I begin lecture by recalling the conditions that a Nash flow must satisfy--that the cost of every s-t path with positive flow must be the same, and the condition that the optimal solution must satisfy:

\[
\text{it minimizes } \sum_{e} f_e c(e), \quad (*)
\]

given that \(f\) is a flow.

Assuming that all the functions \(x(e)\) are convex, we see that the objective function (*) has the property that every local minimum is a global minimum.

Let's see what a local minimum looks like. Recall that if the set of s-t paths is \(P_1, \ldots, P_k\), and an \(x_i\) fraction of the flow travels path \(P_i\), then the cost is

\[
C(x_1, \ldots, x_k) = \sum_i x_i C(P_i) = \sum_i x_i \sum_{e \in P_i} c(e) \quad (*)
\]

(\text{where } f_e = \sum_{i} x_i)
\[ = \sum_{e} f(e) \cdot c(e) \]

Let's see what happens if we shift an infinitesimal amount of flow from one path to another, say moving \( \alpha \) from \( P_1 \) to \( P_2 \), where \( \alpha > 0 \).

The increase in \((\ast)\) will be

\[ \epsilon \left[ \frac{\partial C(\alpha_1, \ldots, \alpha_k)}{\partial \alpha_2} - \frac{\partial C(\alpha_1, \ldots, \alpha_k)}{\partial \alpha_1} \right] \]

as we are at a local minimum, this is \( \leq 0 \), which says that

\[ \alpha_1 > 0 \implies \frac{\partial C(\alpha_1, \ldots, \alpha_k)}{\partial \alpha_2} \leq \frac{\partial C(\alpha_1, \ldots, \alpha_k)}{\partial \alpha_1} \]

Let's see what these terms are.

Note that \( \frac{\partial f(e)}{\partial \alpha_1} = \begin{cases} 1 & \text{if } e \in P_1 \\ 0 & \text{o.w.} \end{cases} \)

So,

\[ \frac{\partial C(\alpha_1, \ldots, \alpha_k)}{\partial \alpha_1} = \sum_{e} \frac{\partial}{\partial \alpha_1} f(e) \cdot c(e) \]

\[ = \sum_{e} \frac{\partial}{\partial \alpha_1} f(e) \cdot c(e) \cdot c(e) \]
= \sum_{e \in \mathcal{P}_i} \frac{\partial}{\partial x_i} t \left( x_i \right) f_e t \left( x_i \right)

= \sum_{e \in \mathcal{P}_i} c_e \left( t \left( x_i \right) \right) + t \left( x_i \right) c'_e \left( t \left( x_i \right) \right)

= \sum_{e \in \mathcal{P}_i} d_e \left( t \left( x_i \right) \right),

where we define $d_e \left( x \right) = c_e \left( x \right) + x c'_e \left( x \right)$.

So, the optimum satisfies:

\[ \forall i \text{ s.t. } d_i > 0 \text{ and all } j, \]

\[ \sum_{e \in \mathcal{P}_i} d_e \left( t \left( x \right) \right) \leq \sum_{e \in \mathcal{P}_j} d_e \left( t \left( x \right) \right) \quad (02) \]

This is just like the conditions for Nash equilibrium, but with the cost function $d_e$!
Let's now apply this analysis to see how high
\[
\frac{\text{Nash}(G)}{\text{opt}(G)}
\]
can be on a 2-link network with linear cost functions.

I assert, and will prove later, that it suffices to consider the case where one cost function has the form \( ax \), and the other is a constant \( b \).

\[
\begin{array}{c}
S \\
\rightarrow \\
& b \\
\leftarrow \\
T
\end{array}
\]

As multiplying costs by a constant has no effect on \( \frac{\text{Nash}}{\text{opt}} \), we can assume \( a = 1 \).

Also, this tells us that the analysis will be the same regardless of how much we flow, so let's assume the flow is 1.
First, let's consider the case $b \leq 1$. In this case, the Nash flow sets $x$ so that $x=b$, and the cost of the flow is $b$.

