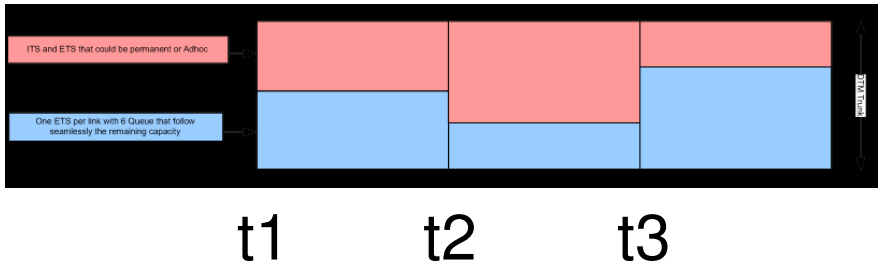
Maximizing DTM Network Utilization Using Dynamic Channel Capacity and Preemption

Jian Liu
jian.liu@netinsight.net
KTH, 2013-06-11

net insight™

Background

- DTM (**D**ynamic synchronous **T**ransfer **M**ode)
  - Circuit switching
  - Time division multiplexing (TDM)
  - End-to-end (multicast) channel with reserved time slots on each link
- Network utilization optimization
  - Prioritize certain channels (video…)
  - Use remaining capacity for lower prioritized channels (IP…)
  - Centralized scheduling or priority-based provisioning

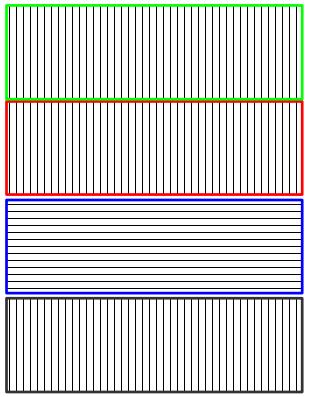


Preemption overview

- What: Enable (optimize) capacity reservation by modifying one or more existing channels
  - Prune branches
  - Change capacity
  - (Moving within same data link or some other data link)
- When: Channel setup and increasing capacity
- How (assumed to be independent):
  - Decision of preemption
  - Candidate selection
  - Capacity releasing/reservation procedure
  - Capacity reversion for preempted channels
- Limitations
  - Distributed routing

Basics

- Each channel has a number of capacity ranges. Each capacity range has a hold priority and a setup priority, hold priority >= setup priority.

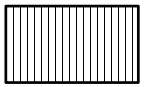


SetupPrio, HoldPrio

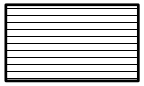
SetupPrio, HoldPrio

SetupPrio, HoldPrio

SetupPrio, HoldPrio



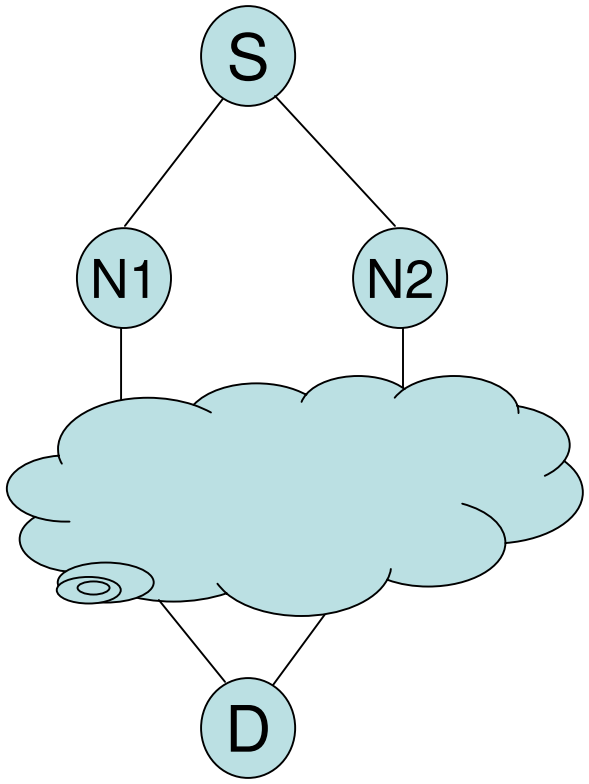
All or nothing (video)



Any value within the range (IP)

Decision on preemption

- Preemption vs alternative next-hop with higher routing cost



Via N2: routing cost 20, preemption required

Via N1: routing cost 10, no preemption needed

Which next-hop node to choose?

