Networks: Tutorial 05

Shinnazar Seytnazarov, PhD

Innopolis University

s.seytnazarov@innopolis.ru

January 31, 2022

Topic of the lecture

- Socket Programming with UDP and TCP
- Transport-layer services
- Multiplexing and demultiplexing
- Connectionless transport: UDP
- Principles of reliable data transfer

Topic of the tutorial

- Performance Evaluation
- Performance Terms
- Computer Network Performance Metrics
- Performance Evaluation Techniques
  - Workload Characterization
  - Simulation Models
  - Analytic Models
- Empirical Measurement Studies
  - What to measure?
  - Choice of measurement tools
  - The Design of Measurement Experiments

Performance Evaluation

- Historically, performance evaluation was initially concerned with computer systems.

- During the 1970’s and 1980’s, computer system performance evaluation emerged as an essential component of Computer Science
  - Due to rapid and concurrent advancements in computer hardware and operating systems.

- The resultant increased complexity of modern computer systems made understanding and evaluating computer systems more difficult.

Performance Evaluation

- Performance evaluation is the application of the scientific method to the study of computer systems.

- Viewed as distinct from computer system design, the goal of performance evaluation is to determine the effectiveness and fairness of a computer system that is assumed to work correctly.

- Performance evaluation techniques have been developed to accurately measure the effectiveness with which computer system resources are managed while providing service that is fair to all customer classes.

Topic of the tutorial

- Performance Evaluation
- Performance Terms
- Computer Network Performance Metrics
- Performance Evaluation Techniques
    - Workload Characterization
    - Simulation Models
    - Analytic Models
- Empirical Measurement Studies
    - What to measure?
    - Choice of measurement tools
    - The Design of Measurement Experiments

Bandwidth

- Bandwidth: Number of bits that can be transmitted over a certain time – typically per unit time.

Some people also refer it to the spectrum

Transmission Time

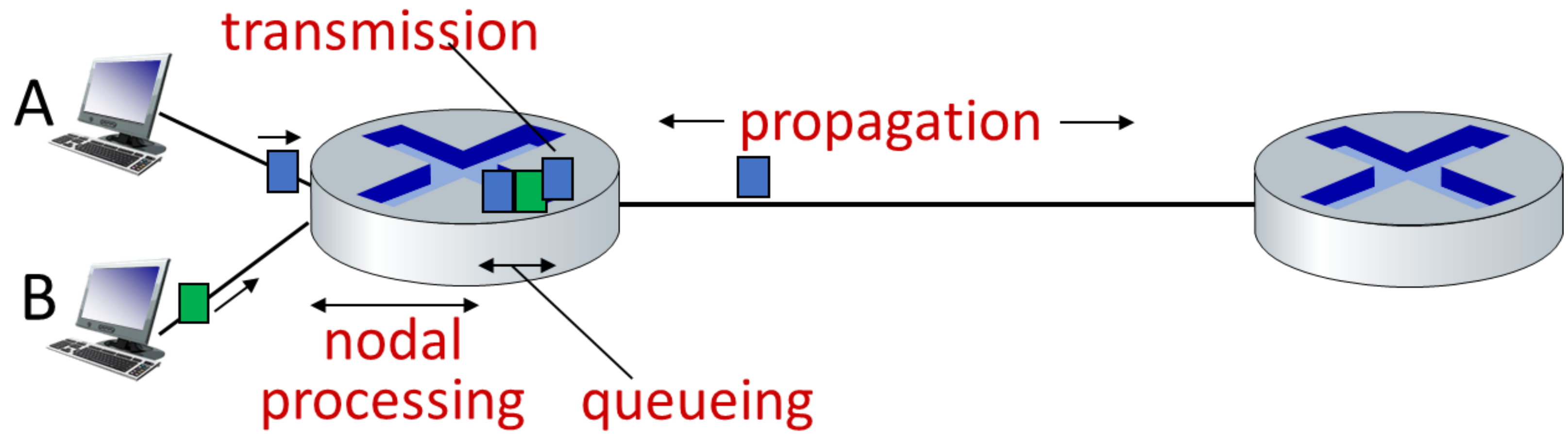
- It is a function of bandwidth

- If bandwidth is B, transmission time is 1/B.

