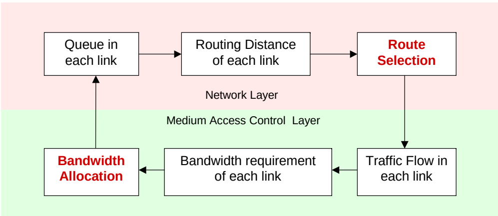
Joint Scheduling and Routing Algorithm for Ad-Hoc Wireless Networks

Yun Li and Anthony Ephremides

Motivation

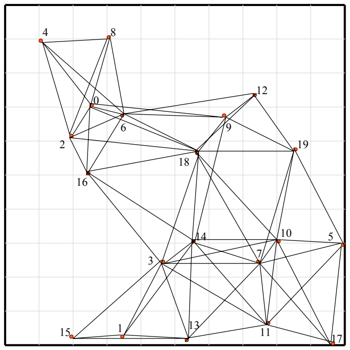
- Strong coupling among the traditional layers of the OSI architecture in ad hoc wireless networks.
- For example, network layer and MAC layer.
- Cross-layer design to improve performance.



Network Model

- A wireless ad-hoc network. No fixed infrastructure.
- All nodes share the same frequency band. Time is slotted. Good global time known to all users.
- A separate low data rate channel for network control, exchange of information, scheduling, and routing.
- Each node is supported by an omni-directional antenna. Same waveform for all users, no multiuser detection.
- Each node is equipped with one transmitter and one receiver. They cannot work simultaneously.
- Transmission power can be adjusted, $0 < P_i < P_{\max}$.
- Nodes within $R_{\max}$ is reachable. $P_{\max} R_{\max}^{-4} = \sigma^2 \beta$.

β is the SIR requirement of successful transmission.



Scheduling

Scheduling Rules:

- A node can only be associated with one active link at a time.
- SIR requirements are satisfied.

$$\frac{P_i G_{ij}}{\sigma^2 + \sum_{k \neq i} P_k G_{kj}} \geq \beta, \forall \text{ link(i, j)}.$$

G_{ij} : Attenuation fator of link(i, j).

- The link with the lowest metric has the top priority.

$$D(i,j) = a \cdot \frac{1}{1+Q_i} + b \cdot \frac{\sum_{\substack{\text{neighbors of } i \text{ or } j \\ k \neq i}} Q_k}{1 + \sum_{\substack{\text{neighbors of } i \text{ or } j \\ k \neq i}} Q_k} + c \cdot \left(\frac{\Lambda_i(j)}{\Lambda_{\max}} \right)$$

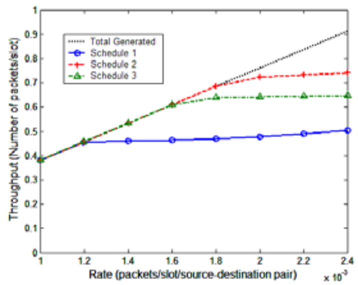
Q_i : Total queue size at node i.
$\Lambda_i(j)$: Average traffic rate at link (i, j).
$\Lambda_{\max}$: Maximum rate among all links.

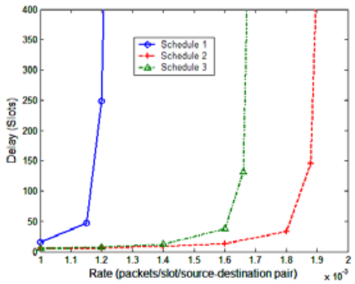
Scheduling Algorithms:

1. Power is preset. Links are added (if SIRs are satisfied) in the order of link metric. Easy for distributed implementation.
2. With iterative power control. Links are added (if SIRs are satisfied) in the order of link metric. Difficult for distributed implementation.
3. Algorithm of T. Elbatt [1]. First find maximal number of links coexist, then run iterative power control. Remove links until SIRs are satisfied. Difficult for distributed implementation.

Simulation Results:

20 nodes,
in 10×10 area,
β=1,
100000 time slots,
a=0.2, b=0.4, c=0.4,
d=0, e=1, (No rerouting)
$Q_{\max}$=10000,
$R_{\max}$=4.





Joint Scheduling and Routing

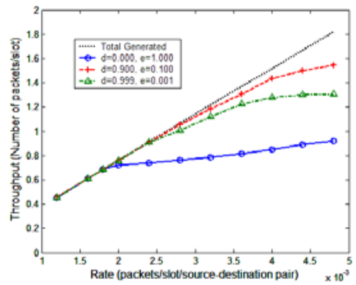
Routing: Bellman-Ford algorithm with routing distance

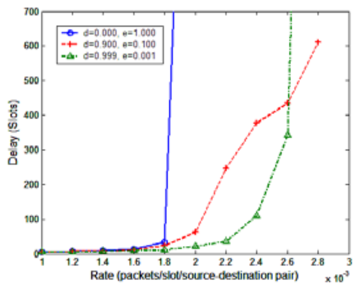
$$D_{ij} = d \cdot \left(\frac{Q_{ij}}{Q_{\max}} \right) + e \cdot \left(\frac{R_{ij}}{R_{\max}} \right)^4 .$$

Q_{ij} : Queue size at link (i, j),
$Q_{\max}$: Buffer size for each link,
R_{ij} : Distance between node i and j,
$R_{\max}$: Maximal reachable distance.

Simulation Result of Periodic Rerouting:

20 nodes,
in 10×10 area,
β=1,
100000 time slots,
a=0.2, b=0.4, c=0.4,
$Q_{\max}$=10000,
$R_{\max}$=4.
Schedule algorithm 2
Reroute every 1000 slots





Summary

- Iterative power control improves the throughput and delays significantly over the preset power scheme.
- Our scheduling algorithm with iterative power control outperforms the algorithm in [1].
- Rerouting periodically improves the network performance significantly, especially for dense network or uneven topology.
- Some modification is needed to achieve distributed implementation of joint scheduling and routing.

Reference: [1]: T. Elbatt, "Transmission power control in wireless ad-hoc networks," Chapter 6, Ph.D. dissertation, University of Maryland, 2000.