Preemption candidates selection

- For capacity increase, optimize:
  - Capacity enabled to be increased, as much as possible, for each branch
  - The priority of the capacity ranges to be preempted, as low as possible
  - Number of channels to be preempted, as small as possible
  - Number of receivers affected by the preemption, as small as possible
  - …
  - Can be deduced to the NP-complete set covering problem.
  - Simple local greedy algorithm not optimal but probably ok.

Preemption candidates selection, cont'd.

- For channel setup, optimize:
  - Number of receivers enabled to be reached
  - Capacity enabled to be reserved, as much as possible, for each receiver
  - The priority of the capacity ranges to be preempted, as low as possible
  - Number of channels to be preempted, as small as possible
  - Number of receivers affected by the preemption, as small as possible
  - …
  - Can be deduced to the NP-complete set covering problem.
  - Simple local greedy algorithm not optimal but probably ok.

Capacity releasing/reservation procedure

- Non source node-aware, request driven
  - Fast
  - Traffic disturbance
  - Possible unnecessary preemption

Preempted channel

Data link

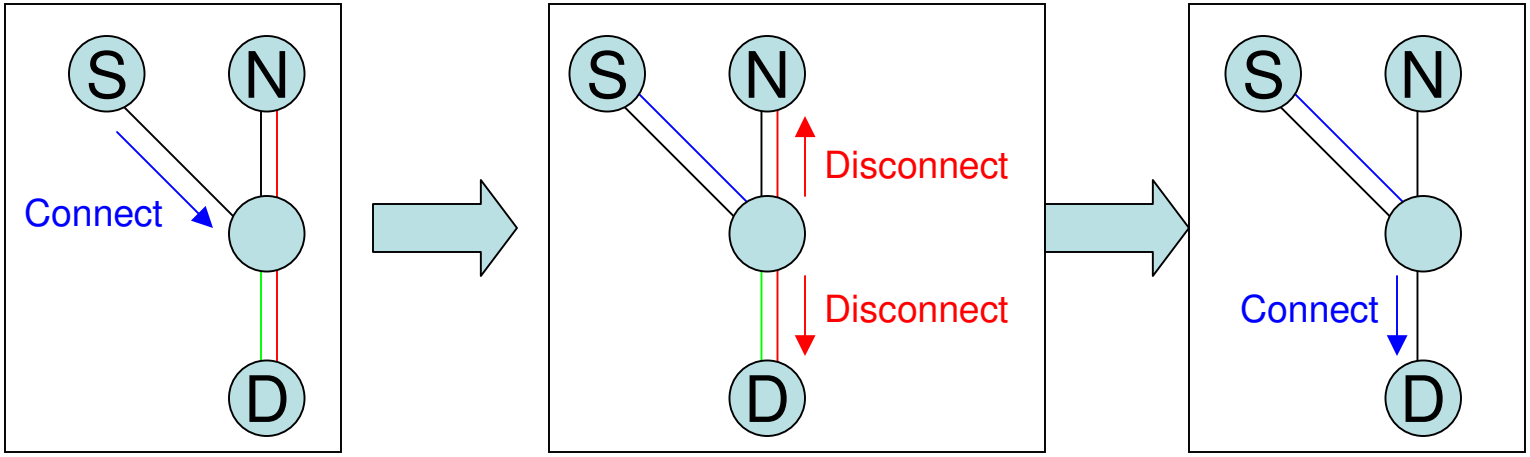
Preempting channel

Bottleneck data link

N = source node of preempted channel

S = source node of preempting channel

D = destination node



Capacity releasing/reservation procedure, cont'd

- Non source node-aware, response driven
  - Traffic disturbance

Preempted channel

Data link

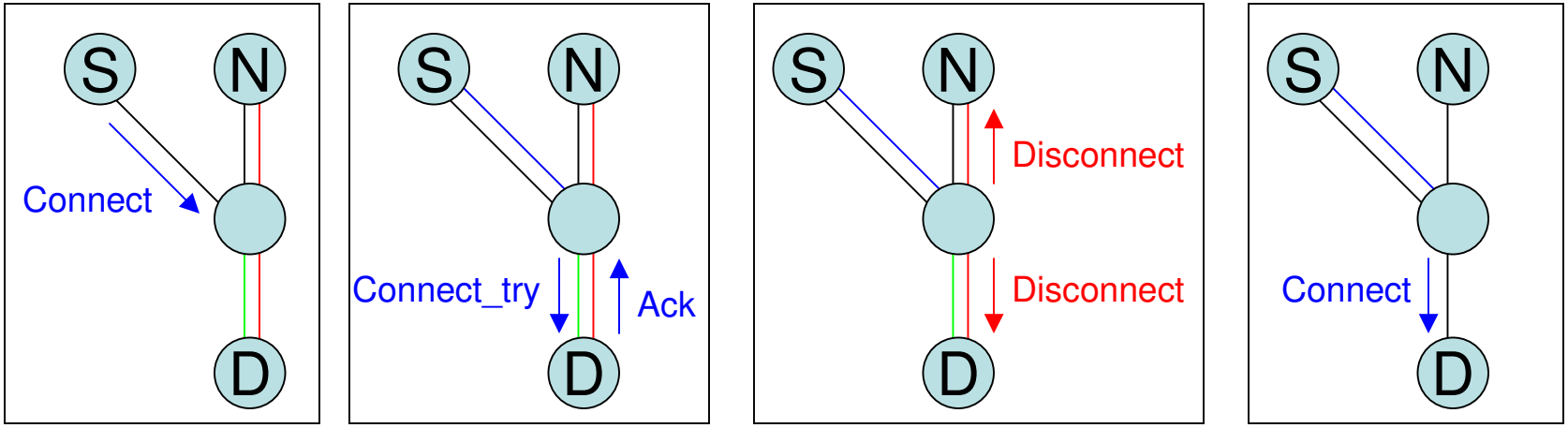
Preempting channel

Bottleneck data link

N = source node of preempted channel

S = source node of preempting channel

D = destination node



Capacity releasing/reservation procedure, cont'd

- Source node-aware, request driven
  - Possible unnecessary preemption

Preempted channel

Data link