- For Example:
  - If bandwidth is 10 Mbps, then transmission time is:

$$1/1000000 = 10^{-6} = 1\ \mu s.$$

Packet delay: four sources



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

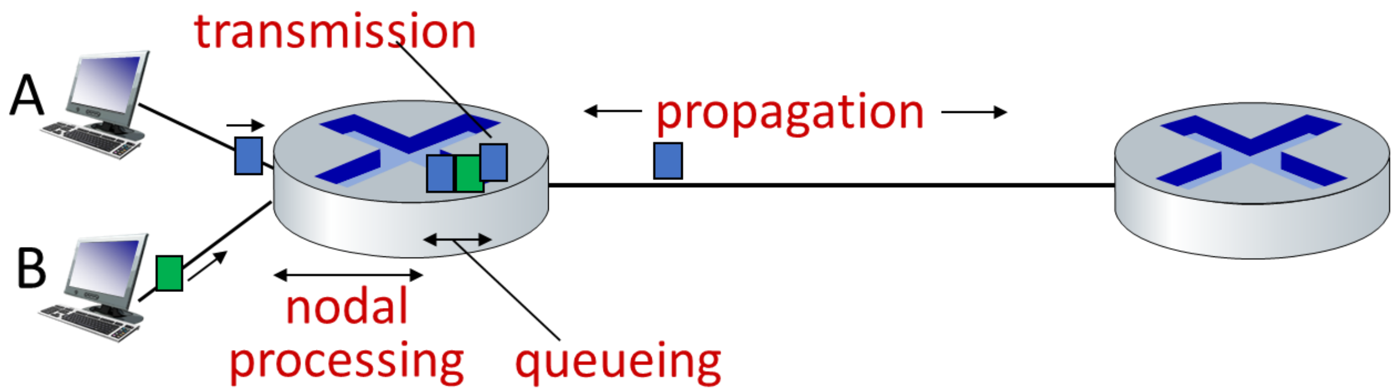
d_{proc}: nodal processing

- check bit errors
- determine output link
- typically < microsecs

d_{queue}: queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

Packet delay: four sources



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

d_{trans}: transmission delay:

- L: packet length (bits)
- R: link *transmission rate (bps)*
- $d_{trans} = L/R$

d_{prop}: propagation delay:

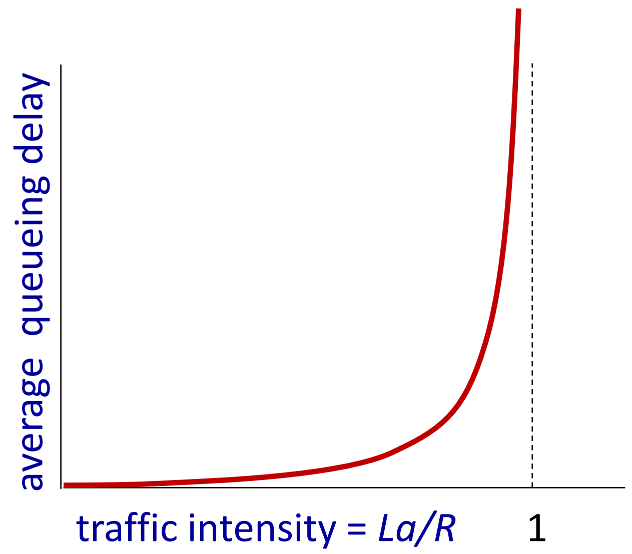
- d: length of physical link
- s: propagation speed (~2×10^8 m/sec)
- $d_{prop} = d/s$

d_{trans} and d_{prop} *very* different

Packet queueing delay (revisited)

- *a:* average packet arrival rate
- *L:* packet length (bits)
- *R:* link bandwidth (bit transmission rate)

$$\frac{L \cdot a}{R} : \quad \frac{\text{arrival rate of bits}}{\text{service rate of bits}}$$ *"traffic intensity"*



