1 Circulations (50 points)

Let’s look at some variations of max-flow. In class, we talked about algorithms to compute the maximum flow from a source $s$ to a sink $t$ in a directed graph $G = (V, E)$ when we are given edge capacities $c : E \rightarrow \mathbb{R}^+$. A valid flow was a function $f : E \rightarrow \mathbb{R} \geq 0$ so that $f(e) \leq c(e)$ for all $e \in E$, and for all nodes $v \in V \setminus \{s, t\}$ we required the flow in to equal the flow out, i.e. $\sum_u f(u, v) = \sum_u f(v, u)$. (In class we modified this a bit to add skew-symmetry, but that was a mathematical convenience – this is the original and equivalent definition).

(a) (25 points) What if we no longer have a single $s$ and a single $t$? Instead, think of every node $v$ as having a demand $d_v$. If $d_v$ is positive then $v$ wants flow; it is a sink. If $d_v$ is negative then $v$ wants to get rid of flow; it is a source. And if $d_v = 0$ then $v$ wants to simply transit flow; it is neither a source nor a sink. A valid circulation is an assignment $f : E \rightarrow \mathbb{R} \geq 0$ such that $\ell(e) \leq f(e) \leq c(e)$ for all $e \in E$, and $\sum_u f(u, v) - \sum_u f(v, u) = d_v$ for all $v \in V$ (i.e. at every node the flow-in minus the flow-out is equal to the demand).

Given a directed graph with capacities and demands, give a polynomial-time algorithm to determine whether a valid circulation exists. Prove correctness and polynomial running time. Hint: reduce to max-flow.

(b) (25 points) Let’s make things even more complicated. Capacities are an upper bound on flow – we are not allowed to send more flow through an edge than its capacity allows. What if we also want to have lower bounds? That is, in addition to capacities $c$, we are also given lower bounds $\ell : E \rightarrow \mathbb{R}_{\geq 0}$ and we require $\ell(e) \leq f(e) \leq c(e)$ for all $e \in E$?

Let’s redefine a valid circulation with these lower bounds. We are now given a directed graph $G = (V, E)$, capacities $c : E \rightarrow \mathbb{R}^+$, lower bounds $\ell : E \rightarrow \mathbb{R}_{\geq 0}$, and for each vertex $v$ we are given a demand $d_v$. A valid circulation is now a function $f : E \rightarrow \mathbb{R} \geq 0$ such that $\ell(e) \leq f(e) \leq c(e)$ for all $e \in E$, and $\sum_u f(u, v) - \sum_u f(v, u) = d_v$.

Give a polynomial-time algorithm to determine whether a valid circulation exists. Prove correctness and polynomial running time. Hint: Try to transform the problem into finding a circulation without lower bounds, i.e. the setting of part (a). You might want to start by initially setting $f(e) = \ell(e)$ for all $e$. This will not be a valid circulation, but what does the “remaining” problem look like?
2 Realizable Degree Sequences (50 points)

Recall that the in-degree of a vertex $v$ in a directed graph is the number of edges that go into $v$, i.e. the number of edges of the form $(u, v)$. Similarly, the out-degree is the number of edges leaving $v$, i.e. the number of edges of the form $(v, u)$. Let $d_{v,\text{in}}$ denote the in-degree of $v$, and let $d_{v,\text{out}}$ denote the out-degree. In this problem we will not allow any loops (edges of the form $(u, u)$) or multiedges (i.e. any edge $(u, v)$ can only be in the graph at most once, although both $(u, v)$ and $(v, u)$ could be in the graph).

(a) (15 points) Suppose you are given the following set of degree constraints for a four-node directed graph:

- $d_{1,\text{in}} = 0, d_{1,\text{out}} = 2$
- $d_{2,\text{in}} = 1, d_{2,\text{out}} = 2$
- $d_{3,\text{in}} = 1, d_{3,\text{out}} = 1$
- $d_{4,\text{in}} = 3, d_{4,\text{out}} = 0$

Is there a directed graph, with no multi-edges or self-loops, that satisfies these constraints? If so, what is it? If not, why not?

(b) (35 points) This type of specification, where we are given the in- and out-degrees of every vertex, is called a degree sequence. The question above was whether a particular degree sequence is realizable – that is, whether there exists a graph having those degrees.

Give a polynomial-time algorithm that, given a degree sequence, will determine if it is realizable, and if it is will produce a directed graph with those degrees. The graph should not have any self loops or multi-edges (i.e. for each directed pair $(i, j)$ with $i \neq j$ there can be at most one edge from $i$ to $j$, although it is fine if there is also an edge from $j$ to $i$). Prove that your algorithm is correct, and runs in polynomial time.

Hint: think flow with integer capacities.