EECS 127/227AT Optimization Models in Engineering
Spring 2020 Discussion 10

1. **The max-flow min-cut theorem**

In this question, we explore how strong duality can be used to solve a canonical problem in network theory. Specifically, we will prove the *max-flow min-cut theorem*, which can be used to calculate the maximum flow through a network of interest. We will first introduce this theorem by means of an example, then prove it for a fairly general case.

Problem definition. Consider a directed graph ("digraph") $G = (V, E)$, where V denotes a set of vertices of size n and $E \subseteq V \times V$ denotes a set of edges of size m. Note that the elements of E are ordered pairs of vertices, and we will refer to an edge $e = (u, v) \in E$ as an *incoming* edge of v and an *outgoing* edge of u. We define two vertices with special properties: a "source" s, out of which data (or water, or current . . .) is flowing, and a "sink" t, into which data is flowing. An example graph with these properties is shown in Fig. 1.

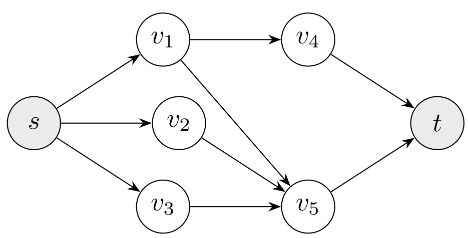
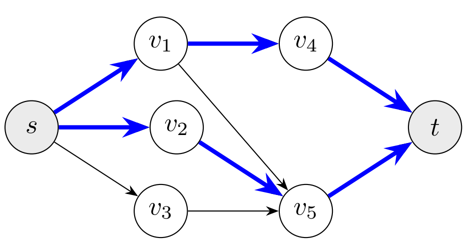
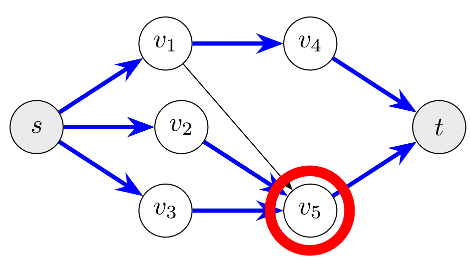


Figure 1: An example digraph with source s and sink t.

We define the *max-flow* problem as the problem of transporting the maximum amount of data from source s to destination t, assuming that s can generate infinite data but we are subject to capacity constraints on each edge of the digraph, and no data can leave the network through any node but t (i.e., for all nodes except s and t, flow in is equal to flow out). For this problem, we will assume that each edge e can support at most one unit of data flow, though the theorem holds in the more general case where edges have different capacities. Example valid and invalid "flows" for the graph above are shown below in Fig. 2, including the maximum flow.



(a) Valid maximum flow from source to sink of 2 units.



(b) Invalid flow of 3 units from source — flow into v_5 is not equal to flow out of v_5.

Figure 2: Valid maximum *(left)* and invalid *(right)* flows through the example digraph.

Lastly, we define the *min-cut* problem as the problem of partitioning the vertices of the digraph into two pieces, with s on one side and t on the other, while slicing along edges with the minimum total flow capacity. (Note that in our problem, where all edges have an equal capacity of one, this is equivalent to finding the cut that intersects the minimum number of edges.) Several possible "cuts" for the graph above are shown below in Fig. 3, including the minimum cut.

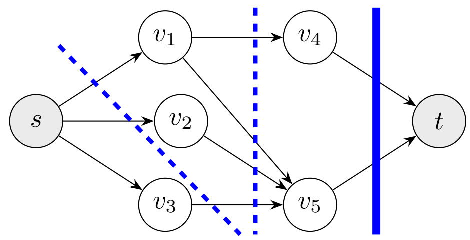


Figure 3: Example cuts of the example digraph, including one of several possible minimum cuts *(solid)* that slices through a capacity of 2 units.