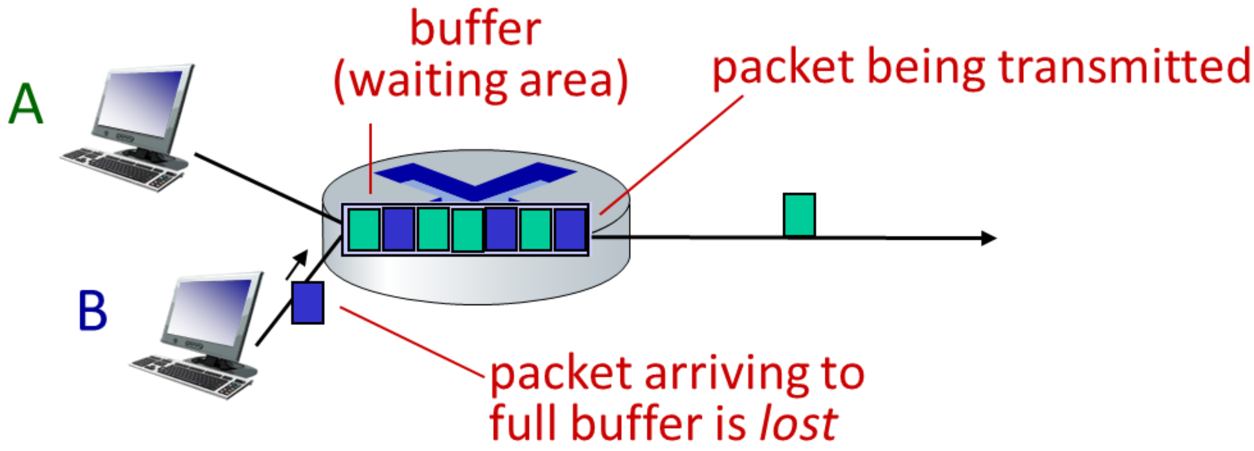
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving is more than can be serviced - average delay infinite!


