Metrics, Network Traffic, Performance

Outline

Define Network Performance Metrics
Application Requirements
Traffic Models
Theoretical Prediction of Delay and Loss
Simulation of Networks
Delay

- Propagation delay = D/s
  - D(m) Distance between NEs
  - s propagation speed in media
    - s = 3x10^8 m/s in free space
    - s = (2/3)c = 2x10^8 m/s in fiber
    - s = 0.88c = 2.64x10^8 m/s in coax

- Transmission delay (clocking time) = L/R
  - L = Packet Length (bits)
  - L is random
  - R = Link rate (bits/sec)

- Arrival times of packets are random
- Queueing delay = random
- Processing delay, e.g.,
  - Error check
  - Read destination address
  - Forwarding
  - Special handling
  - Common to assume Processing delay << other delays

Propagation Delay:

- The Speed of Light Limitation

Propagation delay = Distance (m)/Speed of Light m/s

Example: 3000 km fiber link
  - Speed of light in fiber = s = 0.66*(3x10^8 m/s)
  - Propagation delay = 3000x10^3 m / 0.66*(3x10^8 m/s) = 15 ms

- Other Media
  - Speed of light in free space = 1.0*(3x10^8 m/s)
  - Speed of light in coax = 0.88*(3x10^8 m/s)

Effect of clocking time, L/R "putting the bits on the link"

Example: Distance = 3000 km, Data rate = 1 Mb/s, Packet size = 1000 bits

Source    Destination
Packet
  Clocking time = L/C = 1 ms
  Propagation time = 15 ms
Transmission Delay

- transmits packet into access network at transmission rate $R$
- link transmission rate $R$ (b/s), aka link capacity
- Bandwidth (Hz) $\neq R$ (b/s)

Example: $L=1000$ Bytes, $R=10\text{Mb/s}$, clocking delay= 0.8ms

![Diagram of packet transmission delay](image)

Packet transmission delay = time needed to transmit $L$-bit packet into link = $\frac{L}{R}$ (bits/sec)

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Packet delay: four sources

$A$ transmission $\rightarrow$ propagation $\rightarrow$ $B$

$\begin{align*}
thickness{4pt}
d_{\text{nodal}} &= d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}} \\
\end{align*}$

$d_{\text{proc}}$: nodal processing
- check bit errors
- determine output link
- typically < msec
- Often assumed to be zero

$d_{\text{queue}}$: queueing delay
- time waiting at output link for transmission
- depends on congestion level of router

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020
Packet delay: four sources

\[ d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}} \]

- \( d_{\text{trans}} \): transmission delay:
  - \( L \): packet length (bits)
  - \( R \): link transmission rate (bps)
  - \( d_{\text{trans}} = \frac{L}{R} \)
- \( d_{\text{prop}} \): propagation delay:
  - \( D \): length of physical link
  - \( s \): propagation speed
  - \( d_{\text{prop}} = \frac{D}{s} \)

Response Time-Latency

Response time \( T_{R} \): The time to “correctly” transmit a packet from Source to destination. “correctly” implies Response time includes acknowledgments

Examine key components of delay
Delay vs Latency

Latency and network delay both refer to the time it takes for data to travel from one point in a network to another. Latency is the time it takes for a packet of data to travel from its source to its destination and get a response, e.g., response time. Most common metric is the minimum RTT. Delay is sum of processing, queueing, transmission and propagation times.

For more details see: “Internet Measurement: Infrastructure, Traffic and Applications” Mark Crovella, Balachander Krishnamurthy, Wiley, 2006

Delay-Network Classifications

PAN: Personal Area Networks  [BAN: Body Area Network]
- ~3 m or 10ns
DAN.: Desk Area Networks
- ~3 m or ~10ns
LAN: Local Area Networks
- ~3 km or ~10us
MAN: Metropolitan Area Networks
- ~300 km or ~1ms
NAN: National Area Networks
- ~3000 Km or ~10ms

GAN: Global Area Networks
- ~10,000 Km or ~30ms
- NANs and GANs are typically called WANs Wide Area Networks

Terrestrial Networks

Satellite Networks
- Geosynchronous Earth Orbit (GEO) 35,800 km ~120 ms to satellite
  - One hop ~240 ms
  - RTT ~480 ms
- Low Earth Orbit (LEO) 550 km ~1.8 ms
  - One hop ~3.6 ms
  - RTT ~7.2 ms

Interplanetary Networks
Delay-Bandwidth Product

Delay-Bandwidth Product
- One way propagation delay = $\tau$ sec
- Round-trip-time (RTT) = $2\tau$ sec
- Link rate = $R$ b/s
- Delay-Bandwidth Product (bits) = $2\times R$ bits (# bits in RTT)

Number of packets in Delay-Bandwidth Product
- $L$ Packet length in bits/packet
- Number of packets in Delay-Bandwidth Product = Number of packets in round trip time (RTT) = $(\text{Delay-Bandwidth Product-bits})/(L\text{-bits/packet})$

Example:
- $D = 2000$ km
- $c = 2\times 10^8$ m/s (fiber)
- $\tau = 10$ ms
- $L = 1000$ Bytes
- $R = 10$ Gb/s
- Delay-Bandwidth Product = $200$ Mb
- # packets in DBP = 25,000
- The transmission line is “storing” 25,000 packets

Packet loss

No space in Queue to store incoming packet: Cause-network congestions
Corrupted packets: Cause-bit errors
Networks may try to recover lost packets
Or applications cope with lost packets, e.g., packet voice and video often accept a lost packet as an impairment.
Some protocols “recover” lost packets
- Data Link Control Protocols-DLCs (details later)
- Transmission Control Protocol-TCP (details later)
Packet loss

packets *queue* in router/switch buffers

- packets queue, wait for turn
- arrival rate to link \( R_A + R_B > R \) (temporarily) exceeds output link capacity leads to packet loss

Rate generated by \( A = R_A \)

Rate generated by \( B = R_B \)

packet being transmitted (transmission delay)

R: link transmission rate

packets in buffers (queueing delay)

free (available) buffers: arriving packets dropped (loss) if no free buffers

---

Bit Errors

Bit errors can lead to packet loss, protocols do *error detection*

Bit errors rate (BER) can lead to packet loss

Lost packets can recovered by

- Error correction
  - Forward Error Correction Coding (FEC)
- Retransmission
- Both FEC and Retransmission

Error environment: The BER can range

- Coaxial links: \( 10^{-9} \) to \( 10^{-6} \)
- Fiber optic links: \( 10^{-12} \) (after FEC)
- Wireless links: \( 10^{-5} \) to \( 10^{-3} \)

BER depends on

- Signal strength,
- Signal quality (receiver cost),
- Noise,
- Interference,
- Channel effects, e.g., fading
Bit Errors

Model 1: Random, bit errors are statistically independent
Model 2: Bursty, bit errors are correlated and come in groups

BER = $10^{-12}$  BER = $10^{-1}$

Bit Errors-time between errors

Example:
- Line rate = 600 Mb/s
- Bit error rate (BER) = $10^{-9}$

What is the time between errors?
- On average see one error in $10^9$ bits
- $(10^9 \text{ bits/error})/(600 \text{ Mb/s}) = 1.66 \text{ sec between errors}$
Bit Errors

Example: Impact of delay and errors:
- Link rate 600 Mb/s
- Free Space
- Link distance 3000 km = 10 ms
- Packet size:
  - Payload 1000 bytes

\[ \text{8000 bits/(600Mb/s)} = 13.3 \mu s/\text{packet} \]
\[ \text{10ms/(13.3 \mu s/\text{packet})} = 750 \text{ packets in flight} \]

\[ \rightarrow \text{Many packets can be in transit.} \]

Question: How do you cope with packets in error?

