Network Layer (continued)

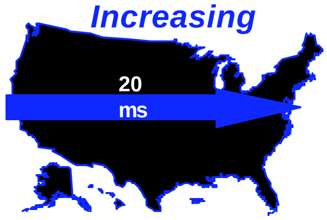
Dr. G. A. Marin

Chapter 5 Homework

- 1,3,5,7,8,9,11,13,14,15,16,19,20,21,22,26,27,28,30
- Due: April 2, 2001
- Test: April 4, 2001
- Final Exam: April 30 1pm - 3:50pm
  - comprehensive

Architecture Drivers:

Data In Flight Drastically Increasing



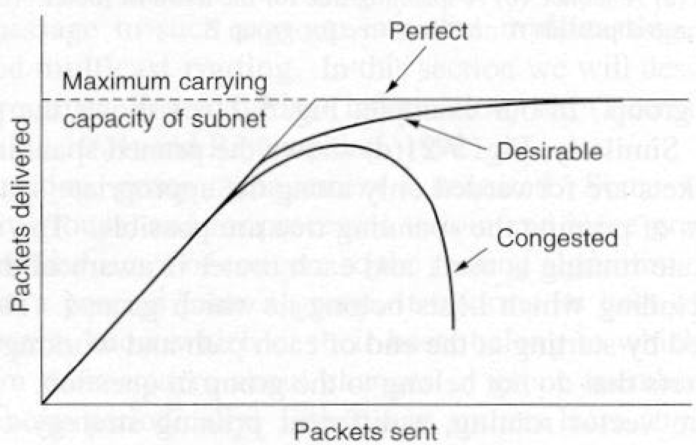
Speed	Bytes/20 ms
1.5 Mbps	3,750
1 Gbps	2,500,000

Preventive congestion control

Rapid dissemination of control information

Keep up with high-speed links

Network Congestion Control



Causes of Congestion

- Queueing in router for particular outbound line
- Lost packets because of insufficient memory
- Too much memory->long delays->timeouts
- Processors that are too slow for demand
- Lines with insufficient capacity (bandwidth)
- NOTE: congestion control is an issue involving performance tuning of the entire subnet.
  - Flow control is between sender and receiver only

Congestion Control Principles

- Open Loop
  - Prevent problems by good design
  - When does net accept new traffic?
  - When does net discard packets?
  - Which packets get prioritized?
- Closed Loop
  - Monitor/detect congestion in network
  - Pass congestion info to where action can be taken (feedback loop)
  - Adjust operation to correct(react) to problem.
- GOOD ARCHITECTURES often use both approaches.

Admission Control

- Open Loop Version
  - Use flowspec parameters to assign resources to every VC admitted to the network. Once all resources have been committed, reject further connections.
- Closed Loop Version
  - When a certain level of congestion has been measured and feedback given to entry nodes of network, reject further connections.

Example Congestion Metrics

- Average queue lengths
- Percent of packets discarded
- Number of packets retransmitted
- Average number of collisions (boadcast)
- Average packet delay (and standard deviation of delay)

Feedback Techniques

- Router detects congestion (from queues or otherwise) and sends information to source.
  - Info can also be collected with probe packets.
  - Which source? How quickly to react?
  - Note that this control traffic increases the load precisely when network already in trouble.
- Bit/bits can be reserved in each header

When congestions occurs...

- Increase resources
  - Bring up additonal (dial-up) lines
  - Bring up additional routers/switches
  - Dedicate more bandwidth on existing lines
  - Split traffic along multiples routes
- Decrease load
  - Don't allow additional users
  - Remove users with lower priority
  - Give some/all users less bandwidth