La/R ~ 0


La/R -> 1

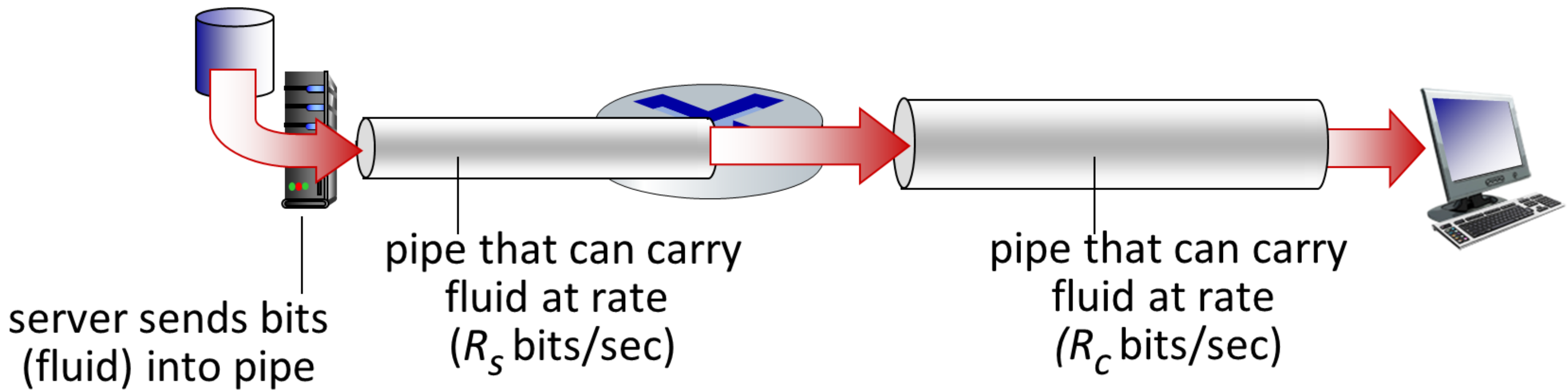
Packet loss

- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



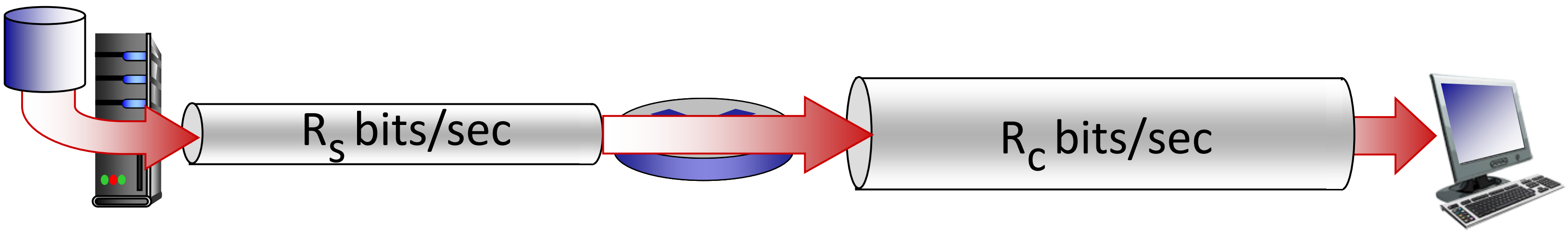
Throughput

- *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

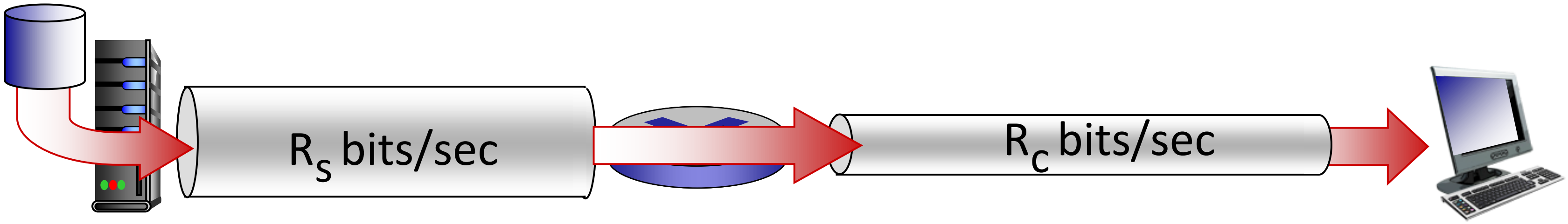


Throughput

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



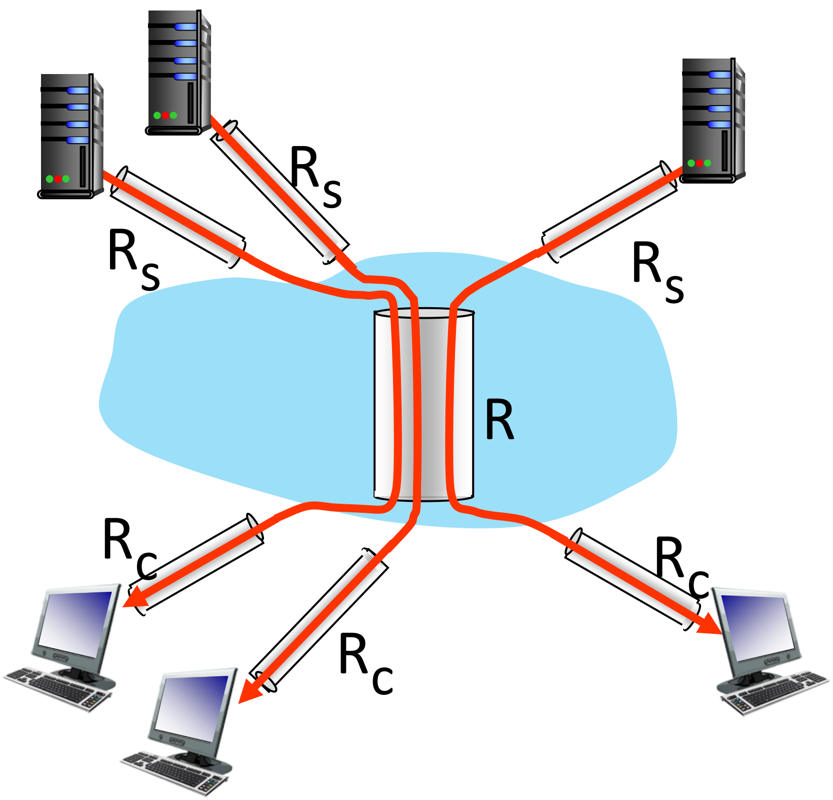
$R_s > R_c$ What is average end-end throughput?



bottleneck link

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

Throughput: network scenario



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

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

Propagation Delay

- Propagation delay is the length of time taken for the quantity of interest to reach its destination.

- It can be computed as the ratio between the link length and the propagation speed over the specific medium.

- Propagation delay = distance/speed of signal.

Queuing Delay

- At each intermediate node or router, a packet is queued.
- Thus, it has to wait prior to transmission.
- How long does it have to wait?
  - Dependent on the load over the network

Latency

- How long does a packet take to go from one host to another.

