

Flow and Congestion Control For High Speed Networks

Wide Area Network

Prof. Taieb Znati
Department Computer Science
Telecommunication Program

Congestion Control in High Speed Networks

- The congestion control problem is more acute in high speed networks supporting QoS, than in “best-effort” networks
- Faster link speeds mean that congestion can happen faster than before
 - e.g., 64 kilobyte buffer
 - @ 64 kbps: 8 seconds
 - @ 10 Mbps: 52 milliseconds
 - @ 1 Gbps: 0.52 milliseconds

Buffering: A Solution?

- Buffering in switches can help alleviate short term or transient congestion problems, but...
- Under sustained overload, buffers will still fill up, and packets will be lost
 - Only defers the congestion problem
- More buffers means more queuing delay
 - Beyond a certain point, more buffering makes the congestion problem worse, because of increased delay and retransmission

Buffering Impact on Delay

- To achieve a worst-case delay of 1 second, buffer requirements increases with link speeds
 - @ 64 kbps → 8 kilobytes buffer
 - @ 10 Mbps → 1.25 Mbytes buffer
 - @ 1 Gbps → 125 Mbytes buffer

Traffic Characteristics

- Traffic is bursty
 - High peak-to-mean ratio, peak rates
 - ◆ Data traffic → 10-to-1, for a peak rate of 1-10 Mbps
 - ◆ Video traffic → 20-to-1, for a peak rate 5-100 Mbps
- Traffic Aggregation, through statistically multiplexing several channels
 - Average behavior is achievable
 - If the aggregate rate of the active traffic flows exceeds the capacity of the channels over a certain period of time, congestion is inevitable

High Speed Network Traffic Control DESIGN ISSUES

Traffic Control Approaches (I)

- Two fundamental approaches to congestion control, reactive and preventive, are possible
- Reactive → feedback-based
 - Attempt to detect congestion, or the onset of congestion, and take action to resolve the problem before things get worse
- Preventive → reservation-based
 - Prevent congestion from ever happening, by reserving resources

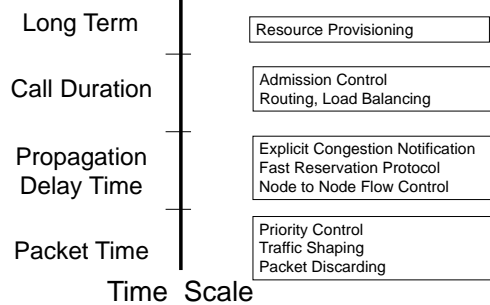
Traffic Control Approaches (II)

- Current Internet approaches to traffic control are mostly based on reactive schemes
 - TCP Slow Start
 - Source Quench
- The large "**Delay*Bandwidth**" is such that most of these approaches are not applicable to high speed networks
- Preventive, reservation-based congestion control strategies are better suited for high-speed networks, with large "**Delay*Bandwidth**" product

Congestion Control in High Speed Network Traffic Levels of Control

- Congestion control can be applied at different levels:
 - Infrastructure Level Control
 - ◆ Network provisioning to meet the **long term** traffic and QoS requirements of a community – Over-engineering?
 - Call Level Control – Also known as **session level** control
 - ◆ Prevent congestion by not allowing new calls or connections into the network unless the network has sufficient capacity to support the new calls
 - A typical approach to call-level congestion control is call admission control
 - Roundtrip Level Control
 - ◆ Enforce congestion control on a propagation delay scale
 - Packet Level Control – Also known as **input rate** control
 - ◆ Control the input rate of traffic sources to prevent, reduce, or control the level of congestion

Traffic Control Mechanisms – Time Scale



Call-Level Control

- At time of call setup (connection establishment), a **connection** (or a flow) requests the resources that it needs for the duration of the call and specifies its QoS requirements
 - Resource include bandwidth, buffers,
 - QoS include upper bounds on packet-loss, delay, jitter, ...
- If resource are available to meet QoS requirements
 - Call accepted
- Else
 - Call rejected

