CO 250: Introduction to Optimization

Module 3: Duality through Examples

Recap: Shortest Paths

In an instance of the shortest path problem, we are given

- a graph $G = (V, E)$, a non-negative length c_e for each edge $e \in E$, and
- a pair of vertices s and t in V.

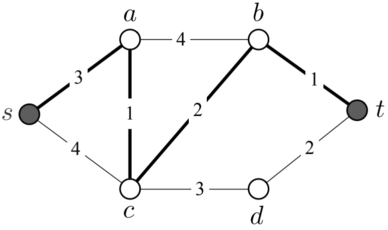
Our goal is to compute an s, t-path P of smallest total length.

Recall: an s, t-path is a sequence

$$P := u_1u_2, u_2u_3, \ldots, u_{k-1}u_k$$

where

- $u_iu_{i+1} \in E$ for all i, and
- $u_1 = s$, $u_k = t$, and $u_i \neq u_j$ for all $i \neq j$.



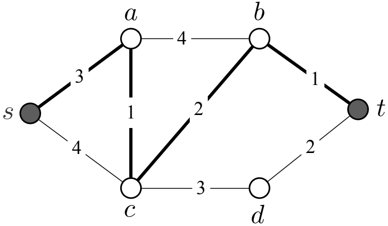
Its length is given by

$$c(P) = c_{u_1u_2} + c_{u_2u_3} + \ldots + c_{u_{k-1}u_k}$$

In the example, we see by inspection that

$$P = sa, ac, cb, bt$$

is a shortest path and that its length is 9.



Question

1. Given a shortest-path instance and a candidate shortest s, t-path P, is there a short proof of its optimality?
2. How can we find a shortest s, t-path?

We will answer both questions in this module. This lecture focus on question 1.

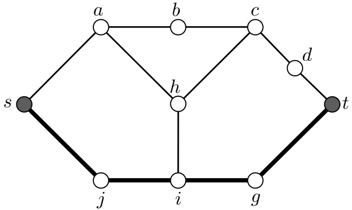
Shortest Paths: Finding an Intuitive Lower Bound

Cardinality Case

To make our lives easier, we will first consider the cardinality special case of the shortest path problem.

We consider shortest path instances where...

- each edge $e \in E$ has length 1, and
- we are therefore looking for an s, t-path with the smallest number of edges.



Example: In the diagram above, one easily sees that

$$P = sj, ji, ig, gt$$

is a shortest s, t-path.

How can we prove this fact?
$\longrightarrow$ The answer lies in s,t-cuts!

Definition

For $U \subseteq V$, we define

$$\delta(U) = \{uv \in E \,:\, u \in U, v \notin U\}$$

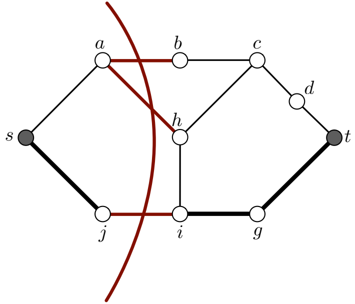
and call it an s, t-cut if $s \in U$, and $t \notin U$.

Example

Let $U = \{s, a, j\}$. It follows that

$$\delta(U) = \{ab, ah, ji\}$$

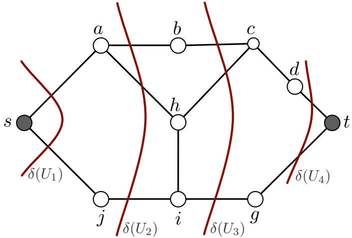
is an s, t-cut.



Recall:

- If P is an s, t-path and $\delta(U)$ an s, t-cut, then P contains an edge of $\delta(U)$.
- If $S \subseteq E$ contains an edge from every s, t-cut, then S contains an s, t-path.

The example on the right shows 4 s,t-cuts, $\delta(U_1), \delta(U_2), \delta(U_3), \delta(U_4)$.



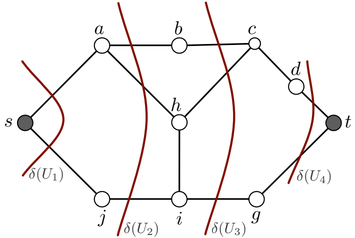
Two important notes:

(1) $\delta(U_i) \cap \delta(U_j) = \emptyset$ for $i \neq j$ and

(2) an s,t-path must contain an edge from $\delta(U_i)$ for all i.

$$\begin{aligned} \delta(U_1) &= \{sa, sj\} \\ \delta(U_2) &= \{ab, ah, ji\} \\ \delta(U_3) &= \{bc, hc, ig\} \\ \delta(U_4) &= \{dt, gt\} \end{aligned}$$

$\longrightarrow$ Every s,t-path must have at least 4 edges.
$\longrightarrow$ sj, ji, ig, gt is a shortest s,t-path!



Question

Notice: hi is not in any of the $\delta(U_i)$. Does this mean that hi is not on any shortest s,t-path?

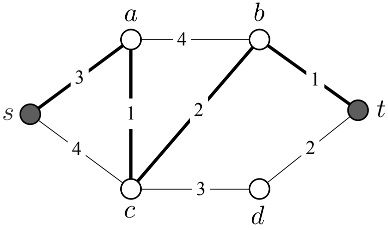
Yes!