Latency = Propagation Delay +
Queueing Delay +
Transmission Delay

Also called "Delay"

Round Trip Time

- Packet is sent from sender to receiver.
- Receiver sends ACK (assume immediately) to sender.
- Total time delay incurred between the instance the packet is set to the time the ACK is received.
- **Note:**
  - If forward delay = backward delay, RTT = 2 * Latency (typically assumed – although not always accurate).

Topic of the tutorial

- Performance Evaluation
- Performance Terms
- Computer Network Performance Metrics
- Performance Evaluation Techniques
  - Workload Characterization
  - Simulation Models
  - Analytic Models
- Empirical Measurement Studies
  - What to measure?
  - Choice of measurement tools
  - The Design of Measurement Experiments

Computer Network Performance Metrics

- **Metric:** a descriptor used to represent some aspect of a computer network's performance.

- For computer networks, metrics can capture performance at multiple layers of the protocol stack.

- For Example:
  - UDP throughput
  - IP packet round trip time
  - MAC layer channel utilization

(Data Unit)		
Data	Application	Layer-7
Data	Presentation	Layer-6
Data	Session	Layer-5
Segments	Transport	Layer-4
Packets	Network	Layer-3
Frames	MAC	Layer-2
Bits	Physical	Layer-1

OSI Layers

Source: http://www.rfwireless-world.com/Terminology/PHY-vs-MAC.html

Sample Performance Measures

Category	Metric	Units
Productivity	throughput effective capacity	Mbps

Sample Performance Measures

Category	Metric	Units
Productivity	throughput effective capacity	Mbps
Responsiveness	delay round trip time queue size	milliseconds packets

Sample Performance Measures

Category	Metric	Units
Productivity	throughput effective capacity	Mbps
Responsiveness	delay round trip time queue size	milliseconds packets
Utilization	channel utilization	percentage of time busy

Sample Performance Measures

Category	Metric	Units
Productivity	throughput effective capacity	Mbps
Responsiveness	delay round trip time queue size	milliseconds packets
Utilization	channel utilization	percentage of time busy
Losses	packet loss rate frame retries	loss percentage

Sample Performance Measures

Category	Metric	Units
Productivity	throughput effective capacity	Mbps
Responsiveness	delay round trip time queue size	milliseconds packets
Utilization	channel utilization	percentage of time busy
Losses	packet loss rate frame retries	loss percentage
Buffer problems	queue overflow buffer underflow	packet drops rebuffer events

Topic of the tutorial

- Performance Evaluation
- Performance Terms
- Computer Network Performance Metrics
- Performance Evaluation Techniques
  - Workload Characterization
  - Simulation Models
  - Analytic Models
  - Empirical Measurement Studies
    - What to measure?
    - Choice of measurement tools
    - The Design of Measurement Experiments

Performance Evaluation Techniques

- Workload characterization for computer networks involves the design and choice of traffic types that provide the inputs for computer network performance evaluation.

Performance Evaluation Techniques

- Workload characterization for computer networks involves the design and choice of traffic types that provide the inputs for computer network performance evaluation.

- Performance measures of computer networks are all dependent to some extent on the input workload, the network topology and the choices in controlled parameters or network default settings.

Performance Evaluation Techniques

- Workload characterization for computer networks involves the design and choice of traffic types that provide the inputs for computer network performance evaluation.

- Performance measures of computer networks are all dependent to some extent on
  - the input workload,
  - the network topology and
  - the choices in controlled parameters or network default settings.

- An evaluation study of a computer network seeks to determine the values for network performance indices under a given traffic workload and network configuration.

Typical Network Traffic Types

- Web Traffic between a Browser and an Internet Server
- Long-Lived File Transfers
  - FTP downloads
- Multimedia Streaming
  - Video clip downloads (UDP and/or TCP)
  - Audio VOIP (Voice Over IP)
- Peer-to-Peer Exchanges
  - Concurrent downloads and uploads

Performance Evaluation Methods

- Discrete Event Simulation
  - Simulating the real systems
- Analytic Modeling
  - Modeling the real system with mathematical equations
- Measurement Studies
  - Empirical measurement of real networks