The *max-flow min-cut* theorem states that the solutions to the *max-flow* and *min-cut* problems are equal — i.e., the maximum flow through a digraph of this form is exactly equal to the total capacity of all edges sliced in the minimum cut. This may be intuitive from the figures above: conceptually, the *max-flow* problem is computing flow directly, and the *min-cut* problem is finding the "bottleneck" that is preventing more data from flowing. We will now prove this theorem mathematically using duality.

(a) ***Formulating the max-flow problem (primal).*** To calculate the maximum flow from s to t through a general network, we first define $\vec{f} \in \mathbb{R}^m$ as the m-dimensional vector whose entries $f_{(u,v)} \geq 0$ are the flow through each edge (u, v). The *max-flow* problem is then the problem of maximizing the sum of these flows $f_{(s,v)}$ out of the source s (or, equivalently, the sum of the flows $f_{(v,t)}$ into t, the sink[1]) while obeying the constraints of the network, i.e.,

edge $s \to v$.

$$\max_{\vec{f} \geq \vec{0}} \sum_{v:(s,v)\in E} f_{(s,v)} \qquad \text{(Max-Flow-LP)}$$

$$\text{s.t.} \quad \underbrace{\sum_{u:(u,v)\in E} f_{(u,v)}}_{\text{Incoming flow}} = \underbrace{\sum_{u:(v,u)\in E} f_{(v,u)}}_{\text{outgoing flow.}} \quad \forall v \notin \{s, t\}$$

$$f_{(u,v)} \leq 1 \quad \forall (u, v) \in E.$$

$f(u,v) - 1 \leq 0$ $\qquad$ $\lambda_{u,v}(f(u,v) - 1)$

Note that the first set of constraints enforces that flow into and out of each non-source/sink node is equal, and the second that flow on each edge does not exceed capacity. We also restrict each entry of $\vec{f}$ to be nonnegative, since negative flow would be physically nonsensical.

as many λ as edges.

i. Compute the Lagrangian $\mathcal{L}(\vec{f}, \vec{\lambda}, \vec{\mu})$ of the above formulation by introducing auxiliary variables μ_v for each of the equality constraints and $\lambda_{(u,v)}$ for each of the inequality constraints. Note that $\vec{\lambda} \in \mathbb{R}^m$ and $\vec{\mu} \in \mathbb{R}^n$ for our graph of m edges and n vertices.

as many μ as {num nodes − 2}

[1]These values are equivalent for any digraph in which s has no incoming edges and t has no outgoing edges.

Solution: The Lagrangian of the linear program is

$$\mathcal{L}(\vec{f}, \vec{\lambda}, \vec{\mu}) = \sum_{v:(s,v)\in E} f_{(s,v)} \qquad f_0(f) \text{ primal objective}$$

$$p^* = \max_{f\geq 0} \min_{\lambda,\mu} L(f,\lambda,\mu) \qquad + \sum_{v\notin\{s,t\}} \mu_v \left[\sum_{u:(u,v)\in E} f_{(u,v)} - \sum_{u:(v,u)\in E} f_{(v,u)} \right]$$

$$+ \sum_{(u,v)\in E} \lambda_{(u,v)}(1 - f_{(u,v)}).$$

Note the sign of the final term; because the primal is a maximization, the dual will be a minimization, and thus for $\vec{\lambda} \geq \vec{0}$ the objective should approach $-\infty$ with increasing $\vec{\lambda}$ when the constraint is violated.

ii. Formulate the dual of the linear program Max-Flow-LP. For simplicity of formulation, assume that there exists no edge between s and t, i.e., $(s,t) \notin E$.