Throughput

- **throughput**: rate (bits/time unit) at which bits are being sent from sender to receiver
  - **instantaneous**: rate at given point in time
  - **average**: rate over longer period of time

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020
Throughput

**Throughput** in b/s or packets/sec, Normalized throughput

\[ R_{\text{ave}} = \text{Average error free rate (b/s) passing a reference point in the network} \]

\[ R = \text{Link Capacity (b/s) = Peak link rate} \]

\[ S = \% \text{ time the network is carrying error free packets-goodput} \]

---

**Throughput measurements**

| Network Technology | Mobility Patterns | Car
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Static</td>
<td>Static</td>
</tr>
<tr>
<td>66.9 (22.0, 202.5)</td>
<td>5</td>
<td>260</td>
</tr>
<tr>
<td>4G</td>
<td>42.6 (21.3, 77.2)</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Application</th>
<th>Mobility Patterns</th>
<th>Car</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Static</td>
<td>Static</td>
</tr>
<tr>
<td>Netflix</td>
<td>13.7 (0.5, 31.1)</td>
<td>10</td>
</tr>
<tr>
<td>Amazon Prime</td>
<td>6.9 (0.3, 11.2)</td>
<td>8</td>
</tr>
</tbody>
</table>

From: Beyond throughput, the next generation: a 5G dataset with channel and context metrics, Darijo Raca, Dylan Leahy, Cormac J. Sreenan, and Jason J. Quinlan MMSys '20: Proceedings of the 11th ACM Multimedia Systems Conference May 2020 Pages 303–308

https://doi.org/10.1145/3330820.3394038
Throughput

$R_s < R_c$ What is average end-end throughput?

$R_s > R_c$ What is average end-end throughput?

link on end-end path that constrains end-end throughput

Throughput: network scenario

- per-connection end-end throughput: $\min(R_c, R_s, R/10)$
- in practice: $R_c$ or $R_s$ is often bottleneck

10 connections (fairly) share backbone bottleneck link $R$ bits/sec

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020
**Throughput**

**Channel Capacity**, $S_{\text{MAX}}$, is the maximum obtainable throughput over the entire range of input traffic intensities, i.e., offered load.

![Graph showing Throughput vs Normalized Offered Load](image)

```
Maximum Throughput = S_{\text{max}}
```

**Normalized Offered Load**

Throughput

From: ATM WAN performance tools, experiments, and results
Utilization

Channel (or link) utilization:
- The % time the channel (or link is busy)

Channel Efficiency
- The % time the channel is carrying user information
  (impact of overhead)

Example: Let
\[ D = \text{\# user data bits/packet} = \text{Payload} \]
\[ H = \text{\# network overhead bits/packet} = \text{number of bits in the header} \]

Then
\[ \text{Channel efficiency} = S\left(\frac{D}{D+H}\right) \]

Reliability

Reliability: The reliability of a network can be defined as the probability that the functioning nodes are connected to working links.

Reliability = 1 - Network Failure

Here lets assume all nodes are working and analyze the “Reliability” of basic ring and tree networks where only links fail
Reliability

Tree Network Topology

Ring Network Topology

NE = Network Element

Aside: Other Network Topologies

Full Mesh Network Topology

Bus Network Topology

- Note a wireless network can be viewed as a bus topology
- All users hear all transmissions.
- Use of the transmission media coordinated using a MAC protocol.
Aside: Other Network Topologies

Key concepts:
1) **Broadcast**
   Everyone on the same “Network” can directly communicate, using a point-to-point link, DLC protocol.
   2) Packet delivered to one special node, i.e., a router, connected to the “Network” is sufficient for delivery to any destination connected to the broadcast network.

Network Performance Criteria: Reliability

Example:
- Reliability for a 5 **node tree network**
- Any of the 4 links fail the network is down
- Let $p = $ probability of link failure
- Assume failures are statistically independent
- $q = 1-p = $ probability of link operational
- Then Reliability = Prob[all links operational] = $(1-p)^4$
- Prob[network failure] = 1 - $(1-p)^4$
Network Performance Criteria:
Reliability

But \((1-p)^4 = 1 - 4p + 6p^2 - 4p^3 + p^4\)

\[
\text{Prob[network failure]} = 4p - 6p^2 + 4p^3 - p^4
\]

Assuming \(p\) is small then for 5 node tree network (4 links)

\[
\text{Prob[network failure]} \approx 4p
\]

Reliability \(\approx 1 - 4p\)

---

Network Performance Criteria:
Reliability

Reliability for a 5 node ring network

Ring network has 5 links

Ring network can have one link failure and still be working, note one more link can fail

Let \(q = 1 - p\) = probability of link good

\[
\text{Prob[network good]} = \text{Prob[all good or (one failed and 4 good)]} = \text{Prob[all good]} + \text{Prob[one failed and 4 good]} = q^5 + 5p q^4
\]

\[
\text{Prob[all good]} = q^5
\]

\[
\text{Prob[one failed and 4 good]} = \sum_{j} \text{Prob[link } j \text{ failed and all other links good]} = 5pq^4
\]

\[
\text{Prob[network failure]} = 1 - q^5 - 5p q^4
\]

Reliability \(= q^5 + 5p q^4\)
Switching, impairments, metrics ...

Network Performance Criteria: Reliability

Expanding Prob[network failure] = 10p²q³ + 10p³q² + 5p⁴q + p⁵
The dominant term (assuming p small) is 10p²q³
Reliability = 1-10p²q³

<table>
<thead>
<tr>
<th>p</th>
<th>4p</th>
<th>10p²q³</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>0.04</td>
<td>0.00097</td>
</tr>
<tr>
<td>0.001</td>
<td>0.004</td>
<td>10⁻⁵</td>
</tr>
<tr>
<td>10⁻⁵</td>
<td>4x10⁻⁵</td>
<td>10⁻⁹</td>
</tr>
<tr>
<td>10⁻⁷</td>
<td>4x10⁻⁷</td>
<td>10⁻¹³</td>
</tr>
</tbody>
</table>

Network Performance Criteria: Reliability

Reliability of Tree vs Ring Network
Other Metrics

Call/Session Blocking Probability
Fairness,
➢ N flows
➢ Allocate $X_i$ resources to flow $i$
➢ Jain’s Fairness Index (JFI)

$$JFI = \frac{\left(\sum_{i=1}^{N} X_i\right)^2}{N \sum_{i=1}^{N} X_i^2}$$

Security

CoS vs QoS

Class of Service,
➢ Provides for priority ordering of packet transmission
➢ No guarantees of delay or packet loss
➢ Example: Video/voice packets are transmitted before “Best Effort (BE)” packets

Quality of Service
➢ Reserve resources for flow
➢ Provides statistical performance guarantee
➢ Example: 95% of all packets receive a delay of less than 50ms.
Application Service Requirements

Data loss

- some apps (e.g., audio) can tolerate some loss
- other apps (e.g., file transfer, telnet) require 100% reliable data transfer

Throughput

- some apps (e.g., multimedia) require minimum amount of throughput to be “effective”
- other apps (“elastic apps”) make use of whatever throughput they get

Timing

- some apps (e.g., Internet telephony, interactive games) require low delay to be “effective”

Service requirements: common apps

<table>
<thead>
<tr>
<th>application</th>
<th>data loss</th>
<th>throughput</th>
<th>time sensitive?</th>
</tr>
</thead>
<tbody>
<tr>
<td>file transfer/download</td>
<td>no loss</td>
<td>elastic</td>
<td>no</td>
</tr>
<tr>
<td>e-mail</td>
<td>no loss</td>
<td>elastic</td>
<td>no</td>
</tr>
<tr>
<td>Web documents</td>
<td>no loss</td>
<td>elastic</td>
<td>no</td>
</tr>
<tr>
<td>real-time audio/video</td>
<td>loss-tolerant</td>
<td>audio: 5Kbps-1Mbps</td>
<td>yes, 10’s msec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>video:10Kbps-5Mbps</td>
<td></td>
</tr>
<tr>
<td>streaming audio/video</td>
<td>loss-tolerant</td>
<td>same as above</td>
<td>yes, few secs</td>
</tr>
<tr>
<td>interactive games</td>
<td>loss-tolerant</td>
<td>Kbps+</td>
<td>yes, 10’s msec</td>
</tr>
<tr>
<td>text messaging</td>
<td>no loss</td>
<td>elastic</td>
<td>yes and no?</td>
</tr>
</tbody>
</table>

Elastic applications make use of available throughput. Elastic services support this applications.
## Application Requirements

<table>
<thead>
<tr>
<th>Traffic type</th>
<th>Bit rate</th>
<th>Loss rate</th>
<th>Delay</th>
<th>Jitter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>E-commerce</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Transaction</td>
<td>Low</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Email</td>
<td>Low</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Telnet</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
</tr>
<tr>
<td>Browsing</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>File transfer</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Video conferencing</td>
<td>High</td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td>PnP control message</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
</tr>
</tbody>
</table>

Jitter is the variation in the delay of consecutive packets.