Call-Level Control Strategies

- The objective is to control traffic so that the QoS requirements of **currently accepted** calls and **new** calls are met
 - Accept enough new calls to achieve **high network resources**,
 - ◆ But just enough calls to ensure that the probability of network **congestion** remains **low**
- Call-level Control Strategies
 - Conservative – reservation accounts for worst case traffic scenario
 - Aggressive – reservation only considers average behavior

Call-Level Control Challenges

- Hard to specify resource requirements and QoS parameters precisely
 - QoS requirements may not be known,
 - QoS may be difficult to measure
 - ◆ Congestion can still occur
- Hard to achieve fairness among calls
 - What policy must be in place to accept and reject calls?
 - ◆ Is FIFO good enough?
 - Long access delay, and possibly denial of service

Packet Level Control

- Control the input rate of traffic sources to prevent, reduce, or control the level of congestion
- A wide range of mechanisms can be used
 - Traffic shaping and traffic policing,
 - ◆ Leaky Bucket and Token Bucket
 - Input traffic tagging (coloring) and traffic discarding
 - Packet scheduling disciplines

Achieving Traffic Control in High Speed Networks

- A combination of various flow control mechanisms must be in place to achieve traffic control in high speed networks
 - **Call Admission Control Scheme**
 - ◆ Overly conservative schemes, based on worst case scenario, are resource wasteful.
 - ◆ Overly optimistic schemes may violate QoS guarantees.
 - **Traffic Descriptor** – necessary to specify the input traffic and used to determine the amount of resources to be reserved
 - ◆ A universal descriptor for different types of applications is not like
 - **Traffic Shaping and Policing** – necessary to guarantee that traffic does not deviate from its traffic specification
 - Scheduling Discipline at Intermediate Nodes.
 - ◆ Tradeoff between efficiency, simplicity and capability of supporting delay bounds.

Traffic Descriptors

- Traffic descriptor can be viewed as a behavior envelope
 - It describes the traffic behavior at different levels
 - Exact behavior – very difficult to achieve
 - Typically used to describe worst or average case behavior
- It forms the basis of a traffic contract – Service Level Agreement (SLA)
 - Source agrees not to violate traffic descriptor.
 - Network guarantees the negotiated level of QoS
- Traffic policing mechanism is used to verify that the source adheres to its traffic specification.

Traffic Descriptors Properties

- **Usability**, the source must be able to describe its traffic easily, and the network must be able to perform feasibility test for admission control easily.
- **Verifiability**, the policing mechanism must be able to verify the source adheres to its traffic descriptor.
- **Preservability**, the network node must be able to preserve the traffic characteristics along the paths, if necessary.
- Three traffic descriptors are commonly used.
 - Peak Rate, Average Rate, and Linear Bounded Arrival Process.

Traffic Descriptor – Peak Rate

- Highest rate of traffic generation
 - For network with fixed size packets
 - ◆ Peak rate is the inverse of the closest spacing between the starting times of consecutive packets.
 - For network with variable size packets
 - ◆ Peak rate defines an upper bound on the total number of packets generated over all window intervals of the specified size.

Peak Rate

- Peak rate descriptor is easy to compute and police.
- It is an **extremal**, loose bound.
 - A single outlier can change the descriptor considerably.
- Only useful for sources with smooth traffic

Average Rate

- Objective is to reduce the effect of outliers
 - Transmission rate is averaged over a period of time
- Two parameters are defined
 - t = time window over which rate is measured.
 - a = number of bits to be sent over t .
- Two average rate mechanisms are used :
 - Jumping window
 - Moving window

Jumping Window

- Source claims that over t no more than a bits will be transmitted to the network
 - A new time interval starts immediately after the last one.
- Jumping window is sensitive to the starting time of the first window.

Moving Window

- Source claims that over all windows of size t no more than a bits will be submitted to the network.
 - Time window moves continuously.
- Enforces tighter bounds on spikes in the input traffic.