Discrete Event Simulation

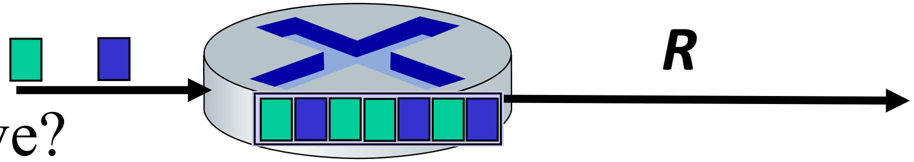
- A simplified model of the system and its load, implemented in software
- Event-driven simulation defines a network in terms of states and transitions where certain events trigger certain transitions
- Attempts to reproduce the behavior of the system in the time domain.
- Simulation is essentially a numeric solution that utilizes systems of equations and data structures to capture the behavior of the simulated network in terms of logical conditions.
- The performance of interest is measured as on a real system
- Advantages
  - Reproducible
  - It is often easier than a measurement study
  - Measurement side-effects are usually not present

Discrete Event Simulation

Event driven simulation example

- Let's take a simple router with one incoming and one outgoing link

  Given parameters:
  - ***max_qsize*** – a maximum queue size (packets)
  - ***R*** – outgoing link speed (bits/sec)
  - ***L*** – packet size (bits)



- What kind of events we may have?
  - ***Arrival*** – a new packet arrival into a router

    Which operations to complete upon each arrival
    - Upon arrivals a new packet is added into queue **if queue isn't already full**
    - Check whether router is transmitting now; if not transmit head-of-line (HOL) packet
  - ***Departure*** – a packet leaves the router when its transmission finishes
    - A packet is removed from queue
    - Another packet transmission starts if there is any in queue

Discrete Event Simulation

- Event driven simulation example

```
# arrange arrival events, ti can be generated from some distr.
arrivals = [(t1, “arr”), (t2, “arr”),.., (tn, “arr”)]
# in the beginning we only have arrival events
events = arrivals
# sort events in increased order of ti
events.sort()
t = 0 # current time in simulator
T = 1 hour # simulation duration
while t ≤ T and events:
    ti, ei = events[0]
    t = ti # current time is updated
    events = events[1:] # event list is updated
    if ei is “arr”:
        if qsize < max_qsize:
            qsize += 1
        if qsize == 1: # queue was empty before
            tdep = t + L/R
            new_event = (tdep,”dep”)
            events.append(new_event)
```

Discrete Event Simulation

- Event driven simulation example (cont.)

```
while t ≤ T and events:
    t_i, e_i = events[0]
    t = t_i # current time is updated
    events = events[1:] # event list is updated
    if e_i is "arr":
        if qsize < max_qsize:
            qsize += 1
        if qsize == 1: # queue was empty before
            t_dep = t + L/R
            new_event = (t_dep,"dep")
            events.append(new_event)
    else: # departure event
        qsize -= 1
        if qsize > 0:
            t_dep = t + L/R
            new_event = (t_dep,"dep")
            events.append(new_event)
    events.sort()
```

Discrete Event Simulation

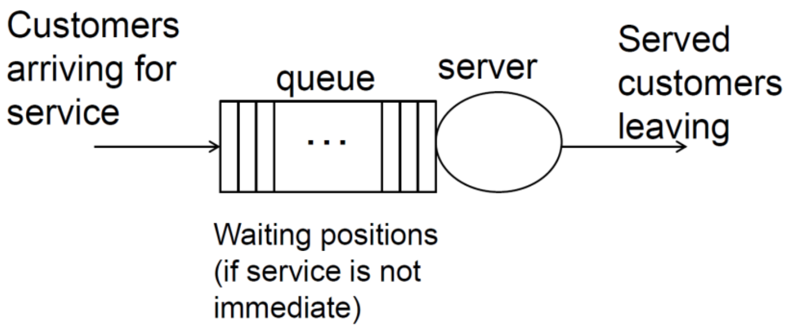
- Which performance metrics can we evaluate in example?
  - Queueing delay
    - can be measured upon departures
  - Packet loss rate
    - is recorded upon new arrivals when qsize = max_qsize
  - Queue length
    - qsize can be recorded upon arrivals and departures
  - Throughput
    - The same as **input traffic rate** if it is <= R
    - Equals to R, otherwise

Analytic Models

- A mathematical model of the system is analyzed numerically.
  - This is viewed by some as a special form of simulation.
  - It is often much faster than a simulation, but sometimes requires wild assumptions to be made in order for the numerical procedures to be applicable
- Analytic models of computer networks usually start with a network of queues model and develop a system of equations that may or may yield a closed form solution.
- Analytic models of computer networks tend to be stochastic models built on the theory of stochastic processes associated with independent random variables.
- There is a special subject called Queueing theory