Policies that Affect Congestion

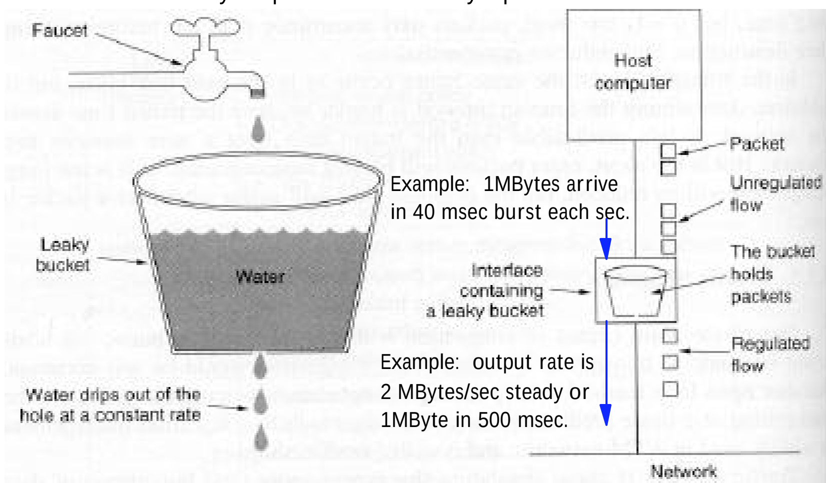
Layer	Policies
Transport	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy • Timeout determination
Network	• Virtual circuits versus datagram inside the subnet • Packet queueing and service policy • Packet discard policy • Routing algorithm • Packet lifetime management
Data link	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy

Traffic Shaping

- Reduces impact of "bursty" traffic on network by regulating average RATE.
- Usually occurs in virtual circuit networks like ATM.
- User and subnet agree on traffic characteristics for particular VC
  - Eg. average burst length, max burst length, average rate
- Subnet monitors VC and discards its traffic when agreement violated (if network is congested).
  - called "traffic policing"

Leaky Bucket Algorithm (1986)

Allows ONLY steady output rate with bursty input.

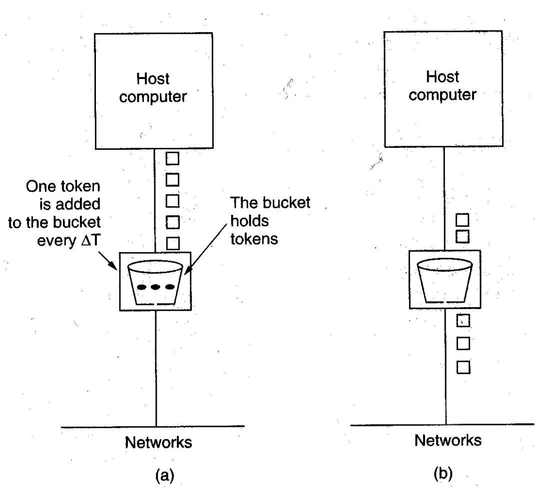


Example: 1MBytes arrive in 40 msec burst each sec.

Example: output rate is 2 MBytes/sec steady or 1MByte in 500 msec.

Token Bucket Algorithm

Allows limited burstiness up to max depth of token bucket.



Note: in both leaky bucket and token bucket versions a variation is possible in which a specific number of bytes is authorized as output per token (or per second) instead of one packet or cell. Useful with variable packet length traffic.

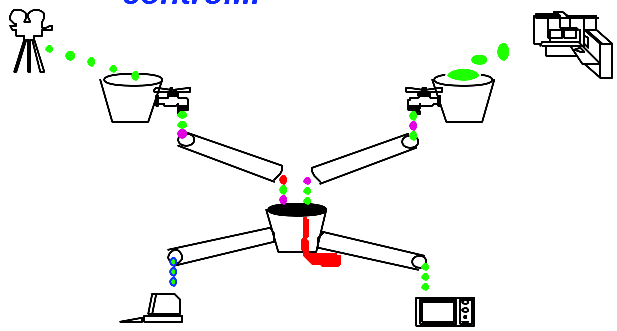
Max Burst Length for Token Bucket Algorithm

- First thought: it is just max data corresponding to max tokens (bucket depth)/output transmission rate.
  - Example: 1Mbyte/(25MByte/sec) = 40 msec
  - WRONG because more tokens arrive as data transmitted.
- Let S=burst length (secs). Let C=bucket capacity (bytes). Let ρ=token arrival rate (bytes/sec). Let M=peak output rate bytes/sec.
  - Output burst contains C+ρS bytes until bucket is empty.
  - Also, whatever S equals, there are MS bytes in the max burst.
  - Implies C+ρS = MS so S = C/(M-ρ).

Example

- A token bucket supports a network of OC3 connections. Max capacity of the bucket is 100 megabits and the bucket is filled at a rate of 10 mbps. What is the max burst length (in bits)?
- Answer:
  - $S=C/(M-\rho)=100x10^6/(155x10^6-10x10^6)$secs
  - S=690 ms
  - Max burst length in bits .69($155x10^6$)=$106.95x10^6$ bits.