Linear Bound Arrival Process

- LBAP – constrained source bounds the number of bits it transmits in any interval of length t by linear function of t .
 - Number of bits transmitted over any interval of length $t \leq \rho t + \sigma$, where
 - ◆ ρ is the long term average rate allocated by the network.
 - ◆ σ is the longest burst a source may send, while still obeying the LBAP descriptor.
- A source has an intrinsic long-term average rate ρ , but can sometimes deviate from this rate, as specified by σ .

Policing and Traffic Shaping

- One of the main causes of the congestion is that traffic is often bursty
- To eliminate, or at least reduce, burstiness traffic shaping may be used
 - The objective is to "shape" traffic into a predictable pattern commensurate with the expected SLA traffic behavior
- Combined with traffic policing, traffic shaping is a useful technique to manage congestion

Traffic Shaping Properties

- Traffic Shaping rules should make description of traffic pattern easy.
 - Scheme should support the description of a wide range of behaviors.
- Traffic Shaping should make Traffic Policing easy to implement and enforce.
 - Allow the network to accept or reject traffic based on descriptor.
- Traffic Shaping and Policing Schemes
 - Leaky Bucket
 - Token Bucket

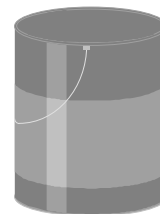
Leaky Bucket

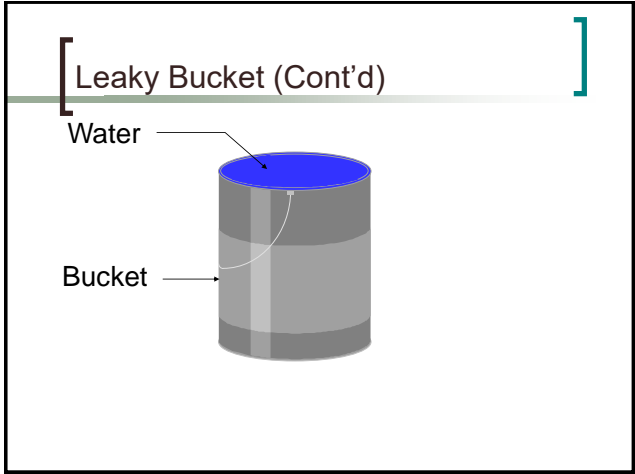
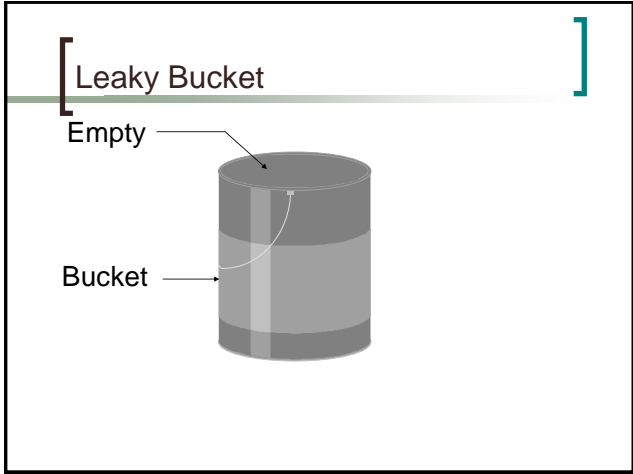
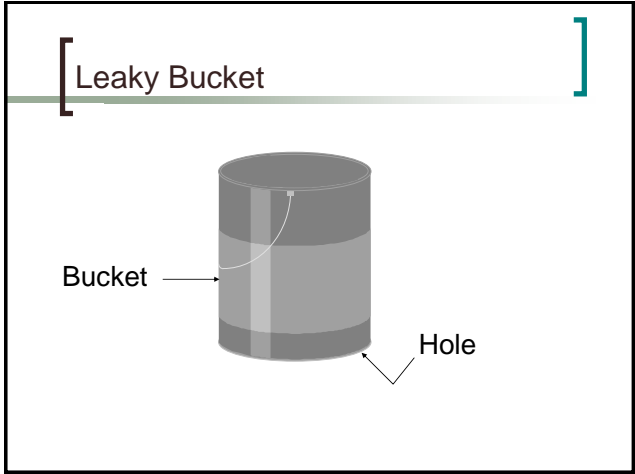
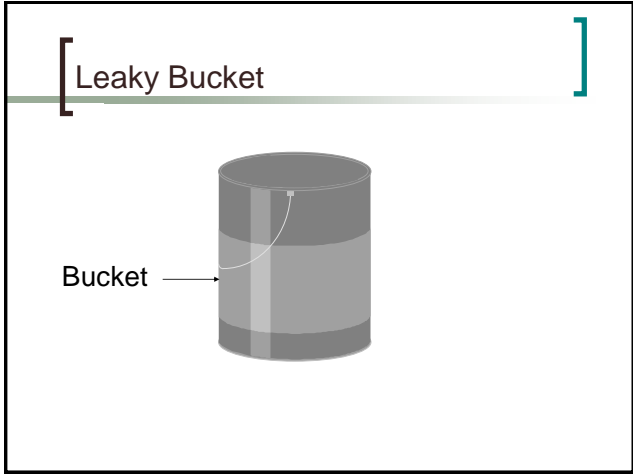
- One of the input rate traffic control mechanisms that has been proposed is the leaky bucket
 - Leaky Bucket as a Traffic Policing Mechanism
 - ◆ As a traffic policing mechanism, leaky bucket checks conformance of a source to its traffic descriptor
 - Leaky Bucket as a Traffic Shaper
 - ◆ As a traffic shaper, leaky bucket "rebuilds" incoming traffic to meet the expected shape according to traffic descriptor

