Note: HW 3 posted, due next thursday.

EE 597
4/17/12

Routing

Static : ETX, Anypath, Backpressure

Ad Hoc Networks : AODV, DSR, Barrage Relay

ICMN : Epidemic flooding, Spray & Wait

Intermittently Connected Mobile Networks with perfectly predictable encounters (e.g. buses/trains with deterministic schedule)

Given: a list of all pairwise encounters & their times in advance. find the best (shortest delay) route from node A to B.

e.g. 4 nodes: A B C D

encounter	start time	end time
A - B	1	1.1
B - C	2	2.1
A - C	4	4.1
A - B	7	7.2
A - B	7.5	7.8
A - D	8	8.2
C - D	8.2	8.3

← at node A towards B at time 3

Goal: given the list of encounters & times, compute the shortest delay path between any two nodes for a given starting time.

One possible algorithm:

- try every permutation

  e.g. A - B - C - D

  A - C - D - B

- for each find the shortest delay ref. & pick shortest of all

Another approach:

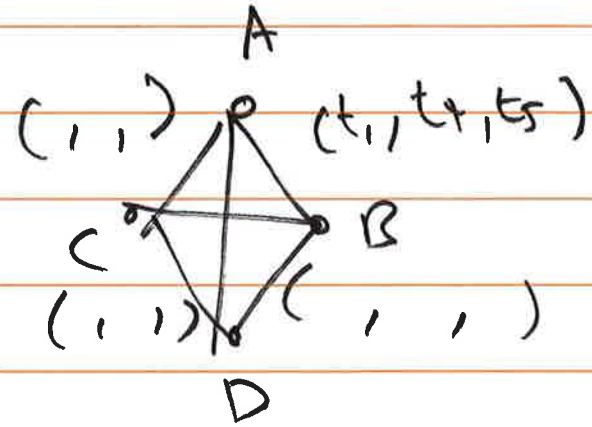
- Use Dynamic Programming

A o

B o

C o

D o



e.g. to go from A to D,
consider all nbrs of A:

$$S_t(A,D) = \min_{K\in N(A)} \; S_t(A,K) + S_0(K,D)$$

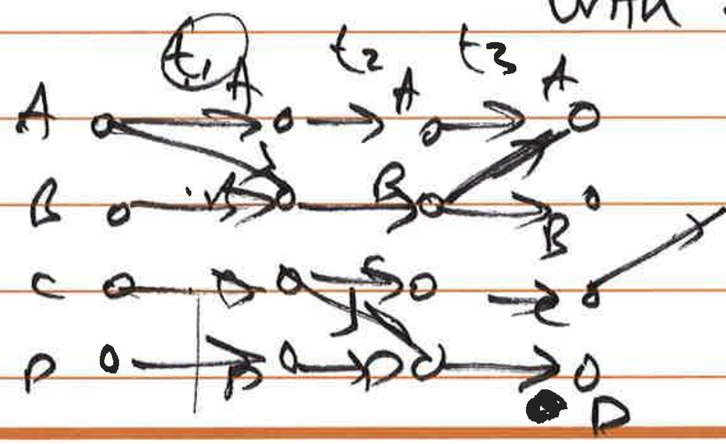
$$t_{AK}^{(u)} + S_{t+t_{AK}^{(u)}}(K,D)$$

Option 3:

- Do a "virtual" flood
- see when the destination gets the packet & trace it back
- this is the shortest delay path.

e.g. D gets the pkt at some time t_0 from C for the first time, it makes a reverse pointer to C, labelled with time t_0.

Option 4:



CS570
Analysis of Algorithm.

Phy MAC Routing Transport ✓

Functions of the Transport Layer

- congestion control & flow control
- End to end reliability
- multiplexing Applications
  - using ports

not fundamentally different in wireless networks.

UDP on a wireless network needs no modification.

Reliability:
- missing packets
- out of order packets

both typically detected using sequence numbers.

by buffering at the transport layer, reordering, & then sending to application.

missing pkts at the <u>end to end</u> perspective can only be dealt with by source retransmissions

=> end to end acks/nacks are needed.

Recall that we also typically use hop by hop ARQ & Ack's. In fact, this

why we use ETX as the routing metric.

Although in theory this should ensure even end to end reliability, in practice number of retransmission attempts are capped at some finite value, resulting in some kind of link loss. (Also due to route breaks).

Congestion Control turns out to be a bit more challenging in Wireless Networks

Two parts to Congestion:

Arriving Traffic
Available Bandwidth

In communication networks, the net effect of congestion is increased end to end delays & packet loss.

Extreme solution: conservatively provision traffic for worst case bandwidth availability.

Congestion control mechanisms must balance two conflicting objectives:

1. avoid congestion
2. maximize utilization

TCP: Transmission Control Protocol (for congestion control)

At its heart are two pieces:

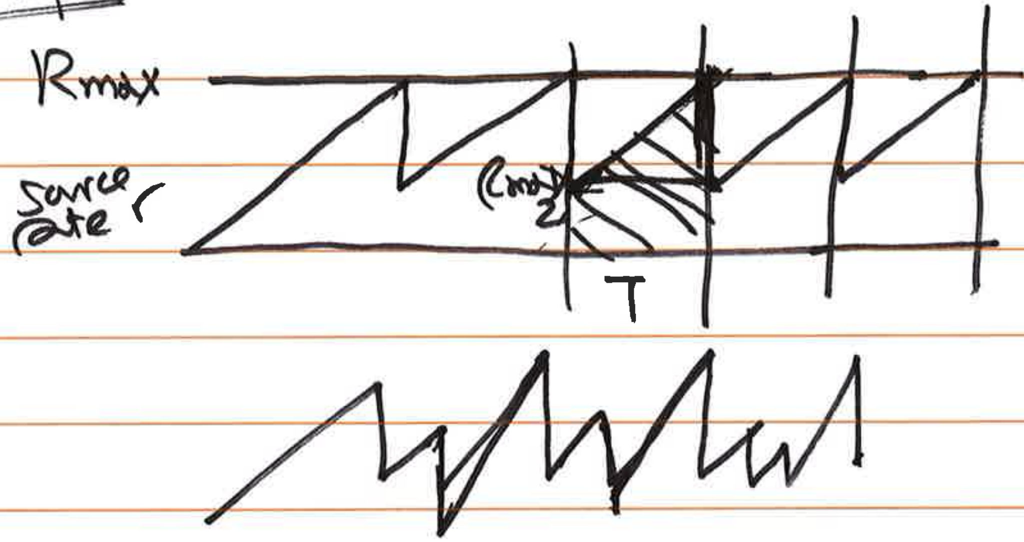
1. Congestion detection
2. Response to congestion

Typically, 1: loss detection via timeouts for Ack reception

2: Additive Increase multiplicative Decrease

Idealized model:

example:



max efficiency is 75%