Solution: We first define dual function $g(\vec{\lambda}, \vec{\mu})$, which is exactly

$$g(\vec{\lambda}, \vec{\mu}) = \max_{\vec{f}\geq\vec{0}} \mathcal{L}(\vec{f}, \vec{\lambda}, \vec{\mu}).$$

To allow us to meaningfully reorder the terms in the Lagrangian and extract the constraints, we first introduce some new notation. For a given vertex $v \in V$, let $I_v \subseteq E$ and $O_v \subseteq E$ to denote the set of incoming and outgoing edges to and from v, respectively. Note that $I_s = O_t = \varnothing$. We can now rearrange the terms in our Lagrangian summations into summations over all output edges of s, all input edges of t, and all other edges, resulting in new dual function expression

$$g(\vec{\lambda}, \vec{\mu}) = \max_{\vec{f}\geq\vec{0}} \sum_{(s,v)\in O_s} f_{(s,v)}\underbrace{(1 + \mu_v - \lambda_{(s,v)})}_{\leq 0} + \sum_{(v,t)\in I_t} f_{(v,t)}(-\mu_v - \lambda_{(v,t)})$$

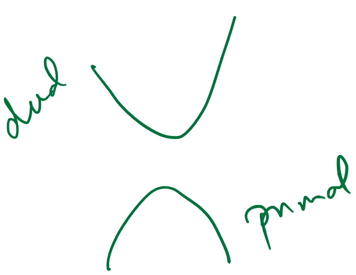
$$+ \sum_{\substack{(u,v)\in E \\ (u,v)\notin I_t\cup O_s}} f_{(u,v)}(\mu_v - \mu_u - \lambda_{(u,v)}) + \sum_{(u,v)\in E} \lambda_{(u,v)}.$$

Assuming that there exist no edges from s to t as given, we have now expressed $g(\vec{\lambda}, \vec{\mu})$ as a weighted sum of all $f_{(u,v)}$ values, plus a summation of $\lambda_{(u,v)}$ values. Because each $f_{(u,v)}$ is nonnegative, we now note that if the coefficient of any $f_{(u,v)}$ is positive, the maximization problem is unbounded above. Writing the final dual problem, we therefore restrict these coefficients to be nonpositive in our constraint set and update our objective accordingly. The final dual problem is thus

$$\min_{\vec{\lambda}\geq\vec{0},\ \vec{\mu}} \sum_{(u,v)\in E} \lambda_{(u,v)} \qquad \text{(Max-Flow-Dual)}$$

$$\begin{aligned} \text{s.t. } & 1 + \mu_v - \lambda_{(s,v)} \leq 0 \quad \forall (s,v) \in O_s \\ 0 & - \mu_v - \lambda_{(v,t)} \leq 0 \quad \forall (v,t) \in I_t \\ & \mu_v - \mu_u - \lambda_{(u,v)} \leq 0 \quad \forall (u,v) \in E, (u,v) \notin I_t \cup O_s. \end{aligned}$$

$\mu_v - (-1)$



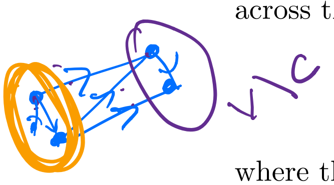
$$\boxed{p^* \leq d^*}$$

max flow $=$ p^* (primal LP)

strong duality

min cut $=$ d^* (dual LP)

(b) ***Formulating the min-cut problem (dual).*** Recall that the *min-cut* problem is the problem of partitioning our digraph $G = (V, E)$ into two sides while slicing through the minimum number of edges. To formalize this, we define a *cut* C in G as a partition of V into two sets C and $V \setminus C$ such that $s \in C$ and $t \in V \setminus C$. The *min-cut* solution is thus the total capacity across the cut that crosses the minimum number of edges, i.e., the minimum value of



$$q(C) \doteq \sum_{(u,v)\in E} \mathbb{I}_{\{u\in C \cap v \notin C\}},$$

edges b/w node in C & not in C

$q(C)=3$

where the indicator function $\mathbb{I}_{\{\cdot\}}$ is equal to 1 for values in the subscript set and 0 otherwise. Note that when the capacity of each edge is one, as we assume here, $q(C)$ is exactly the number of edges crossed by a partition into C and $V \setminus C$. For clarity, we denote the minimum cut value $q^* \doteq q(C^*)$ for optimal cut C^*.