---

## Network Performance Criteria: Example

### TABLE II

<table>
<thead>
<tr>
<th>QCI</th>
<th>Resource Type</th>
<th>Priority</th>
<th>Packet Budget [ms]</th>
<th>Delay [ms]</th>
<th>Packet Loss Rate</th>
<th>Example services</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GBR</td>
<td>2</td>
<td>100</td>
<td>10^{-2}</td>
<td></td>
<td>Conversational voice</td>
</tr>
<tr>
<td>2</td>
<td>GBR</td>
<td>6</td>
<td>150</td>
<td>10^{-3}</td>
<td></td>
<td>Conversational video (live streaming)</td>
</tr>
<tr>
<td>3</td>
<td>GBR</td>
<td>5</td>
<td>300</td>
<td>10^{-6}</td>
<td></td>
<td>Non-Conversational video (buffered streaming)</td>
</tr>
<tr>
<td>4</td>
<td>GBR</td>
<td>7</td>
<td>50</td>
<td>10^{-3}</td>
<td></td>
<td>Real-time gaming</td>
</tr>
<tr>
<td>5</td>
<td>non-GBR</td>
<td>9</td>
<td>100</td>
<td>10^{-6}</td>
<td></td>
<td>IMS signaling</td>
</tr>
<tr>
<td>6</td>
<td>non-GBR</td>
<td>3</td>
<td>100</td>
<td>10^{-3}</td>
<td></td>
<td>Voice, video (live streaming), interactive gaming</td>
</tr>
<tr>
<td>7</td>
<td>non-GBR</td>
<td>4</td>
<td>300</td>
<td>10^{-6}</td>
<td></td>
<td>Video (buffered streaming)</td>
</tr>
<tr>
<td>8</td>
<td>non-GBR</td>
<td>8</td>
<td>300</td>
<td>10^{-6}</td>
<td></td>
<td>TCP based (e.g., WWW, e-mail), chat, FTP, P2P file sharing</td>
</tr>
<tr>
<td>9</td>
<td>non-GBR</td>
<td>1</td>
<td>300</td>
<td>10^{-6}</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The larger the value of the Priority value the higher priority (more important) the packet.

QCI = Quality-of-Service Class Identifier

GBR = Guaranteed Bit Rate

IMS = IP Multimedia Core Network Subsystem

LTE = Long Term Evolution, aka 4G

From: CAPOZZI et al.: DOWNLINK PACKET SCHEDULING IN LTE CELLULAR NETWORKS: KEY DESIGN ISSUES AND A SURVEY, IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 15, NO. 2

Switching impairments, metric...
Implementing Class of Service: Non-preemptive Priority

- **Scheduling discipline** chooses among queued datagrams for transmission
- Datagrams contain a field (tag or label) containing the level of priority

Packet Priority Assignment: Each incoming packet has a field (tag or label) containing the level of priority. Here the larger the value of the label the higher priority (more important) the packet.

Priority Queues: The router maintains multiple priority queues, each corresponding to a different priority level. Packets are placed into these queues based on their assigned priorities.

Packet Selection: The router selects the packet from the highest-priority queue that is ready to be transmitted. Non-preemptive priority means that once a packet is chosen for transmission the packet completes transmission.

Complete Transmission: The selected packet is transmitted without interruption. Only when the packet is fully sent will the router consider packets other for transmission.

Continue Scheduling: After transmitting a packet, the router selects the highest-priority packet that is ready to be transmitted.
Implementing Class of Service:
Non-preemptive Priority

Problems with non-preemptive priority

- Lower priority packets may never get to be sent, blocked by higher priority packets
- Packets within a priority class are severed FIFO, therefore there are no delay/loss guarantees even for higher priority packets.

Example: Traffic going to output port.

- Class 3 (highest priority, e.g., network management packets)
  - 100 bits/packet at 1000 packets/sec = 0.1 Mb/s
- Class 2 (Medium priority, e.g., web packets)
  - 45000 bits/packet at 1000 packets/sec = 45 Mb/s
- Class 1 (Low priority, e.g., best effort)
  - 3500 bits/packet at 1000 packets/sec = 3.5 Mb/s

Arriving from switch fabric

Packets

Server (R b/s)

Case 1: High load
- R= 50Mb/s
- Total Load=(0.1+45+3.5)/50=97.2%
- Class 3 packet may have to wait
  45000/(50Mb/s)=0.9ms for one class 1 packet to complete before an opportunity to transmit
- All classes see finite delays (large)

Case 2: Over loaded
- R= 47Mb/s
- Total load >1
- Class 3 + Class 2 Load = ~96% & see finite delays
- Class 1 packets see an over loaded system and average delay $\to \infty$

Case 3: Low load
- R=1 Gb/s Total Load=(0.1+45+3.5)/1000=5%
- Average delay $\approx$clocking time, i.e., queues likely empty
Network Design Problem

Goal

➢ Given
  - QoS requirement, e.g.,
    Average delay
    Loss probability
  - Characterization of the traffic: the input to the
    network
    Common traffic characteristics
    Average interarrival time (arrival rate)
    Average holding time (message length)

➢ Design the system

Network Performance Evaluation

Solution methodologies:

➢ Mathematical analysis
  - Model this type of process as a Queueing
    System ➔ good for initial design
  - Provide insights into protocol operation and
    performance

➢ Simulation techniques ➔ good for more
detailed analysis
Performance

Traffic modeling
- Describes the nature of what is transported over communications networks.
- Understanding traffic can be used to improve network performance
- Traffic is random
  - Time between packet arrivals, interarrival time, $T_a$ is now a random variable
  - Average rate of packet arrivals = $\lambda$, e.g., in packets/sec
  - Packet length, $L$, is now a random variable
    - $E[L] = \text{Expected value of the length (mean or average), e.g., in bits/packet}$
    - Clocking time (Holding time = $L/R=T_{hi}$) is now a random variable
      - $E[L]/R$
      - Example, $E[L]=1000 \text{ bits}$, $R=1\text{Mb/s}$ then average holding time = $1\text{ms}$

Traffic Characterization

Customers request information
Rate of requests = $\lambda$ requests/sec
- Calls/sec
- Packets/sec
- mp3’s/hour

The volume of information requested
- Length of the phone call (sec/call)
- Length of movie (Bytes)
- Size of picture (Bytes)
Traffic Characterization

Traffic characterization describes the user demands for network resources

- How often a customer:
  - Requests a web page
  - Downloads an MP3
  - Makes a phone call

- Size/length (how long you hold network resources)
  - Web page
  - Song
  - Phone call


Sample Realization of an Traffic Process

<table>
<thead>
<tr>
<th>Message number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interarrival time between (i) and (j) message (seconds)</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td>Length of (i) message (seconds)</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

Arrival Events & Lengths

A TES-based model for compressed "Star Wars" video, B. Melamed, D. Pendarakis, 1994 IEEE GLOBECOM. Communications: Communications Theory Mini-Conference Record
Traffic:

General Characteristics

Highly variable
Likely to change as new services and applications evolve.
Highly bursty, where one definition of burstyness is:

\[
\text{Burstyness} = \frac{\text{Peak rate}}{\text{Average rate}}
\]

Example: During a remote login connection over a 19.2kb/s modem a user types at a rate of 1 symbol/sec or 8 bits/sec and then transfers a 100 kbyte file. Assume the total holding time of the connection is 10 min.

