Here we consider multi-hop network model. The key point of this lecture is that an appropriate max-weight scheduling algorithm achieves max'ed throughput.

Model.

There are n queues in the network. The network is discrete time. Each queue has dedicated arrival process: let $A_i(t)$ be net arrival to queue $i$ up to time $t$.

We assume that:

$A_i(0) = 0$, $A_i(t) - A_i(t^-) \in \{0, 1\}$, for all $t \geq 1$, for all $i$.

Bernoulli arrival process: $P(A_i(t) - A_i(t^-) = 1) = \lambda_i$.

Departure from $i$th queue either leaves network or enters another queue say $j$. This is represented as follows:

$$R_{ij} = \begin{cases} 1 & \text{if } i \rightarrow j \\ 0 & \text{o.w.} \end{cases}$$
Let \( D_j(z) = \begin{cases} 1 & \text{if departure from } j\text{th queue} \\ 0 & \text{otherwise} \end{cases} \)

\[
A_j(z) = A_j(z) - A_j(z-1) + \sum_{i} R_{ij} D_i(z)
\]

Here, we have assumed the following notation:

\[
\begin{aligned}
&Q(z) \quad A(z) \quad A(z+1) \\
&S(z) \quad D(z)
\end{aligned}
\]

As before, schedule is constrained to be in \( S \subset \{0,1,3\}^N \). That is, \( S(z) \in S \) for all \( z \).

\[
Q(z+1) = Q(z) + A(z) - S(z) \cdot \{Q(z) > 0\}
\]

\[
D(z) = S(z) \cdot \{Q(z) > 0\}
\]

Define, \( \Delta_i(z) = A_i(z) - D_i(z) \).

Then, \( Q(z+1) = Q(z) + \Delta(z) \).
Also,

\[
\mathbb{E} \left[ \Delta_i(2) \mid \mathcal{F}_t \right] = \mathbb{E} \left[ \mathcal{A}_i(2) \mid \mathcal{F}_t \right] - D_i(2)
\]

\[
= \lambda_i + \sum_j R_{ji} D_j(t) - D_i(t).
\]

Before proceeding further, define

\[
\lambda_{\text{eff}} = \lambda + R \lambda_{\text{eff}}
\]

That is, \( \lambda_{\text{eff}} = (I - R)^{-1} \lambda \), where we assume that \((I - R)^\dagger\) exists.

We assume that \( \lambda \) is s.t.

\[
\lambda_{\text{eff}} = (I - R)^{-1} \lambda \in \text{co} (S)^\circ.
\]

That is,

\[
\lambda_{\text{eff}} = \sum_{\pi \in S} \alpha_n \pi \text{ s.t. } \sum \alpha_n < 1 ; \alpha_n > 0.
\]
Now, we will derive appropriate algorithm using Quadratic Leapunov function as follows:

\[ L(Q(z)) = \sum_{i=1}^{n} Q_i^2(z). \]

\[ L(Q(z_{t+1})) - L(Q(z_t)) = \sum_{i=1}^{n} \left( Q_i^2(z_{t+1}) - Q_i^2(z_t) \right) \]

\[ = \sum_{i=1}^{n} \Delta_i^2(z) \left( 2Q_i(z) + \Delta_i(z) \right) \]

\[ = \sum_{i=1}^{n} \Delta_i^2(z) + 2 \sum_{i=1}^{n} Q_i(z) \cdot \Delta_i(z). \]

Now, \[ \sum_{i=1}^{n} \Delta_i^2(z) \leq O(n^2) \] since at most \[ n \cdot \text{arrivals} \text{ and } n \cdot \text{departures} \text{ with probability } 1. \]

Now, we consider the average drift in the second term,
\[ \sum_{i=1}^{n} \mathbb{E} \left[ D_i(z) \cdot Q_i(z) \mid \mathcal{F}_z \right] = \sum_{i=1}^{n} Q_i(z) \cdot \mathbb{E} \left[ A_i(z) \mid \mathcal{F}_z \right] \]

\[ = \sum_{i=1}^{n} Q_i(z) \cdot \mathbb{E} \left[ A_i(z) - D_i(z) \mid \mathcal{F}_z \right] \]

\[ = \sum_{i=1}^{n} Q_i(z) \cdot \left[ A_i(z) - D_i(z) + \sum_j R_{ij} D_j(z) - D_i(z) \right] \]

\[ = \sum_{i=1}^{n} Q_i(z) \left[ \lambda_i + \sum_j R_{ij} D_j(z) - D_i(z) \right] \]

\[ = \sum_{i=1}^{n} Q_i(z) \cdot \lambda_i + \sum_{i=1}^{n} Q_i(z) \left[ \sum_j R_{ij} D_j(z) - D_i(z) \right] \]

Let, \( Q_{\Pi(z)} = Q_j \) if \( R_{ij} = 1 \)

\[ = 0 \quad \text{o.w.} \]