Leaky Bucket

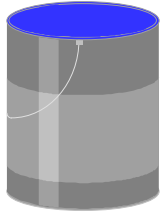
- Leaky Bucket Concept
 - A bucket (pail), with a hole in the bottom, is filled with water
 - ◆ Water drips out the bottom at a constant rate
 - The size of the whole determines the rate at which the water drips
 - The dripping rate determines the rate at which packets enter the network
 - A packet is presented to the network at each drip
- The leaky-bucket provides the basis for flow control schemes to manage network congestion by controlling what gets out of the bucket

Leaky Bucket Illustration

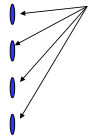




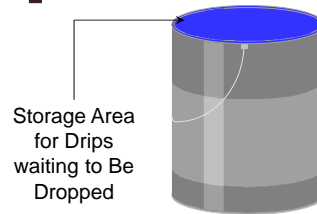
Leaky Bucket (Cont'd)



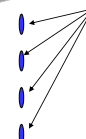
- ❖ Constant Rate Stream of Drips,
 - Periodic
 - Equally Spaced



Leaky Bucket (Cont'd)



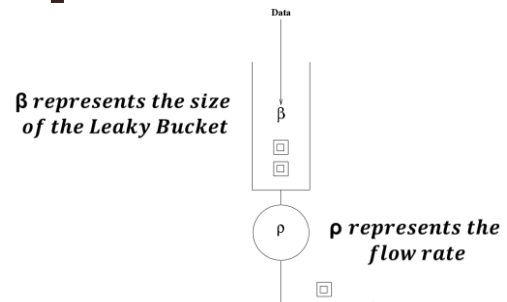
- ❖ Constant Rate Stream of Drips,
 - Periodic
 - Equally Spaced



Isochronous Traffic Shaping Simple Leaky Bucket

- Purpose is to shape bursty traffic into a regular stream of packets.
 - A flow is characterized by a rate ρ
 - A bucket is characterized by a size β
- Rate is enforced by a regulator at the bottom of the bucket.