Analytic Models

- Queueing Problem
  - Queueing system = queue + server



- Queueing models are characterized by the following 6 features
  - Arrival pattern of customers: eg) exponential, deterministic, general, etc.
  - Service pattern: eg) exponential, deterministic, general, etc.
  - Queue discipline: first come first service (FCFS), last come first service (LCFS), random service selection (RSS), priority schemes (PR)
  - System capacity: size of queue, eg) finite, infinite
  - Number of service channels

Analytic Models

Kendall's Notation

- A / B / X / Y / Z
- Standard symbols for A and B
  - M : exponential
  - D : deterministic
  - E_k : Erlang type k
  - G : general
- Standard symbols for X and Y
  - 1, 2, 3, …., ∞
- Standard symbols for Z
  - FCFS, LCFS, RSS, PR
- If y = ∞ or Z = FCFS, then omitted
  - Ex: D/M/3/∞/FCFS => D/M/3/

- Steady state
  - State of the system after it has been in operation for a long time
- We want to know how many servers are needed in M/M/1/K queue to get steady state
  - λ : average rate of arriving customers (exp. dist.)
  - μ : average rate of serving customers (exp. dist.)
  - 1 server and queue size is K
- $\rho = \lambda/\mu$: traffic congestion or intensity

Analytic Models

Traffic Congestion (intensity)

- If $\rho>1$
  - as time goes on, the queue gets bigger
  - Queue never settles down
- If $\rho=1$
  - Unless arrivals are deterministic and perfectly scheduled, the randomness will not allow for a steady state
- If $\rho<1$
  - Steady state exists

Topic of the tutorial

- Performance Evaluation
- Performance Terms
- Computer Network Performance Metrics
- Performance Evaluation Techniques
  - Workload Characterization
  - Simulation Models
  - Analytic Models
- Empirical Measurement Studies
  - What to measure?
  - Choice of measurement tools
  - The Design of Measurement Experiments

Empirical Measurement Studies

The planning phase objectives of an empirical measurement are:

1. To decide what to measure?
2. To choose the measurement tools
3. To design the experiments.

Empirical Measurement Studies

- Network measurements can be either active or passive.
- Active measurement involves purposely adding traffic to the network workload specifically to facilitate the measurement
- An example of a passive measurement tool is a network sniffer running in promiscuous mode to collect information about all packets traversing a network channel.

What to Measure?

- The overall objective of the computer network measurement study guides the choice of performance indices to be measured.

- Due to the large data volume associated with network traffic, measurement of computer networks often involves filtering of data or events (e.g., It is common for network measurement tools to only retain packet headers for off-line analysis).

- When the measurement strategy involves probabilistic sampling, the duration of the experiments is determined using confidence interval techniques.

Network Measurement Tools

- While hardware probes provide the best quality measurements, they are expensive and not always available.

- Network software measurement tools provide ‘hooks’ within the network layering software to capture and store network measurement data.

Choice of Measurement Tools

Key issues in the usability of network measurement tools are:

1. Tool location
2. Interference or bias introduced by the tool.
3. Accuracy of the tool.
4. Tool resolution

   - This has become a problem with respect to the granularity of system clocks relative to the speed of modern high speed network links.

The Design of Measurement Experiments

Measurement Experiments are divided into two major categories:

1. Live measurements
   - With live empirical studies, the objective is to measure the performance of the computer network while it is handling real traffic.
   - The advantage of this type of study is that the measurement involves a real workload.
   - Disadvantage of live traffic measurement is that reproducibility of the exact same traffic workload is usually not possible.

The Design of Measurement Experiments

2. Controlled-traffic measurements

- Traffic generator tools or traffic script files provide repeatable, controlled traffic workloads on the network being measured.
- Controlled-traffic workloads are chosen when the goal of the performance study is to evaluate the impact of different versions of a network component, strategy or algorithm on network performance.
- Controlled, repeatable traffic makes it easier to conduct cause-and-effect performance analysis.
- One difficulty with controlled-traffic is being confident in the accuracy of the traffic generator tool and the ability to conduct measurement experiments where the traffic workload choices are adequately varied to provide representative, robust network performance evaluation.

Acknowledgements

- Most part of this tutorial was prepared by M.Fahim, G.Succi, and A.Tormasov

Reference

- This tutorial is based on the lecture materials of Performance Evaluation of Computer Networks by Prof. Bob Kinicki and relevant material over the internet.