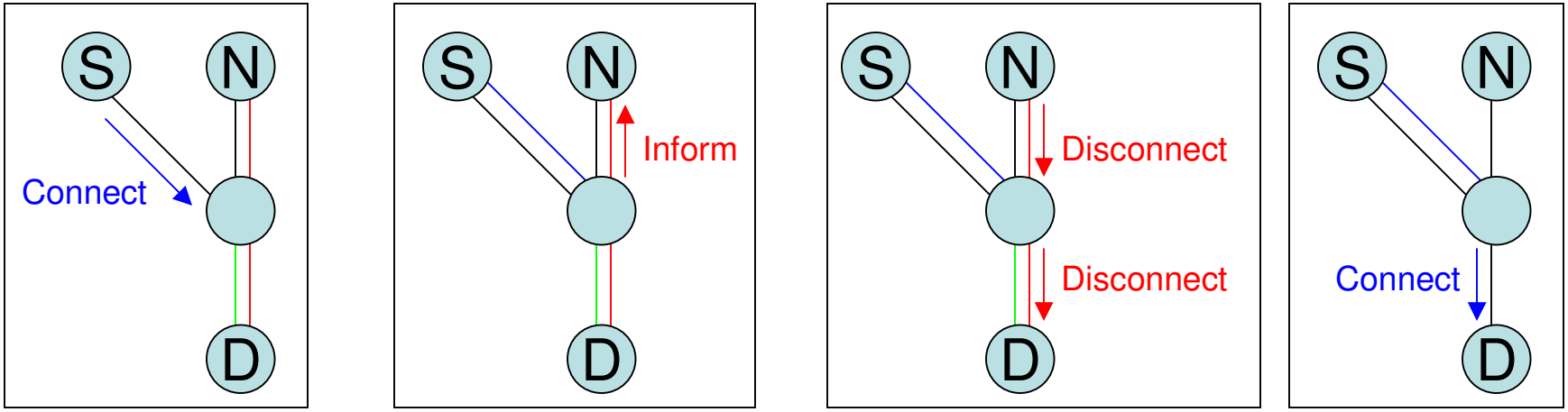
Preempting channel

Bottleneck data link

N = source node of preempted channel

S = source node of preempting channel

D = destination node



Capacity releasing/reservation procedure, cont'd

- Source node-aware, response driven
  - Slow, especially when a channel preempts at many nodes.

Preempted channel

Data link

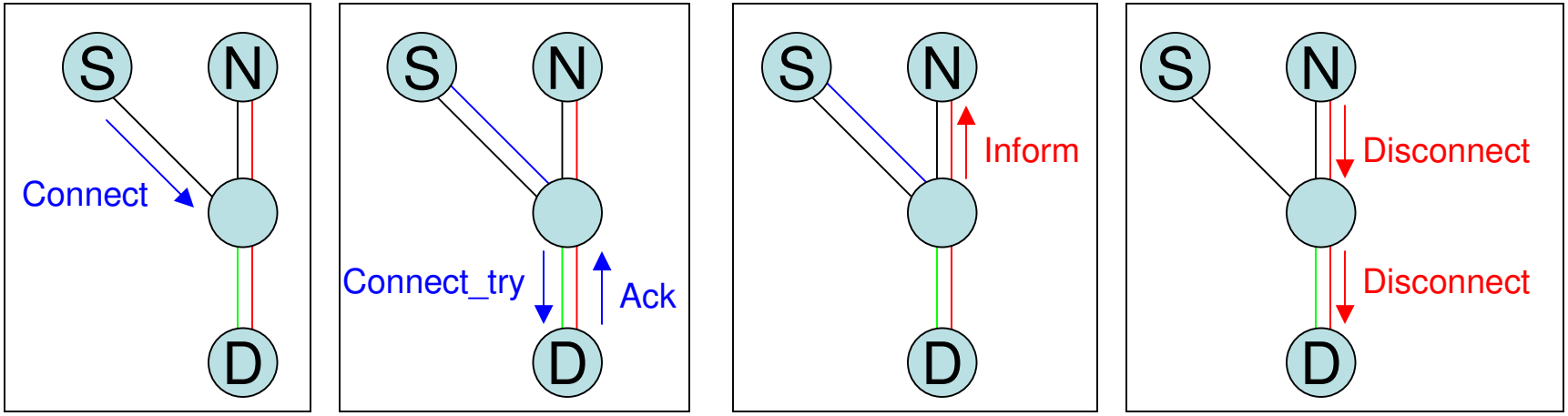
Preempting channel

Bottleneck data link

N = source node of preempted channel

S = source node of preempting channel

D = destination node



Other issues

- One channel preempts at different nodes
  - Message processing detail
- Preemption cascading
  - Preemption loops are avoid by not allowing lower hold priority than setup priority
- Capacity reversion
  - Polling
  - Preempting node informs the source node when capacity available
- Implementation issues
  - Logic complexity
  - Time to market
  - Implementation cost