Simple Leaky Bucket Characterization



Leaky Bucket Effect

- Main effect is to coerce a bursty flow into a flow of equally spaced packets, typically of fixed size.
 - Packets are drained out the bottom of the bucket and sent a rate ρ .
 - ◆ A packet is injected every $1/\rho$ units of time

Leaky Bucket Effect

- The effect of β is to :
 - Bound the amount of delay a packet can incur.
 - Limit the maximum bucket size
 - ◆ Burst bigger than β will be discarded.

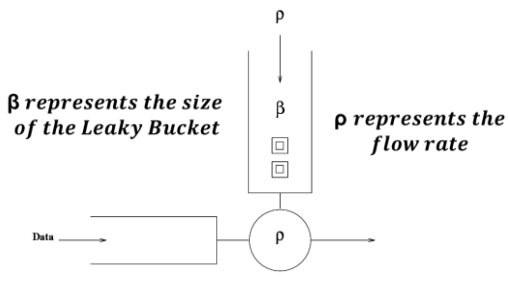
Limitation of Isochronous Schemes

- Traffic shaping is limited to fixed rate data flows
 - Variable rate flows must request data rates equal to their peak rate.
 - ◆ Wasteful
- Isochronous shaping with priority (coloring)
 - Marking may be difficult.

Shaping Bursty Traffic Token Bucket

- Token bucket is an enhanced form of leaky bucket to allow for burstiness
 - Buckets no longer hold flow's data.
 - ◆ Buckets hold tokens
 - Tokens are used to regulate flow's data.
 - ◆ A token is required for the transmission of a unit of data
 - A unit of data can be bit, byte or a fixed size packet