For $\text{opt}$, compute $d_{e_1}(x) = 2x$ and $d_{e_2}(x) = b$, so it sends $x$ over the top link satisfying

$$2x = b \implies x = \frac{b}{2}$$

The cost for $\text{opt}$ is then

$$\frac{b}{2} \left(\frac{b}{2}\right) + \left(1-\frac{b}{2}\right)b = b - \frac{b^2}{4} = b \left(\frac{1}{4}\right)$$

So, $\frac{\text{Nash}}{\text{opt}} = \frac{b}{b\left(\frac{1}{4}\right)} = \frac{4}{1-b} \leq \frac{4}{3},$ for $b \leq 1$

For $b > 1$, Nash will send all its flow over the top link, at a cost of $1$. As we increase $b$ from 1, the cost of $\text{opt}$ can only increase. So
\[
\frac{\text{Nash}(b > 1)}{\text{opt}(b > 1)} \leq \frac{\text{Nash}(b = 1)}{\text{opt}(b = 1)} = \frac{4}{3}
\]
Bounding the Price of Anarchy.

We will now proceed to analyze networks with linear cost functions.

The outline of our analysis is as follows:
We will let $f$ be the Nash flow in $G$.
We will then add edges to $G$ to get a network $\hat{G}$.
$\hat{G}$ will be based on $G$.
We will show that $f$ is a Nash flow in $\hat{G}$, and we will use $f$ to construct an optimal flow in $\hat{G}$, called $g^*$.

We will then show

$$\text{Nash}(\hat{G}) \leq \frac{4}{3} \text{Opt}(\hat{G})$$

we will also have $\text{Nash}(G) = \text{Nash}(\hat{G})$
and $\text{Opt}(\hat{G}) = \text{Opt}(G)$, which together prove

$$\text{Nash}(G) \leq \frac{4}{3} \text{Opt}(G)$$
First, note that it suffices to consider cost functions of the form $ax + b$, as we can replace $ax + b$ by $ax$. 

We build $\widehat{G}$ by creating, for every edge $e$ in $G$, a parallel edge $\widehat{e}$ of constant cost $c(\widehat{e})$.

**Claim 1.** The flow $f$ is still a Nash flow in $\widehat{G}$.

**Claim 2.** The flow $g^* = \frac{f_e}{2}$ and $g^*_\widehat{e} = \frac{f_{\widehat{e}}}{2}$ is an optimal flow in $\widehat{G}$. 

---

462 Page 8
Proof: Note that
\[ d_e(x) = 2ae \cdot x + be \]
so \[ d_e(\hat{g}^*) = a\hat{e} + b\hat{e} = c(e\hat{f}e) \]
and \[ d_e(x) = c(e\hat{f}e), \] so
\[ d_e(\hat{g}^*) = c(e\hat{f}e). \]

So, under cost function \( d \), all \( s-t \) paths in \( \hat{G} \) have the same cost under flow \( \hat{g}^* \), which implies \( g^* \) is optimal for \( \hat{G} \).

Now, let's compute the cost of \( f \), divided by the cost of \( g^* \):

\[
\frac{\text{Nash}(G)}{\text{Opt}(\hat{G})} = \frac{\sum_{e} f_e \cdot (c(e\hat{f}e))}{\sum_{e} \left( \frac{f_e}{2} c\left(\frac{f_e}{2}\right) + \frac{f_e}{2} c\left(\frac{f_e}{2}\right) \right)}
\]

\[
\leq \max_{e} \frac{f_e \cdot c(e\hat{f}e)}{\frac{f_e}{2} c\left(\frac{f_e}{2}\right) + \frac{f_e}{2} c\left(\frac{f_e}{2}\right)}
\]
\[
\max_e \frac{c(e, \vec{e})}{\frac{1}{2} c(e, \vec{e}) + \frac{1}{2} c(e, \vec{e})}
\]

Now, if \(c(e)\) is a constant, this is 1. Whereas if \(c(e) = \alpha x\), it is given as

\[
\frac{\alpha e}{\frac{1}{4} \alpha e + \frac{1}{2} \alpha e} = \frac{1}{\frac{3}{4}} = \frac{4}{3}.
\]

Note that this proof technique can in general be used to reduce to the case of 2-edge graphs.