What is the burstyness of this data session?
Traffic:
General Characteristics

The time to transfer the file is
\((800,000 \text{ bits})/(19,200 \text{ b/s}) = 41 \text{ sec.}\)
So for 600 - 41sec = **559 sec**.
the data rate is 8 bits/sec or
4,472 bits were transferred in 559 sec.
Thus in 600 sec. 4,472 + 800,000 bits were transferred,
yielding a average rate of:
804,472 bits/600 sec = **1,340 bits/sec**.
The peak rate was 19.2 Kb/s so the burstyness for this
data session was:

\[\frac{19,200}{1,340} = 14.3\]

Traffic:
General Characteristics

Asymmetric Nature of Interactive Traffic

This Asymmetric property has lead to asymmetric services
Time of Day Variations

From the Internet into Datavision
Mean = 8.876 Mb/s.
Maximum = 18.952 Mb/s

From Datavision out to the Internet
Mean = 5.133 Mb/s.
Maximum = 12.093 Mb/s

Video Traffic
January 2015
Top line (Total) is HTTP+HTTPS
Red is (HTTPS)
YouTube
Green is NetFlix
Blue is Twitch

From: NetFlix: Traffic Characterization, Michel Laterman Department of Computer Science, University of Calgary; https://pages.cpsc.ucalgary.ca/~carey/CPSC641/slides.html
In General Traffic

Very bursty
Problems with traffic modeling
- Rapidly evolving applications
- Complex network interactions

Packet Voice (applies to packet video)

Packet voice/video looks like a steady flow or Constant Bit Rate (CBR) traffic
However, voice/video can be Variable Bit Rate or VBR
- “silence detection”
- Variable rate coding
Problem: After going through the network the packets will not arrive equally spaced in time. Thus playback of packet voice must deal with variable network delays
Packet Voice (applies to packet video)

Example: Parameters for a packet voice system
- 1 source
- Sample rate = 8000 samples/sec (ITU G.711)
- 8 bits/sample (1 byte/sample)
- **8 ms of voice/packet ← Critical parameter**
- Packet size (bytes/packet) = (8ms/packet)*(8000 bytes/sec)=64 Bytes [assuming no overhead bytes]
- Link rate = 10 Mb/s
- Clocking time/packet
  (or holding time/packet or service time/packet aka, serialization time)
  = (64bytes/packet)*8bits/byte)/(10 Mb/s)= 51.2us

Voice Traffic: Packet Voice

51.2 us | Transmit at a Constant Bit Rate (CBR)
----|---

8 ms | 8 ms

51.2 us | Receive with variable interpacket arrival times
----|---

X ms

X not equal 8ms because of random network delays
If X is too big packet may arrive too late for play out
Voice Traffic: Packet Voice

Assume network delay is uniformly distributed between [25 ms, 75 ms]
- Same as having a fixed propagation delay of 25 ms with a random network delay uniformly distributed between [0 ms, 50 ms]

Note receiver will run out of bytes to playout after 8 ms.
Solution:
- Jitter Buffer Memory to hold 50 ms of the voice signal (or 8 packets or 2.8 Kbits)
- Worst case, receiver will run out of data just as a new packet arrives

Delay Jitter

Consider end-to-end delays of two consecutive packets: difference is random (transmission time difference)

Streaming Multimedia: Client Buffering

- client-side buffering, playout delay compensate for network-added delay, delay jitter

Voice Traffic: Packet Voice

- New problem: networks delays are unknown and maybe unbounded
- A voice packet may arrive at 85 ms and be too late to be played back
  - Late packets are dropped
  - Last packet may be played out in dead time
- Packet voice (video) schemes must be able to deal with variable delay and packet loss
  (Should voice/video packets be retransmitted?)

Fixed Playout Delay

- sender generates packets every 20 msec during talk spurt.
- first packet received at time r
- first playout schedule: begins at p
- second playout schedule: begins at p'

Streaming stored video

1. video recorded (e.g., 30 frames/sec)
2. video sent
3. video received, played out at client (30 frames/sec)

network delay (fixed in this example)

streaming: at this time, client playing out early part of video, while server still sending later part of video
Streaming stored video: playout buffering

- **client-side buffering and playout delay**: compensate for network-added delay, delay jitter

---

**Voice over IP (VoIP) Quality**

The mouth-to-ear delay is the time taken from when a user begins to speak until when the listener actually hears the speech. This one-way latency is known as mouth-to-ear delay.

The E-model (ITU-T Rec. G.107) is a transmission planning tool that provides a prediction of the expected voice quality, as perceived by a typical telephone user.

---

ITU-T Recommendation G.14-One-way transmission time, May 2003
VoIP- Delay budget
Factors in End to End Delay

Assumption: maximum delay from mouth-to-ear needs to be on the order of 200-300 ms

ITU G.114
- < 150 ms acceptable for most applications
- [150ms, 400 ms] acceptable for international
- > 400 ms unacceptable

Source
Codec → Packetization → Queue → Clock Out

Network

Destination
Clock In → Jitter Buffer → Codec

Codec=coder/encoder, aka, Analog-to-digital converter (A/D)

Clock Out, aka, serialization, modeled by the “server”

Delay & Packet Loss Sources

Processing time 400ms
Packetization 400ms
Serialization 400ms
End device 400ms

Clocking time

PSTN=Public Switched Telephone Network

VoIP- Delay budget
Factors in End-to-End Delay

Example: Delay Budget (depends on assumptions)
- Formation of VoIP packet at TX ~ 30 ms
  20ms of voice/packet is default for Cisco 7960 router
- Other VoIP packet processing ~70 ms
  (see: http://www.rmav.arau.br/pdf/voip.pdf)
- Propagation ~10 ms
- Network Delays ~10 ms
- Extraction of VoIP packet at Receiver ~30 ms
- Jitter Buffer ~ 100 ms
  Compensates for variable network delay
- Total 250 ms

Possible trade-offs:
- Jitter Buffer vs voice packet loss
- VoIP packet size vs length of jitter buffer

For examples see: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/5125-delay-details.html#packetizationdelay


Voice Traffic: Packet Voice

G.723.1 is a voice coding standard, linear prediction compression algorithm

From: Performance Evaluation of the Architecture for End-to-End Quality-of-Service Provisioning,
Katsuyoshi Iida, Kenji Kawahara, Tetuya Takine, and Yuji Oie, IEEE Communications Magazine, April 2000
Network Design Problem

Goal

- Given
  - QoS requirements, e.g.,
    - Average delay
    - Loss probability
  - Characterization of the traffic, e.g.,
    - Average interarrival time (arrival rate)
    - Average holding time (message length)

- Design the system, e.g., determine link capacity and system size

Three systems will be studied:

- Ideal (infinite buffer) router output port, e.g., determine link capacity
  System 1 \( \rightarrow \) M/M/1 (M/M/1/\infty / \infty)

- Real router output port (finite buffer), e.g., determine link capacity and buffer size
  System 2 \( \rightarrow \) M/M/1/S (M/M/1/S/\infty / \infty)

- Circuit switch, e.g., determine the # lines
  System 3 \( \rightarrow \) M/M/S/S (M/M/S/S/\infty / \infty)

Network Performance Evaluation:

Elements of a Queueing System

One Queue/One Server
Network Performance Evaluation: Elements of a Queueing System

Traffic: Arrivals & Departures

Packet Arrivals

Interarrival Time = \( T_a \)

Packet Departures

Assuming No Queueing

\( T_H = \text{Holding, aka Clocking, aka Service time } = \frac{L}{R_{out}} \)

Average Service time = \( E[L]/R_{out} \)  \( E[] = \text{Expected value} \)
Network Performance Evaluation:
Assumptions and Definitions

Packet interarrival times $T_a$ are exponentially distributed – Markov Process
- Arrival Rate (packets/sec) = $\lambda$
  $$P[T_a < t] = 1 - e^{-\lambda t}$$

Clocking, (Service or Holding) times $T_H$ are exponentially distributed – Markov Process
- Packet length= $L$ = packet length in bits
- Average Service (Holding time) = $E[T_H] = E[L]/R_{out}$ where $R_{out}$ = link capacity in b/s
- Service rate (packets/sec) = Departure rate (packets/sec) = $\mu = 1/E[T_H] = R_{out}/E[L]$

Average input rate (b/s) $R_{in} = \lambda E[L]$ bits/sec
Average departure rate (b/s) $R_{out} = \mu E[L]$ bits/sec
Traffic intensity (load) =
  $$\rho = R_{in}/R_{out} = \lambda E[T_H] = (\lambda E[L])/R_{out} = \lambda/\mu = \text{Arrival rate/service rate (units Erlang)}$$

A. K. Erlang was a Danish mathematician, statistician and engineer, who invented the fields of traffic engineering and queueing theory. Major paper 1909.
Show animated example

Arrival - Departure Process

Example

Average packet length = 1000 bytes/packet
Output link rate = \( R_{out} = 50 \text{ Mb/s} \)
Arrival rate = \( \lambda = 4000 \text{ packets/sec} \)
- \( R_{in} = \lambda E[L] = (4000 \text{ packets/sec}) \times (8 \times 1000 \text{ bits/packet}) = 32 \text{ Mb/s} \)
- Load = \( \rho = \frac{32}{50} = 0.64 \)
- Service rate \( \mu = (50 \text{ Mb/s}) / (8 \times 1000 \text{ bits/packet}) = 6250 \text{ packets/sec} \)
- Load = \( \rho = \frac{\lambda}{\mu} = \frac{4000}{6250} = \frac{R_{in}}{R_{out}} = 0.64 \)
Network Performance Evaluation: Queueing System Notation (Kendall’s notation)