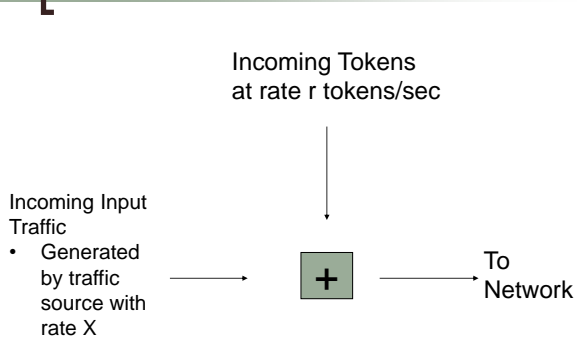
Token Bucket Scheme



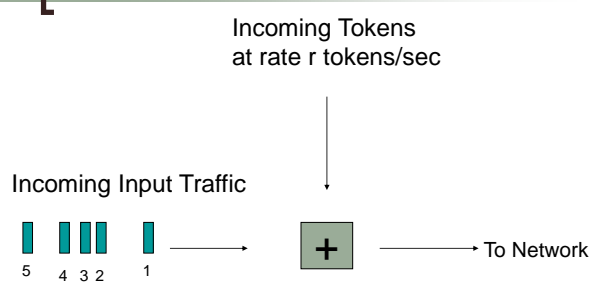
Token Bucket Scheme

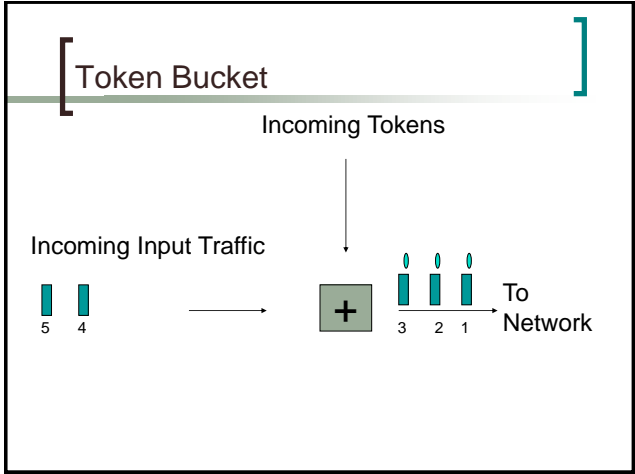
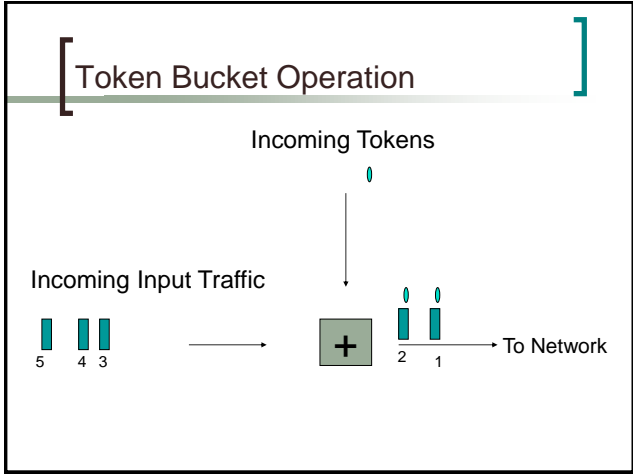
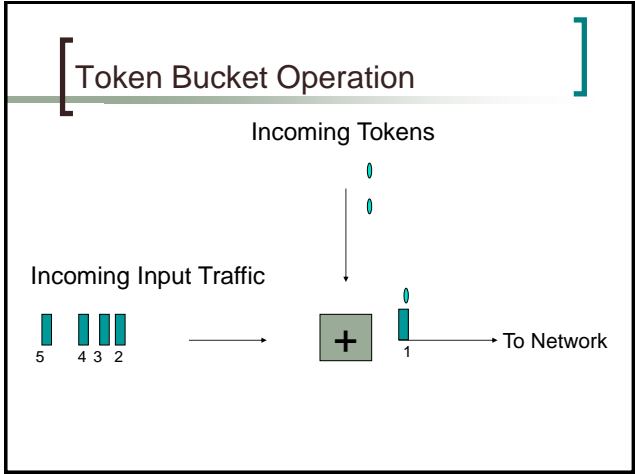
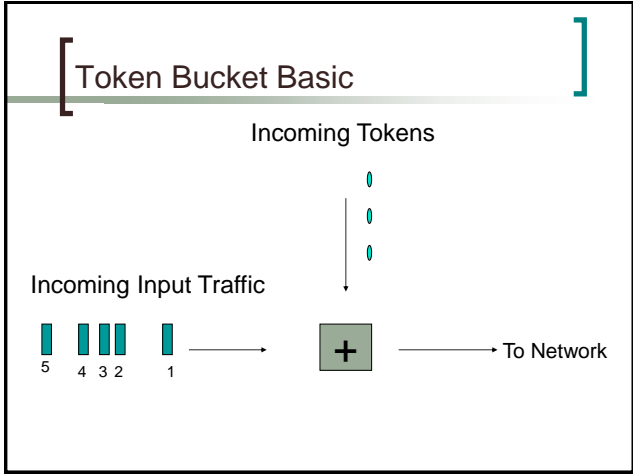
- Token are placed at rate ρ in the bucket.
 - If the bucket fills, newly arriving tokens are discarded.
- To transmit a packet, the regulator removes from the bucket a packet size worth of tokens.

