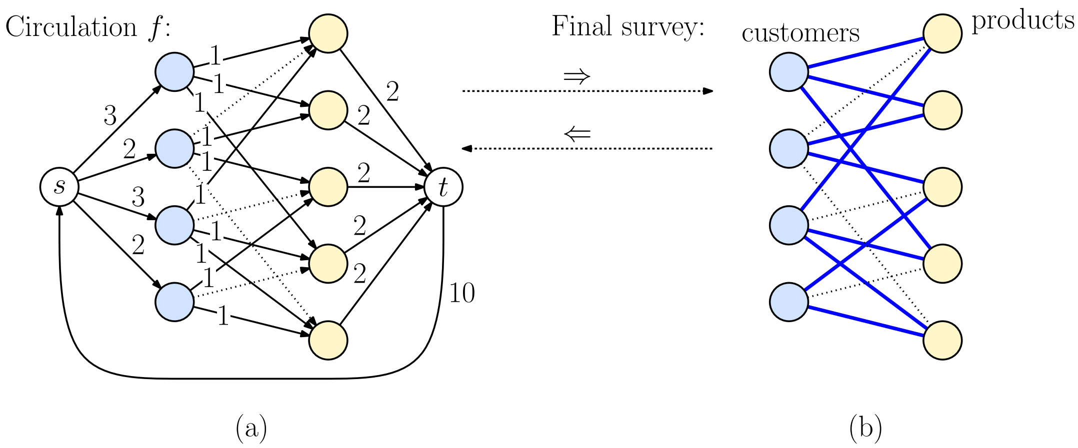

Fig. 8: Correctness of the survey-design reduction to circulation.

($\Leftarrow$) Suppose that there is a valid survey design. We construct a flow in G as follows. For each customer-product pair (i, j) involved in the survey, we create a flow of one unit on edge (i, j), and otherwise we set this flow to 0. Since we only survey pairs where a purchase took place, this edge exists in G. Next, for each customer i, we set the flow along the edge (s, i) to the total number of surveys sent to customer i. For each product j, we set the flow along the edge (j, t) to the total number of surveys involving product j. Finally, to keep everything balanced, we set the flow on edge (t, s) to the total number of surveys.

We will show that this is a valid circulation in G. Because the survey is valid, the number of surveys sent to customer i is in interval $[c_i^-, c_i^+]$, satisfying this edge's capacity constraints. The flow on each (i, j) edge is either 0 or 1, satisfying this edge's capacity constraints. Also, by the validity of the survey, the number of surveys involving product j is in the interval $[p_j^-, p_j^+]$, satisfying this edge's capacity constraints. Finally, observe that the flow into each vertex equals the flow out of this vertex, implying that all the vertex demands (zero) are satisfied. Therefore f is a valid circulation in G, as desired (see Fig. 8).

Summary: We introduced the notion of a circulation in a network. Unlike network flows, each individual vertex has requirements for the net flow through this vertex. These can be used for various applications. We showed that circulations can be solved by any standard network flow algorithm. We also showed that a generalization where edges can have both lower and upper capacities can be reduced to the standard circulation problem (which has only upper capacity bounds). Circulations have many applications, and we one example from survey design.