We will now show that the Max-Flow-Dual problem we formulated above is equivalent to computing this minimum cut. We first rewrite this dual as

$$d^* \quad \min_{\lambda\geq 0,\ \mu_v} \sum_{(u,v)\in E} \lambda_{(u,v)} \qquad \text{(Min-Cut-LP)}$$
$$\text{s.t. } \mu_v - \mu_u - \lambda_{(u,v)} \leq 0 \quad \forall (u,v) \in E$$
$$\mu_s = -1,\ \mu_t = 0.$$

min cut problem $\min_C q(C)$ minimal: optimal cut

$n-2$ μ dual variable

n μ dual variable

Note that this problem is exactly equivalent to Max-Flow-Dual; we can rewrite all 3 sets of Max-Flow-Dual constraints as one by enforcing the values of μ_s and μ_t as indicated. We refer to this as the Min-Cut-LP, though we have not yet shown that it is equivalent to calculating the minimum cut. We will show this equivalence by proving that the solution of Min-Cut-LP both upper and lower bounds the minimum value of $q(C)$.

i. Show that the optimal value of Min-Cut-LP is at most $q^* = q(C^*)$, the value of the minimum cut of G.
Hint: Show that for any cut C, the optimal value of Min-Cut-LP is less than $q(C)$.
Solution: For any cut C, we can define a feasible point $(\vec{\lambda}, \vec{\mu})$ of Min-Cut-LP as follows:

$d^* \leq q(C^*)$

$$\mu_v = \begin{cases} -1 & v \in C \\ 0 & v \notin C \end{cases}, \qquad \lambda_{(u,v)} = \max\{0, \mu_v - \mu_u\}. \geq 0$$

C

is when it is 1.

From the above definition, we see that $\lambda_{(u,v)} = 1$ if and only if $u \in C$ and $v \notin C$ (i.e., if the edge (u,v) is cut by the partition) and is otherwise uniformly 0. Thus, our Min-Cut-LP objective function $\sum_{(u,v)\in E} \lambda_{(u,v)}$ is exactly equal to $q(C)$ at this feasible point. This means that $q(C)$ constitutes an upper bound for the value of Min-Cut-LP, since optimizing over $\vec{\lambda}$ and $\vec{\mu}$ could possibly lower the value of the objective function. Since this bound holds for all cuts C, it holds for $q(C^*) = q^*$, i.e., the optimal value of Min-Cut-LP is at most q^*, as desired.

$d^* \leq$ value at feasible λ, μ $= q(C)$ for all C

$\Rightarrow d^* \leq q(C^*)$

ii. (Optional) Show that the optimal value of Min-Cut-LP is at least $q^* = q(C^*)$, the value of the minimum cut of G.
Hint: For an arbitrary feasible $(\vec{\lambda}, \vec{\mu})$, first sort the distinct values of μ_v in increasing order, then consider cuts of the form $C_\alpha \doteq \{v : \mu_v \leq \alpha\}$ for different values $\alpha \in \mathbb{R}$.
Note: The proof of this inequality involves a combinatorial argument and is beyond the scope of this class. Do not feel obligated to understand it in full; we present it for completeness.

Solution: Consider an arbitrary feasible point $(\vec{\lambda}, \vec{\mu})$. Let $\alpha_1 < \alpha_2 < \cdots < \alpha_k \in \mathbb{R}$ be the k distinct values that the μ_v take for $v \in V$. Note that there must exist indices $l, p \in \{1, \ldots, k\}$ such that $\alpha_l = -1$ and $\alpha_p = 0$ because the source and destination vertices have dual variables constrained to $\mu_s = -1$ and $\mu_t = 0$, respectively. For every α, let $C_\alpha \doteq \{v \in V : \mu_v \leq \alpha\}$ and let $F \doteq \{(u,v) \in E : \mu_u \leq \mu_v\}$. Note that when $\alpha \in [-1, 0)$, C_α is a cut that satisfies $s \in C_\alpha$ and $t \notin C_\alpha$.