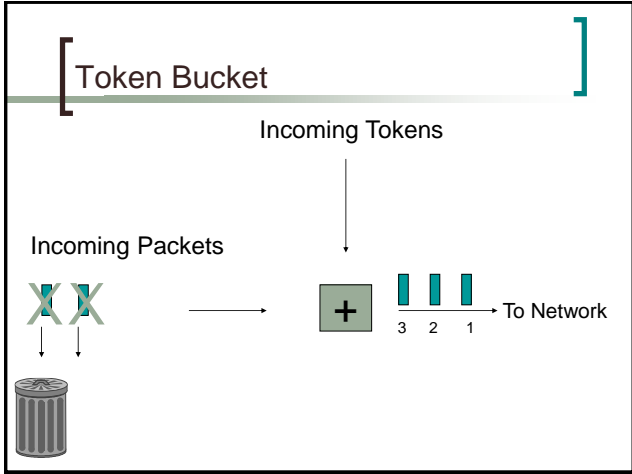
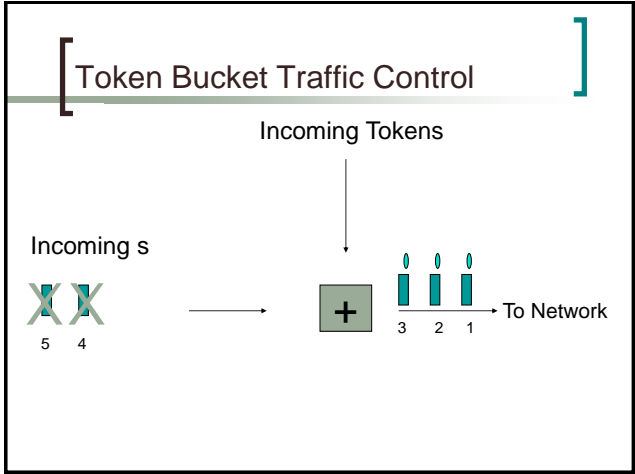
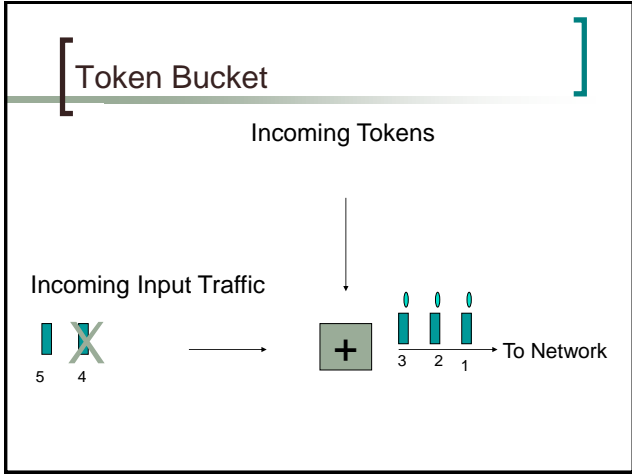
Token Bucket Basic Operation



Token Bucket Basic Operation





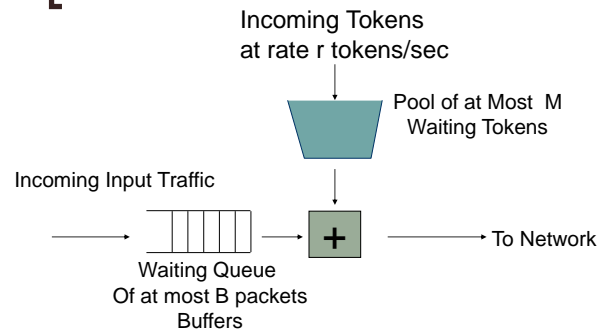


- ### [Token Bucket Traffic Control]
- Input traffic must obtain tokens in order to proceed into the network
 - If no token available, then input traffic is discarded
 - Constrains the rate at which input traffic can enter the network to be the rate negotiated at the time of call setup
 - Shapes traffic, and reduces "burstiness"

Buffered Token Bucket

- Arriving input traffic that finds a token waiting can proceed directly into the network
- Arriving input traffic that finds no token ready must wait in queue for a token
- Input traffic that arrives to a full queue are lost
- Tokens that arrive to a full token pool are simply discarded

Buffered Token Bucket



Buffered Token Bucket Operation

- Incoming input traffic rate: X
- Token rate: r
- If $X > r$, then input traffic must wait in buffer until tokens are available
 - Output traffic is r packets/sec, “nicely” paced
 - ◆ Shape typically conforms to traffic specification
- If $X < r$, then tokens always ready
 - Output traffic rate is X ($< r$)

Buffered Token Bucket

- A station can “store” at most M tokens
 - Station can send at most M packets back to back, if the transfer unit is a packet
 - ◆ Limits the maximum burst size in the network to M packets
- The buffer size, B , can be set to balance the tradeoff between packet loss and packet delay
 - The worst case delay packet is a factor of B and the rate at which tokens are generated