An s,t-path that contains hi must also contain an edge from each of the s,t-cuts $\delta(U_i)$. $\longrightarrow$ It must contain at least 5 edges!

$$\begin{aligned}
\delta(U_1) &= \{sa, sj\} \\
\delta(U_2) &= \{ab, ah, ji\} \\
\delta(U_3) &= \{bc, hc, ig\} \\
\delta(U_4) &= \{dt, gt\}
\end{aligned}$$

Back to the General Case

In general instances, we assign a non-negative width y_U to every s,t-cut $\delta(U)$.



Definition

A width assignment $\{y_U : \delta(U)\ s,t\text{-cut}\}$ is feasible if, for every edge $e \in E$, the total width of all cuts containing e is no more than c_e.

Using math: y is feasible if for all e

$$\sum (y_U : \delta(U)\ s,t\text{-cut and } e \in E) \leq c_e$$

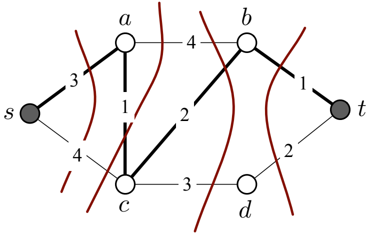
Back to the General Case

Consider the example on the right with 4 s, t-cuts.

The width assignment

$$\begin{aligned} y_{U_1} &= 3 \\ y_{U_2} &= 1 \\ y_{U_3} &= 2 \\ y_{U_4} &= 1 \end{aligned}$$

is easily checked to be feasible.



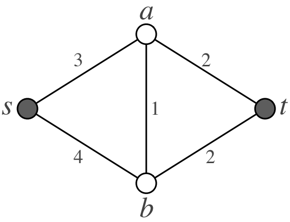
$$\begin{aligned} U_1 &= \{s\} \\ U_2 &= \{s, a\} \\ U_3 &= \{s, a, c\} \\ U_4 &= \{s, a, b, c, d\} \end{aligned}$$

Back to the General Case

Proposition

If y is a feasible width assignment, then any s,t-path must have length at least

$$\sum(y_U \ : \ U \ \ s,t\text{-cut}).$$



Example:

$$y_{U_1} + y_{U_2} + y_{U_3} + y_{U_4} = 7$$

$\longrightarrow$ Path sa, ac, cb, bt is a shortest path!

$$\begin{aligned} U_1 &= \{s\} \\ U_2 &= \{s,a\} \\ U_3 &= \{s,a,c\} \\ U_4 &= \{s,a,b,c,d\} \end{aligned}$$

Back to the General Case

Proposition

If y is a feasible width assignment, then any s,t-path must have length at least

$$\sum (y_U \,:\, U \ s,t\text{-cut}).$$

Example:

$$y_{U_1} + y_{U_2} + y_{U_3} + y_{U_4} = 7$$

$\longrightarrow$ Path sa, ac, cb, bt is a shortest path!

Proof: Consider an s,t-path P. It follows that

$$\begin{aligned} c(P) &= \sum (c_e \,:\, e \in P) \\ &\geq \sum \Big(\sum (y_U \,:\, e \in \delta(U)) \,:\, e \in P\Big) \\ &\geq \sum (y_U \,:\, \delta(U) \ s,t\text{-cut}) \end{aligned}$$

where the last inequality follows from the feasibility of y.

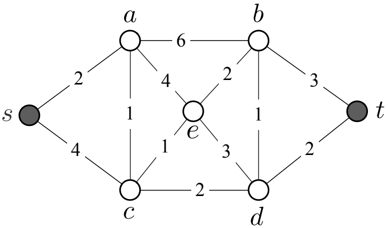
Note: if $\delta(U)$ is an s,t-cut, then P contains at least one edge from $\delta(U)$.
$\longrightarrow$ Variable y_U appears at least once on the right-hand side above, and hence ... $\square$

One More Example

Question: Can you spot a shortest s, t-path?
$\longrightarrow$ $P = sa, ac, cd, dt$ of length 7.

Question: Can you prove your guess?
$\longrightarrow$ Yes! There is a feasible dual width assignment of value 7:

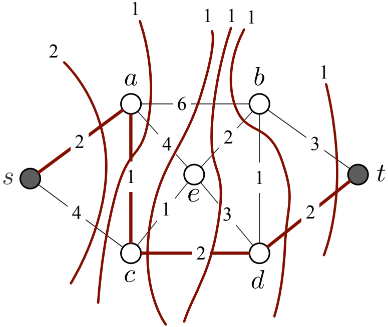
$$
\begin{aligned}
y_{\{s\}} &= 2 \\
y_{\{s,a\}} &= 1 \\
y_{\{s,a,c\}} &= 1 \\
y_{\{s,a,c,e\}} &= 1 \\
y_{\{s,a,c,d,e\}} &= 1 \\
y_{\{s,a,b,c,d,e\}} &= 1
\end{aligned}
$$



Question

(A) In an instance with a shortest path, can we always find feasible widths to prove optimality?

(B) If so, how do we find a path and these widths?



We will answer (A) affirmatively, and provide an efficient algorithm for (B) shortly.

Recap

- A shortest path instance is given by a graph $G = (V, E)$ and non-negative lengths c_e for all $e \in E$.
- A width assignment $y_U \geq 0$ for all s,t-cuts $\delta(U)$ is feasible if

$$\sum (y_U \,:\, e \in \delta(U)) \leq c_e$$

  for all $e \in E$.
- If y is a feasible width assignment and P an s,t-path, then

$$c(P) \geq \sum y_U$$