We now lower bound the optimal value of Min-Cut-LP as follows:

$$\sum_{(u,v)\in E} \lambda_{(u,v)} \geq \sum_{(u,v)\in F} \lambda_{(u,v)} \geq \sum_{(u,v)\in F} (\mu_v - \mu_u) = \sum_{i=1}^{k-1} (\alpha_{i+1} - \alpha_i) q(C_{\alpha_i})$$

The first inequality proceeds from the nonnegativity of $\vec{\lambda}$, and the second from the dual's set of inequality constraints $\mu_v - \mu_u - \lambda_{(u,v)} \leq 0 \Rightarrow \mu_v - \mu_u \leq \lambda_{(u,v)}$.

The subsequent equality proceeds from a combinatorial argument as follows. Consider a single edge $(u,v) \in F$ such that $\mu_u = \alpha_{i_{(u,v)}}$ and $\mu_v = \alpha_{j_{(u,v)}}$; we can then write each term in the previous summation as

$$\mu_v - \mu_u = \sum_{k=i_{(u,v)}}^{j_{(u,v)}-1} (\alpha_{k+1} - \alpha_k).$$

This reformulation is necessary to write the terms as a function of index k. (To see this equality more explicitly, write out the sum above and note that all terms except $\alpha_{j_{(u,v)}}$ and $\alpha_{i_{(u,v)}}$ cancel.) Plugging each term back into the final summation, we have

$$\sum_{(u,v)\in F} (\mu_v - \mu_u) = \sum_{(u,v)\in F} \sum_{k=i_{(u,v)}}^{j_{(u,v)}-1} (\alpha_{k+1} - \alpha_k) = \sum_{i=1}^{k-1} (\alpha_{i+1} - \alpha_i) q(C_{\alpha_i}).$$

Here, the second equality results from a subtle counting argument. Consider a single term $\alpha_{k+1} - \alpha_k$; the number of times this term occurs in the sum is the number of edges for which $\mu_u \leq \alpha_k$ and $\mu_v \geq \alpha_{k+1}$. This is exactly the number of edges from C_{α_k} to $V \setminus C_{\alpha_k}$, or $q(C_{\alpha_k})$.

We can now return to bounding our Min-Cut-LP objective function, and write

$$\begin{aligned}
\sum_{(u,v)\in E} \lambda_{(u,v)} &\geq \sum_{i=1}^{k-1} (\alpha_{i+1} - \alpha_i) q(C_{\alpha_i}) \\
&\geq \sum_{i=l}^{p-1} (\alpha_{i+1} - \alpha_i) \underline{q(C_{\alpha_i})} \quad \geq q^* = q(C^*) \\
&\geq q^* \sum_{i=l}^{p-1} (\alpha_{i+1} - \alpha_i) \\
&= q^*.
\end{aligned}$$

Since the above bound holds for all feasible points $(\vec{\lambda}, \vec{\mu})$, q^* constitutes a lower bound on the value of Min-Cut-LP as desired, concluding the proof of the statement.

(c) ***Concluding.*** Conclude that the *max-flow min-cut* theorem holds for the examined set of digraphs.

Solution: From parts (b)i. and (b)ii. above, we have that the value of Min-Cut-LP is both upper and lower bounded by the value of the minimum cut of G, and is therefore equal to the value of that minimum cut. This means that the dual of the *max-flow* problem is exactly the *min-cut* problem. The primal problem is convex (linear) and Slater's condition holds (consider an arbitrarily small flow through one branch of the network), so strong duality holds. Thus, the value of maximum flow is equal to that of the minimum cut, and the *max-flow min-cut* theorem holds.