Token Bucket Traffic Control

- The token rate r is set based on the rate declared at the time of call setup
 - The Token Bucket ensures that each source obeys the rate that was specified during the call admission phase
 - ◆ Traffic descriptor
- A single Token Bucket can be used to police the peak rate
 - A measure a burstiness
- A Dual Token Bucker can be used to police both peak rate and average rate
 - Allow burstiness but enforce average rate on the long run

Token Bucket – Unit Impact

- Sending unit can be expressed in bits, bytes, fixed size cells,
- Assuming a sending unit of one byte, to send a packet of size b bytes:
 - If token bucket full, packet is sent and b tokens are removed.
 - If token bucket empty, packet must wait for b tokens.
 - If token partially full ($\leq b$), packet waits for difference.
- The burstiness is controlled at the byte level
 - Up to token-size worth of bytes can be sent back-to-back.

LBAP Regulator Token-bucket

- A token-bucket can be used to regulate LBAP descriptor.
 - It shapes incoming traffic to conform a LBAP specification.
- Regulator collects tokens in a bucket of size σ which fills at a steady rate, ρ .
 - A token allows a source to send a predetermined number of bits, bytes, packets, etc.
 - If bucket fills, excess tokens are discarded.
- Regulator submits a packet only if the bucket has enough tokens.
 - Packet waits if not enough tokens.

Token Bucket and LBAP

- A token bucket limits the size of a transmitted burst to the bucket's depth.
 - Actually, slightly more as tokens may arrive while the bucket's worth of data is being transmitted.
- Over a long term, the rate at which packets depart is limited by the rate at which tokens are added to the bucket.
- Can a minimal LBAP be achieved?

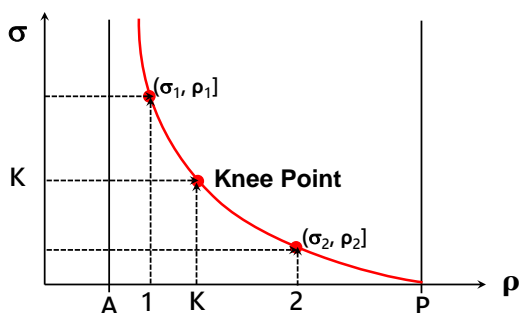
LBAP Parameter Selection

- An LBAP descriptor is said to be minimal if no other descriptor has both a smaller σ and a smaller ρ .
 - Minimal descriptors are likely to be cheaper, if resources are paid for.
- Unfortunately, the minimal LBAP descriptor is not unique.
- Given the size of the data buffer at the regulator and the maximum loss allowed, each of the choice of the token arrival rate has a corresponding minimum burst size so that the loss parameter is met.

LBAP Minimality

- A source with peak rate, P , and average rate, $A \geq \rho$, causes the regulator buffer to grow without bound.
 - Avoiding packet losses requires σ to be infinite.
- If $\rho \geq P$, then there are always tokens available when a packet arrives.
 - σ can be as small as one maximal-sized packet.
- As ρ increases in the range $[A, P]$, the minimum σ needed to meet the loss bounds decreases.
 - Any ρ and its corresponding σ is a minimal bound.

LBAP Minimality



Token Bucket Variations

- There are several different variations of the basic leaky bucket concept described in the literature, such as the virtual leaky bucket, spacer, others
- The schemes differ on how strictly are rates enforced
 - Rather than strictly enforcing rates, schemes allow senders to occasionally exceed their prescribed rate, as long as they mark or tag the excess input traffic

Conclusion

- Traffic control for high speed networks
 - Call level control
 - Input rate traffic control
- Traffic Descriptors
 - Moving Window
 - Jumping Window
 - Linear Bounded Arrival Process
- Traffic Shaping and policing
 - Leaky Bucket
 - Token Bucket
 - Variations
- LBAP Minimality