Then,

\[ = \langle Q(z), \lambda \rangle + \sum_{i=1}^{n} D_i(z) \left[ Q_{\Pi(z)}(z) - Q_i(z) \right] \]

\[ = \langle Q(z), \lambda \rangle + \sum_{i=1}^{n} S_i(z) \cdot \mathbb{1}_{\{Q_i(z) > 0\}} \left[ Q_{\Pi(z)}(z) - Q_i(z) \right] \]
\[
\langle Q(\omega), \lambda \rangle - \sum_{i=1}^{n} S_{i}(\omega) \mathbb{1}_{\{Q_{i}(\omega) > 0\}} (Q_{i}(\omega) - Q_{\pi^*(\omega)}(\omega))
\]

Now: for \(x \in \mathbb{N}, \ y \in \mathbb{N}\):

\[
\mathbb{1}_{\{x \geq 2\}} (x-y) = \begin{cases} (x-y) & \text{if } x > 0 \\ 0 & \text{if } x = 0 \end{cases}
\]

\[
(x-y) = -y \text{ if } x = 0.
\]

\[
\leq 0 = \mathbb{1}_{\{x \geq 0\}} (x-y) \text{ if } x = 0
\]

Thus,

\[
\mathbb{1}_{\{x \geq 0\}} (x-y) \geq (x-y), \text{ for } x, y \in \mathbb{N}.
\]

Therefore

\[
- \sum_{i=1}^{n} S_{i}(\omega) \mathbb{1}_{\{Q_{i}(\omega) > 0\}} (Q_{i}(\omega) - Q_{\pi^*(\omega)}(\omega))
\]

\[
\leq - \sum_{i=1}^{n} S_{i}(\omega) (Q_{i}(\omega) - Q_{\pi^*(\omega)}(\omega)).
\]

\[
= - \langle S(\omega), (I-R^T)Q(\omega) \rangle
\]
Since \( \lambda \) is the solution of \( \sum_{\pi \in S} \alpha_\pi \pi ; \sum_{\pi \in S} \alpha_\pi < 1 ; \alpha_\pi > 0 ; \)

and \( \lambda = (I-R) \left( \sum_{\pi \in S} \alpha_\pi \pi \right) \)

Putting above together,

\[
\sum_{i=1}^{n} \mathbb{E} \left[ D_i(t) \cdot Q_i(t) \left| \mathcal{F}_t \right. \right]
\]

\[
= \left< Q(t), \sum_{\pi \in S} (I-R) \pi - \alpha_\pi \right> - \left< S(t), (I-R^+) Q(t) \right>
\]

\[
= \sum_{\pi \in S} \left< \pi, (I-R) Q(t) \right> - \left< S(t), (I-R^+) Q(t) \right>
\]

\[
= \sum_{\pi \in S} \left< \pi, (I-R^+) Q(t) \right> - \left< S(t), (I-R^+) Q(t) \right>
\]

ALGORITHM:

\[
S(t) \in \operatorname{argmax} \left< \pi, (I-R^+) Q(t) \right> \quad \pi \in S
\]
Then, above is

\[ \langle 1 - \sum \alpha_n \rangle \langle s(t), (I-R^T)Q(t) \rangle \]

The above algorithm provides -ve drift but only if the weight is non-neg. Therefore, we change the above algorithm as follows:

**Algorithm**

\[ S(t) \in \operatorname{argmax} \left\{ \langle T, \left[ (I-R^T)Q(t) \right]^+ \rangle \mid \pi \in \$ \right\} \]

Note that in the above Lyapunov analysis by replacing \( (I-R^T)Q(t) \)
by \( \left[ (I-R^T)Q(t) \right]^+ \), the following happens:
\[ \langle \lambda, (\mathbf{I} - R^T) \mathbf{Q}(\tau) \rangle \leq \langle \lambda, (\mathbf{I} - R^T) \Theta(\tau) \rangle^T. \]

For \[ \langle \mathbf{s}(\tau), (\mathbf{I} - R^T) \mathbf{Q}(\tau) \rangle, \]
we change the algorithm as: \[ s_i(\tau) = 0 \text{ if } q_i(\tau) - q_{\pi(i)}(\tau) < 0. \]

Therefore,
\[ \langle \mathbf{s}(\tau), (\mathbf{I} - R^T) \mathbf{q}(\tau) \rangle = \langle \mathbf{s}(\tau), (\mathbf{I} - R^T) \Theta(\tau) \rangle^T. \]

This will complete the proof following the Foster-Lyapunov criterion of the fact that the algorithm Max-Weight with weight of node \( i \) as \( (q_i(\tau) - q_{\pi(i)}(\tau))^+ \) with fact that \( s_i(\tau) = 0 \) when \( q_i(\tau) < q_{\pi(i)}(\tau). \)