A / b / m / K / L

A = type of arrival process: \( M = \) Markov Process
- Time between arrivals has an exponential pdf

b = type of service process: \( M = \) Markov Process
- Service time between arrivals has an exponential pdf

m = number of servers

K = maximum number of elements allowed in the system = system size (if K missing then \( \infty \))

L = population size (if L missing then \( \infty \))

Network Performance Evaluation: Specific cases

Three types of systems considered in this class

- One server (Stat Mux) \( \rightarrow \) Router output port
  - Infinite memory-\( M/M/1 \)
  - Finite memory (if full then drop packet) - \( M/M/1/S \)

- \( S \) servers and a system size of \( S \)
  (if full then drop) - \( M/M/S/S \)

No buffer/queue
Network Performance Evaluation:
Summary of results for specific cases

M/M/1
- One Server
- Infinite system size
- Link Rate= $R_{out}$ (b/s)
- $L =$ Packet Length (bits): pdf ~ exponential; Average packet length=E[L]
- Packet Arrival rate = $\lambda$ (packets/sec)
- $T_a =$ interarrival times: pdf ~ exponential; $E[T_H] = E[L]/R_{out} = 1/\mu$
- Service rate= $\mu = R_{out}/E[L]$ (packets/sec)
- $E[T_H] = E[L]/R_{out} = 1/\mu$
- Load = $\rho = R_{in}/R_{out} = \lambda E[T_H] = \lambda E[L]/R_{out} = \lambda/\mu$

Probability of $k$ in system = $P[K=k] = \rho^k(1-\rho)$
Probability of system busy = utilization = $\rho$
Probability of system empty = $1-\rho$

Average Number in System = $E[K] = \frac{\rho}{1-\rho}$
Variance of Number in System = $\text{Var}[K] = \frac{\rho^2}{(1-\rho)^2}$

Average Delay = $E[D] = \frac{E[T_H]}{1-\rho} = \frac{E[L]}{R_{out}} = \frac{1}{\mu-\lambda}$

Network Performance Evaluation:
Summary of results for specific cases

M/M/1/S
- One Server
- System size = $S$ (server+buffer)
- Rate= $R_{out}$ (b/s)
- Packet arrival rate = $\lambda$ (packets/sec)
- $L =$ Packet Length (bits): pdf ~ exponential; $E[L]$
- $T_a =$ interarrival times: pdf ~ exponential; $E[T_H] = E[L]/R_{out}$
- Load = $\rho = R_{in}/R_{out} = \lambda E[T_H] = \lambda E[L]/R_{out}$

$P[K = k] = \frac{(1-\rho)\rho^k}{1-\rho^{k+1}}$ for $k \leq S$
$P[K = k] = 0$ for $k > S$

$P_{\text{Blocking}} = P[K = S] = \frac{(1-\rho)\rho^S}{1-\rho^{S+1}}$

Table to be provided on test and Excel spreadsheet provided on class web site
see http://www.ittc.ku.edu/~frost/EECS\_563/M-M-1-K-Blocking%20cal.xls
Network Performance Evaluation:
Summary of results for specific cases

M/M/S/S (Erlang B)
- \( S = \) Servers
- \( S = \) System size
- Arrival rate = \( \lambda \) (items/sec)
- \( L = \) Item Length (sec): pdf ~ exponential; \( E[L] \)
- \( T_a = \) interarrival times: pdf ~ exponential;
- Load = \( \rho = \lambda E[L] \rightarrow \) units Erlang

Erlang B blocking Formula
Tabulated and there are web calculators see:
http://www.erlang.com/calculator/index.htm

Network Performance Evaluation:
Summary of results for specific cases

M/M/1
- Probability of \( k \) in system = \( P[K=k] = \rho^k(1-\rho) \)
- Probability of system busy = utilization = \( \rho \)
- Probability of system empty = 1 - \( \rho \)
- Average Number in System = \( E[K] = \frac{\rho}{1-\rho} \)
- Variance of Number in System = \( \text{Var}[K] = \frac{\rho(1-\rho)}{(1-\rho)^2} \)
- Average Delay = \( E[T_D] = \frac{E[L]}{R_{out}} = \frac{1}{1-\rho} = \frac{1}{\mu-\lambda} \)
- Load = \( \rho = \frac{R_{in}}{R_{out}} = \lambda E[T_H]/R_{out} = \frac{\lambda E[L]}{R_{out}} = \lambda/\mu \)

M/M/1/S
- \( P[K=k] = \frac{(1-\rho)\rho^k}{1-\rho^{S+1}} \) for \( k \leq S \)
- \( P[K=k] = 0 \) for \( k > S \)
- Erlang B blocking Formula
Tabulated and there are web calculators see:
http://www.erlang.com/calculator/index.htm

Table to be provided on test and Excel spreadsheet provided on class web site
http://www.ittc.ku.edu/~frost/EECS_563/M-M-1-K-Blocking%20cal.xls

M/M/S/S
- \( P[K=k] = \frac{\rho^k}{k! \sum_{n=0}^{\infty} \rho^n n!} \) for \( k \leq S \)
- \( P[K=k] = 0 \) for \( k > S \)
- Erlang B blocking Formula
Tabulated and there are web calculators see:
http://www.erlang.com/calculator/index.htm
Delay Analysis for M/M/1

Network Performance Evaluation: M/M/1

$\lambda=500$ & $E[L]=1000$ bits, $C=1$Mb/s $\Rightarrow$ Load $= 0.5$ & $E[K]=1$
Network Performance Evaluation: $M/M/1$

\[ \lambda = 950 \text{ & } E[L] = 1000 \text{ bits, } C = 1 \text{Mb/s} \rightarrow \text{Load} = 0.95 \quad E[K] = 19 \]

Final simulated value = 12.05

Network Performance Evaluation: $M/M/1$

Impact of High Load on Variance

Load = 0.95 \quad E[K] = 19
Examples of Delay Analysis for M/M/1

Example 1: \( \lambda = 950 \), \( E[L] = 1000 \) bits, \( R_{out} = 1\) Mb/s
- \( E[L]/R_{out} = 1\) ms \( \Rightarrow \) Load = \( \rho = 0.95 \)
- \( E[Delay] = 1\) ms/(1-0.95) = 20 ms

Example 2: \( \lambda = 500 \), \( E[L] = 1000 \) bits, \( R_{out} = 1\) Mb/s
- \( E[L]/R = 1\) ms \( \Rightarrow \) Load = \( \rho = 0.5 \)
- \( E[Delay] = 1\) ms/(1-0.5) = 2 ms

Example 3: \( \lambda = 100 \), \( E[L] = 1000 \) bits, \( R_{out} = 1\) Mb/s
- \( E[L]/R = 1\) ms \( \Rightarrow \) Load = \( \rho = 0.1 \)
- \( E[Delay] = 1\) ms/(1-0.1) = 1.11 ms \( \sim \) \( E[L]/R_{out} = 1\) ms
- At low loads \( \sim \) no queueing and delay = service time

Network Performance Evaluation:

Example

Which is better?

Dedicated Links

\[ \lambda \rightarrow \mu \]

Total capacity is the same in both cases

\[ \lambda \rightarrow \mu \]

versus

One consolidated system

\[ \lambda \rightarrow \mu \]

\[ m\lambda \rightarrow m\mu \]

Shared Link

Example: Traffic:
Assume average packet length = 1000 bits/packet
Assume arrival rate = \( \lambda = 50 \) packets/sec/system
Network Performance Evaluation:
Example

Example: Traffic:
Assume average packet length=1000 bits/packet
Assume arrival rate = \( \lambda \) = 50 packets/sec/system

Dedicated Links
- 10 \((m=10)\) systems each with their dedicated 100 kb/s link to a server. \((\text{total capacity} = 10*100\text{ kb/s} = 1\text{Mb/s})\)
- Service time = 1000 bits/100,000 b/s = 10ms
- Traffic intensity=load = 50*1000/100,000 = 0.5
- Average delay = 10ms/.5 = 20ms

Example: Traffic:
Assume average packet length=1000 bits/packet
Assume arrival rate = \( \lambda \) = 50 packets/sec/system

Shared link: all traffic shares one statistical multiplexer with \( R = 1 \text{Mb/s} \)
- Traffic intensity = 0.5
- Service time = 1 ms
- Average delay = 2 ms

This shows that traffic aggregation improves system performance

Advantage of Statistical Multiplexing
Do the Design Problem:
Find the link capacity between the stat mux and
the server such that the delay is 20 ms.