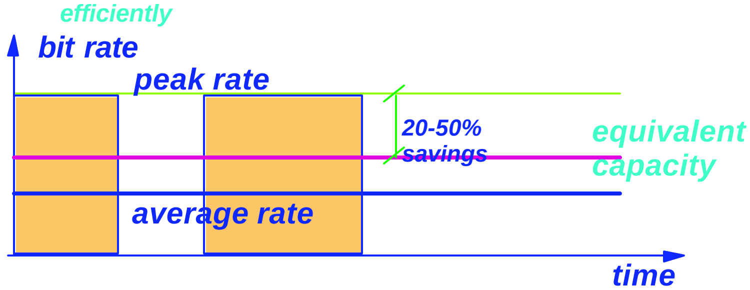
Congestion control...



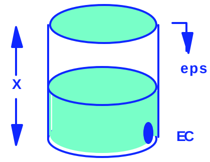
- Choose best available end-to-end route
- Reserve required bandwidth
- Smooth input bursts ("rate-based")
- Adjust bandwidth as needed
- Discard packets if necessary

Bandwidth allocation

- guarantee loss requirements
- use network resources efficiently



R = Peak Rate
m = Mean Rate
b = Mean Burst Length



Equivalent Capacity = the amount of bandwidth to guarantee that CLR is below a given maximum eps. A function of:

- R, m , b
- Buffer size, X
- Maximum CLR, eps

EC = the size of opening to guarantee that overflow ratio is below a given maximum eps.

Fluid model

Flow Specification

- For this to work sender, subnet, receiver must agree to traffic parameters as described in a flow spec.
  - describes characteristics of injected traffic
  - describes the desired quality of service.
  - Source describes its traffic and desired QoS to subnet. Subnet may accept or negotiate or reject. If successful, source then works with receiver to get same agreement.

Partridge FlowSpec (1992)

Characteristics of the Input	Service Desired
Maximum packet size (bytes)	Loss sensitivity (bytes)
Token bucket rate (bytes/sec)	Loss interval (μsec)
Token bucket size (bytes)	Burst loss sensitivity (packets)
Maximum transmission rate (bytes/sec)	Minimum delay noticed (μsec)
	Maximum delay variation (μsec)
	Quality of guarantee

Open problem: most applications do not know what QoS they want and do not know their traffic characteristics.

Choke Packets

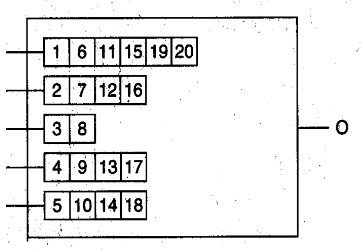
- Each router monitors utilization of each output line.
- Computes $u_{new} = au_{old} + (1\text{-}a)f$, where f is instant utlization and a is a weight between 0 and 1.
- If u is above predetermined threshold (like 0.6), output line is set in a "warning" state.
- When a packet arrives, that needs an output line in a warning state, router sends a choke packet back to the source host giving the destination of this packet. Router tags the packet (sets a bit) and forwards.
  - Tag prevents the next router from sending choke packet.
- Host reduces traffic to that destination by agreed percent.
- Host ignores further choke packets (same destination) for a fixed period.
- Host then can reduce further or increase traffic depending on future choke packets arriving or not arriving.
- Choke packets may instead be triggered by queue length thresholds.

Hop-by-Hop Choke Packets

- Choke packet takes effect in each router as it passes through.
- Moves congestion problem one router at a time until choke packet reaches the source host.
- Requires each router to have sufficient buffers to store incoming traffic that it cannot forward as rapidly as they arrive.

Weighted Fair Queueing

- Problem: hosts respond to choke packets voluntarily. Cooperating host gets penalized unfairly if others don't coorperate.
- Fair Queueing algorithm: at each output line router has separate queue corresponding to each source. Queues are serviced round-robin.
  - Still favors source that sends longer packets.
- Improvement: simulate byte-by-byte round robin.
- WFQ: then add weights according to source priority or bandwidth.



(a)

Packet	Finishing time
C	8
B	16
D	17
E	18
A	20

(b)

Load Shedding (Packet Discard)

- Ultimate hammer for router...throw away certain packets depending on queue thresholds, etc.
- Selective discard:
  - Toss newer packets first if worried about window-based flow control ("wine")
  - Toss older packets first for audio/video ("milk")
- Can do better if applications mark packets according to priority.