Example: Assume average packet length = 1000 bits/packet
Assume arrival rate = \( \lambda = 50 \) packets/sec/system
10 systems sharing one link

\[
E[D] = 0.02 \text{ sec} = \frac{1}{\mu - \lambda} \\
\mu - \lambda = 50 \\
\lambda = 500 \text{ (packets/sec)} \\
\mu = 550 \text{ packets/sec} \\
R_{\text{out}} = 1000 \text{(bits/packet)} \times 550 \text{(packets/sec)} = 550 \text{kb/s}
\]

Network Performance Evaluation:
Example: Finite buffers- M/M/1/S

Find blocking probability for this system:
Traffic specification:
- Average packet length = 1000 bits
- Arrival rate = 700 packets/sec

System Specifications:
- \( R_{\text{out}} = 1 \text{ Mb/s} \)
- System size (buffer + server) = 9 packets

\( R_{\text{in}} = \lambda E[L] = 700 \text{kb/s} \)
Load = \( \rho = \frac{R_{\text{in}}}{R_{\text{out}}} = 0.7 \)

\[
P_{\text{Blocking}} = P[K = S] = \frac{(1 - \rho)\rho^{S}}{1 - \rho^{S+1}}
\]

Or use http://www.ittc.ku.edu/~frost/EECS_563/M-M-1-K-Blocking%20cal.xls

Blocking Probability \( \sim 1.2\% \)

Show Extend Model:
Network Performance Evaluation:
Example: Design problem for M/M/1/S

Find the link rate $R_{out}$ to achieve a blocking probability of $\sim 3\%$.

Traffic specification:
- Average packet length = 1000 bits
- Arrival rate = 900 packets/sec

System Specifications:
- System size (buffer + server) = 9 packets


From table a load = 0.8 provides a Block Probability $\sim 3\%$

Load = $R_{in}/R_{out} = 0.8$
$R_{in} = 900$ kb/s
$R_{out} = 1.125$ Mb/s

At this load what is the maximum delay (for packets transmitted)?

$9 \times 1000$ bits/1.125Mb/s = 9 service times = 8 ms

Network Performance Evaluation:
Trade-offs- Finite buffers - M/M/1/S

Network Performance Evaluation: Design Example

Example:

- Link Rate = R b/s
- D = 2000m
- Business Campus Computer Center

Design the system, i.e., find the system size and link rate, \( R_{out} \), to meet the customer requirements

- Delay < 100 ms
- Loss < 10%

Assume customer traffic:

- Average packet length = 9000 bytes/packet
- 55 sources
- Packets are generated at a rate of 2 per second/source

Approach (This is results in an over designed system, why?)

- Find \( R_{out} \) first using only the delay specification, Delay < 100 ms, with the M/M/1 result, i.e., assume infinite systems size to find \( R_{out} \) and \( \rho \)
- Find \( S \) to get Loss < 10% using \( \rho \) and the M/M/1/S result
Network Performance Evaluation: Design Example

Step 1: M/M/1 analysis to find $R_{out}$
- $\lambda = 55 \times 2 = 110$ packets/sec
- $E[D] = 100 \text{ ms} = 1/10 = 1/(\mu - \lambda)$
- $\mu = 120$ packets/sec
- $R_{out} = \mu \times L = 120 \times 9000 \text{ Bytes/packet} \times 8 \text{ bits/Byte} = 8.64 \text{ Mb/s}$

Step 2: M/M/1/S analysis to find system size (K)
- $R_{in} = \text{Rate_in} = (110 \text{ packets/sec}) \times 9000 \text{ Bytes/packet} \times 8 \text{ bits/Byte} = 7.92 \text{ Mb/s}$
- $\rho = R_{in} / R_{out} = (7.92 \text{ Mb/s}) / (8.64 \text{ Mb/s}) = \lambda / \mu = 110 / 120 = 0.916$
- $\rho = 0.916$ and Blocking Prob = 0.1 $\rightarrow K = 7$

Final design is:
- $R = 8.64 \text{ Mb/s}$
- Average system size $\geq 7$ packets

Network Performance Evaluation: M/M/S/S

Holding time = 3 min,
Arrival rate = 0.833 calls/min
N = 4
Load = 3 $\times$ 0.833 = 2.5 Erlangs
$\rightarrow$ Theory $P_b = 0.15$
(From Erlang B table)
Simulated $P_b = 0.198$

https://www.erlang.com/calculator/

Show Extend Model:
http://www.ittc.ku.edu/~frost/EECS_563/LOCAL/Extend_Models_2019-v10/Telephone_Model-ESI0.mox
Network Performance Evaluation: M/M/S/S

Erlang B

\[
P[K = k] = \frac{\rho^k}{k!} \sum_{n=0}^{S} \frac{\rho^n}{n!}
\]

Example

- \(S=9\) (lines)
- Holding time = 3 min/call
- Arrival rate 2.16 calls/min
- Load = 3*2.16=6.5
  - From Table Blocking Prob ~10%

https://www.erlang.com/calculator/

Show Extend Model:

Network Performance Evaluation: Example

Design of a building phone system. The design goal is to minimize the number of lines needed between the building and the phone company. The blocking specification is \(P_{\text{blocking}} < 5\%\).

A building has four floors, on each floor is a separate department. Each department has 22 phones, each busy 10\% of the time during the busy hour.
Compare two designs

Acquire one telephone switch for each floor and purchase lines from each separate switch to the phone company.

Acquire one telephone switch for the building and purchase lines from the one switch to the phone company.

Network Performance Evaluation: Example-Case A

Acquire one telephone switch for each floor.

22 phones*0.1=2.2 Erlangs/floor

Use Erlang B table with 22 and $P_{\text{blocking}} \ 5\%$ to find $S=5$

5 lines/floor or 20 lines for the building.
Network Performance Evaluation:
Example-Case B

Acquire one telephone switch for the building.
88 phones @ .1 Erlang/phone = 8.8 Erlangs
8.8 Erlangs & B=5% gives:
13 lines for the building
Select Case B, Shared capacity
Again ➔ Aggregation/sharing improves system performance

Theoretical development of performance results

In extra slides and see....
http://www.ittc.ku.edu/~frost/EECS_563/LOCAL/
EECS_563_Class_Notes-Fall-
2021/Network_Performance_Analysis_2021.cdf
Communication Network Simulation

Communication network simulation involves generating *pseudo-random sequences representing network traffic* (message lengths and interarrival times or other input processes, e.g. time varying link quality) then using these sequences to *exercise an algorithmic description of the network operation to estimate system performance*.

<table>
<thead>
<tr>
<th>Message number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interarrival time between (i+1) and (i) message (seconds)</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td>Length of (i)th message (seconds)</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>--</td>
</tr>
</tbody>
</table>

Verification and Validation of Simulation Models

*Model*

- Mathematical (Algorithmic) Description of Behaviour of “Real Thing”

*Verification*

- Determining Whether the Simulation Model Performs As Intended
- In Programming Terminology, “Debugging”
- Example: Is *Statistical Multiplexer* Model Producing the specified Message Lengths?

*Validation*

- Determining Whether the Simulation Model Itself Is an “Accurate” Representation of the Communication Network Under Study (the “Real Thing”)
- Example: Is the Assumption the statistical message length model accurate?
Some Comments on Validation

Simulation Models Are Always Approximations
A Simulation Model Developed for One Application
May Not Be Valid for Others
Model Development and Validation Should Be Done Simultaneously
Specific Modeling Assumptions Should Be Tested
Sensitivity Analysis Should Be Performed
Attempt to Establish That the Model Results Resemble the Expected Performance of the Actual System
Generally, Validation Is More Difficult Than Verification

Quality of Performance Estimates

Simulation results are like laboratory measurements, they can be modeled as random variables
Performance estimates should have acceptable variance
More observation reduces the variance.
HOWEVER → Often observations, e.g., a sequence of packet delays, taken from network simulation will be correlated
   ➢ Cannot directly apply standard statistical approaches based on iid (Independent, Identically Distributed) observations
Dealing with Lack of Independence

Replication: Multiple Simulation Runs

- Assume Results for Each Replication Are Independent
- Can be Inefficient Because of Multiple Startup Periods

Multiple simulation runs are commonly used to estimate the standard deviation of performance estimates and form confidence intervals.

Extend®

View Discrete Event Quick Start Videos
SoE has 25 Extendsim Licenses

Assume you hold a Extendsim license for 2 hours
Requests for licenses come in at a rate of 11.5/hour
Load = 23 Erlangs
Probability you will be blocked from getting access to Extendsim = ~ 10%

Extra Slides
Network Simulation

Outline

- Define network simulation
- Discuss attributes and application of simulation
- Present implementation of simulation systems
- Discuss analysis of simulation results
- Discuss selection of simulation tools
- Provide an overview of ExtendSim. On the start up ExtendSim window there is:
  - A button for tutorials and a video showing how to build models
  - A link to “ExtendSim for DESS Textbook”, a that is a tutorial on the tool.
  - Other useful tools.
  - There is a link to getting the whole user manual on the class web page.
    (It is long DO NOT PRINT the whole pdf file.)

A Definition of Communication Network Simulation

Communication network simulation involves generating pseudo-random sequences representing network traffic (message lengths and interarrival times or other input processes, e.g. time varying link quality) then using these sequences to exercise an algorithmic description of the network operation.

<table>
<thead>
<tr>
<th>Message number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interarrival time between i-1 and i message (seconds)</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td>Length of i(^{th}) message (seconds)</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
Attributes of Simulation

Simulation Is a **Very Flexible** Evaluation Tool
- General Network Characteristics (Sources, Topology, Protocols, Etc.)
- Minute Detail

Simulation Models Can Be **Expensive to Construct**
- Human Effort

Simulation Models Can Be **Expensive to Run**
- Computer Effort

Statistical **Analysis of the Results** Can Be Difficult
- Requires Careful Interpretation

**Difficult to Gain Insight** Into System Behavior
- Simulate Only a Set of Specific Scenarios

When to Use Simulation

Whenever Mathematical **Analysis Is Difficult or Impossible**
- For Studying Transient Behavior of Networks
- For Systems With Adaptive Routing
- For Systems With Adaptive Flow Control
- For Systems With Blocking (Finite Buffers)
- For Systems With General Message Interarrival Statistics

For **Validating Analytic Models** and Approximations
- How Accurate Is the Model?
- Do Approximations Distort the Results?

For **Experimentation Without Disturbing** an Operational System
- Test Possible Modifications and Adjustments
Modeling Elements for Communication Networks

Traffic and Input Processes
- Message Arrival Process
  - Often Interarrival Times (probability density function)
- Message Lengths (probability density function)
- Other Message Attributes
  - Service Class
  - Error models

Algorithmic Descriptions of Network Processing
- Protocols
- Links and Queues
- Routing

Event Scheduling Approach to Network Simulation

Variable Time Advance
- Advance Time To Next Occurring Event

Update System State Only When Events Occur
- For Example, Arrivals or Departures

Event Calendar
- Events: Instantaneous Occurrences That Change the State of the System
- An Event is Described by
  - The Time the Event is to Occur
  - The Activity to Take Place at the Event Time
- The Calendar is a Time-Ordered List of Events
Event Scheduling Approach: Simplified Flow Control

An Executive (or Mainline) Controls the Selection of Next Event

- Use Event List to determine next event to process
- Advance simulation clock to event time
- Update system state using event routines
- Update event list using event routines

Event Scheduling for Simple Statistical Multiplexer

<table>
<thead>
<tr>
<th>Message number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interarrival time between i-1 and i message (seconds)</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>--</td>
</tr>
<tr>
<td>Length of i-th message (seconds)</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

Arrival
- Use random number generator to obtain next arrival time
- Schedule next arrival
- Assign a message length using a random number generator
- Schedule End of Transmission at time + message length
- Add message to buffer (if)
- Return

Departure (End of Transmission)
- Change status of transmission facility to idle
- Read & remove from file the attributes of next message to send
- Schedule End of Transmission
- Return

Buffer (Queue)
Verification and Validation of Simulation Models

Model
- Mathematical (Algorithmic) Description of Behaviour of “Real Thing”

Verification
- Determining Whether the Simulation Model Performs As Intended
- In Programming Terminology, “Debugging”
- Example: Is Statistical Multiplexer Model Producing the specified Message Lengths?

Validation
- Determining Whether the Simulation Model Itself Is an “Accurate” Representation of the Communication Network Under Study (the “Real Thing”)
- Example: Is the Assumption the statistical message length model accurate?

Some Comments on Validation

Simulation Models Are Always Approximations
A Simulation Model Developed for One Application May Not Be Valid for Others
Model Development and Validation Should Be Done Simultaneously
Specific Modeling Assumptions Should Be Tested
Sensitivity Analysis Should Be Performed
Attempt to Establish That the Model Results Resemble the Expected Performance of the Actual System
Generally, Validation Is More Difficult Than Verification
Analysis of Results: Statistical Considerations

Starting Rules
- Overcoming Initial Transients
- An Initial Transient Period Is Present Which Can Bias the Results
- Achieving Steady State
  - Use a Run-in Period:
    - Determine $T_b$ Such That the Long-Run Distribution Adequately Describes the System for $t > T_b$
  - Use a “Typical” Starting Condition (State) to Initialize the Model

Quality of Performance Estimates
- Variance of Estimated Performance Measures

Quality of Performance Estimates

Simulation results are like laboratory measurements, they can be modeled as random variables.
Performance estimates should have acceptable variance.
More observation reduces the variance.
HOWEVER ➔ Often observations, e.g., a sequence of packet delays, taken from network simulation will be correlated
  - Cannot directly apply standard statistical approaches based on iid (Independent, Identically Distributed) observations
Dealing with Lack of Independence

Replication: Multiple Simulation Runs

- Assume Results for Each Replication Are Independent
- Can be Inefficient Because of Multiple Startup Periods

Criteria for Selecting a Network Simulation Tool

- Availability
- Cost
- Usage
- Documentation
- Ease of Learning
- Computation Efficiency
- Flexibility
- Portability
- User Interface
- Extendibility

Common Tools

- ns-3 (http://www.nsnam.org/)
- Opnet (http://www.opnet.com/)
- QualNet (http://www.scalable-networks.com)
- ExtendSim
Guidelines to Network Modelling and Simulation

Things to Know
- Know the Customer
- Know the Network
- Know the Important Performance Metrics

Things to Do
- Establish a Credible Model
- Expect the Model to Evolve → Plan for success
- Apply Good Software Management Techniques

Conclusions

Simulation Can Be an Important Tool for Communication Network Design and Analysis
Care and Thought Must Go Into Construction of Communication Network Models
Care and Thought Must Go Into Interpretation of Model Output
Extend® Overview

Allows Graphical Description of Networks
 Sources, Links, Nodes, Etc.
Data Flow Block Diagrams
Hierarchical Structure to Control Complexity
Be sure and create libraries when creating complex models
Basic Queueing Theory

Network Performance Evaluation: Approach

- Analysis of a pure birth process to characterize arrival processes
- Extension to general birth/death processes to model arrivals and departures
- Specialization to the specific cases to find:
  - Probability of system occupancy,
  - Average buffer size,
  - Delay,
  - Blocking probability

**Goal:** Design and analyze statistical multiplexers and circuit switching systems
Network Performance Evaluation: Analysis of a Pure Birth Process

Arrivals and no departures

λ = Arrival rate

0 → 1 → ... → k → ... → k+1 → ...

Only Births (Arrivals) Allowed

k = System State (number in system)
- number of arrivals for 0 to t sec
- number in system at time t

Goal: Find Prob [k arrivals in a t sec interval]
Network Performance Evaluation:
Analysis of a Pure Birth Process

The number represents the **State** of the system. In networks this is usually the number in the buffer plus the number in service. *The system includes the server.*

The time to clock the message bits onto the transmission facility is the service time. The server is the model for the transmission facility.

Goal: Find Prob [k arrivals in a t sec interval] = P[k, t]

---

Network Performance Evaluation:
Analysis of a Pure Birth (Poisson) Process: Assumptions

\[
\begin{align*}
\text{Prob}[ \text{1 arrival in } \Delta t \text{ sec}] &= \lambda \Delta t \\
\text{Prob}[ \text{0 arrivals in } \Delta t \text{ sec}] &= 1 - \lambda \Delta t \\
\end{align*}
\]

Number of arrivals in non-overlapping intervals of times are statistically independent random variables, i.e.,

\[
\text{Prob}[ \text{N arrivals in } t, t+T \text{ AND M arrivals in } t+T, t+T+\tau] = \\
\text{Prob}[ \text{N arrivals in } t, t+T] * \text{Prob}[ \text{M arrivals in } t+T, t+T+\tau]
\]
Network Performance Evaluation:

Define probability of k in the system at time t = Prob[k, t]

Probability of k in the system at time t+ Δ t = Prob[k, t+ Δ t ]

= Prob[k, t+ Δ t] Prob[(k in the system at time t and 0 arrivals in Δ t)]
or (k-1 in the system at time t and 1 arrival in Δ t)]

= (1- λ Δ t ) Prob[k,t] + λ Δ t Prob[k-1,t]
Network Performance Evaluation: Analysis

Rearranging terms
\[
\frac{(\text{Prob}[k, t+ \Delta t] - \text{Prob}[k,t])}{\Delta t} + \lambda \text{Prob}[k,t] = \lambda \text{Prob}[k-1,t]
\]

Letting \( \Delta t \to 0 \) results in the following differential equation:

\[
\frac{d\text{Prob}[k, t]}{dt} + \lambda \text{Prob}[k, t] = \lambda \text{Prob}[k-1, t]
\]

For \( k = 0 \) the solution is:

\[ \text{Prob}[0, t] = e^{-\lambda t} \]

For \( k = 1 \) the solution is:

\[ \text{Prob}[1, t] = \lambda t e^{-\lambda t} \]

For \( k = 2 \) the solution is:

\[ \text{Prob}[2, t] = \frac{(\lambda t)^2 e^{-\lambda t}}{2} \]
Network Performance Evaluation: Analysis

In general the solution is a Poisson probability mass function of the form:

\[
\text{Prob} [k, t] = \frac{(\lambda t)^k e^{-\lambda t}}{k!}
\]

Network Performance Evaluation: Analysis

A Poisson pmf of this form has the following moments:

\[
E[k] = \lambda t
\]
\[
Var[k] = \lambda t
\]

Poisson Arrival Process
The number of arrivals in any T second interval follows a Poisson probability mass function.
Network Performance Evaluation: Interarrival Time Analysis

Let \( \Delta t \to 0 \) results in the following

\[
\text{Prob}[t<T_a<t+\Delta t] = \text{Prob}[0 \text{ arrivals in } t \text{ sec and } 1 \text{ arrival in } \Delta t]
\]

\[
\text{Prob}[t<T_a<t+\Delta t] = \text{Prob}[k=0,t]\text{Prob}[k=1, \Delta t]
\]

\[
\text{Prob}[t<T_a<t+\Delta t] = (e^{-\lambda t})\lambda\Delta t e^{-\lambda \Delta t}
\]

\[
P[T_a < t] = 1 - e^{-\lambda t}
\]
Network Performance Evaluation:
Interarrival Time Analysis

**MAIN RESULT:**
The interarrival time for a Poisson arrival process follows an exponential probability density function.

\[ E[T_a] = \frac{1}{\lambda} \quad Var[T_a] = \frac{1}{\lambda^2} \]

Network Performance Evaluation:
Birth/Death Process Analysis

Now allow arrivals and departures. The Model for the Birth/Death Process

Note that the arrival and service rates are now state dependent.
Network Performance Evaluation: Birth/Death Process Analysis

The departure process is Poisson--
Prob[ 1 departure in $\Delta t$ sec when the system is in state k ] = $\mu_k \Delta t$
Prob[ 0 departure in $\Delta t$ sec when the system is in state k ] = 1 - $\mu_k \Delta t$
Number of departures in non-overlapping intervals of times are statistically independent random variables
Probability[arrival AND departure in $\Delta t$] = 0
Poisson service process implies an exponential probability density function for the message length.
Network Performance Evaluation: Birth/Death Process Analysis

Specific queuing systems are modeled by

- Setting state dependent arrival rates, $\lambda_{k}$
- Setting the state dependent service rates, $\mu_{k}$
- Solving for the steady state probabilities


Network Performance Evaluation: Special cases: $A / b / m / K / L$

- $A = M \Rightarrow$ the arrival process is Poisson and the interarrival times are independent, identically distributed exponential random variables. ($M = \text{Markov}$)
- $b = M \Rightarrow$ the service process is Poisson and the interdeparture times are independent, identically distributed exponential random variables.
- $A$ or $b = G \Rightarrow$ times are independent, identically distributed general random variables.
- $A$ or $b = D \Rightarrow$ times are deterministic, i.e., fixed times

Examples:

- $M/M/1/\infty/\infty$ (Ideal router output port)
- $M/M/1/S/\infty$ (Real-finite-buffer router output port)
- $M/M/S/S/\infty$ (Circuit Switch)
Network Performance Evaluation: M/M/1

No limitation on buffer size means that the arrival rate is independent of state or \( \lambda_k = \lambda \)

Only one server means that the service rate is independent of state or \( \mu_k = \mu \)

Solving for the state occupancy probabilities

\[
P[K=k] = \rho^k (1-\rho)
\]

*With* \( \rho = \frac{\lambda L}{C} = \frac{R_{in}}{C} \)

The expected number in the systems is

\[
E[K] = \frac{\rho}{1-\rho}
\]

and the variance is

\[
Var[K] = \frac{\rho}{(1-\rho)^2}
\]

For M/M/1, if the load is greater than 1 then the systems is not stable and the buffer occupancy grows without bound.
Network Performance Evaluation: M/M/1/N

Only one server means that the service rate is independent of state or $\mu_k = \mu$

The limitation on system size means that the arrival rate is dependent of state or

$\lambda_k = \lambda$ for $k < N$

$\lambda_k = 0$ for $k \geq N$

Arrivals to a full system are blocked so there can be no arrivals to a full system.

Network Performance Evaluation: M/M/1/N

Solving for the state occupancy probabilities

$$P[k] = \frac{(1 - \rho)\rho^k}{1 - \rho^{N+1}} \text{ for } k \leq N$$

$$P[k] = 0 \text{ for } k > N$$
The Quality of Service (QoS) metric in this case is the probability of blocking.

For a M/M/1/N queue the blocking probability is given by

\[
P_B = \frac{(1 - \rho)\rho^N}{1 - \rho^{N+1}}
\]

Design Problem: Given \( P_B \) and \( \rho \) find \( N \)
(recommend constructing a spreadsheet)

Network Performance Evaluation: M/M/1/N

\( \lambda = 950 \) \& \( E[L] = 1000 \) bits, \( C = 1\text{Mb/s} \) \( \rightarrow \) Theory \( P_B = 0.23 \) Simulated \( P_B = 0.219 \)

What is going on during this time?

\( N = 3 \)
The limitation on system size means that the arrival rate is dependent of state or

\[ \lambda_k = \lambda \quad \text{for } k < N \]
\[ \lambda_k = 0 \quad \text{for } k \geq N \]

Arrivals to a full system are blocked so there can be no arrivals to a full system.

Multiple servers means that

\[ \mu_k = k\mu \quad \text{for } k \leq S \]
\[ \mu_k = S\mu \quad \text{for } k > S \]
This model is difficult to solve in general. The case of special interest is $S=N$: the M/M/N/N case. This case models a circuit switch system with $N$ transmission facilities. A call arriving to the system with all transmission facilities busy is blocked.

$\begin{array}{c}
\text{Call Arrivals} \\
\xrightarrow{\text{No Buffer}} \\
1 \\
N
\end{array}$

Network Performance Evaluation: Erlang B formula

Solving for the state occupancy probabilities

$P[k] = \frac{\rho^k}{\sum_{n=0}^{N} \rho^n n!} \quad k=0...N$

and

$P_B = P[k = N] = \frac{\rho^N}{\sum_{n=0}^{N} \rho^n n!}$

Relationship among $P_B$, $N$, $\rho$ found using provided table or web Erlang B calculator.