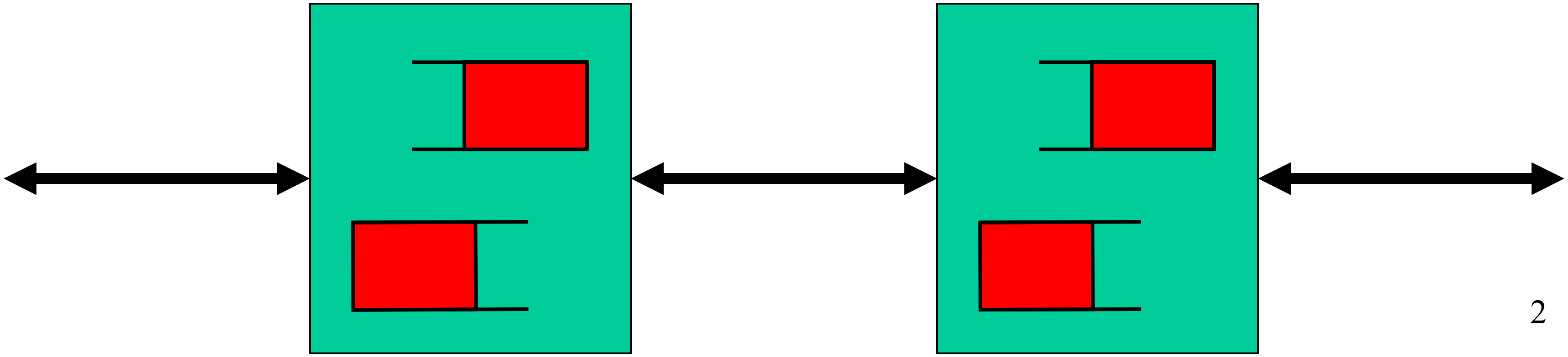
Application: Active measurements - delay

Prof. Anja Feldmann, Ph.D.
Dr. Nikolaos Chatzis

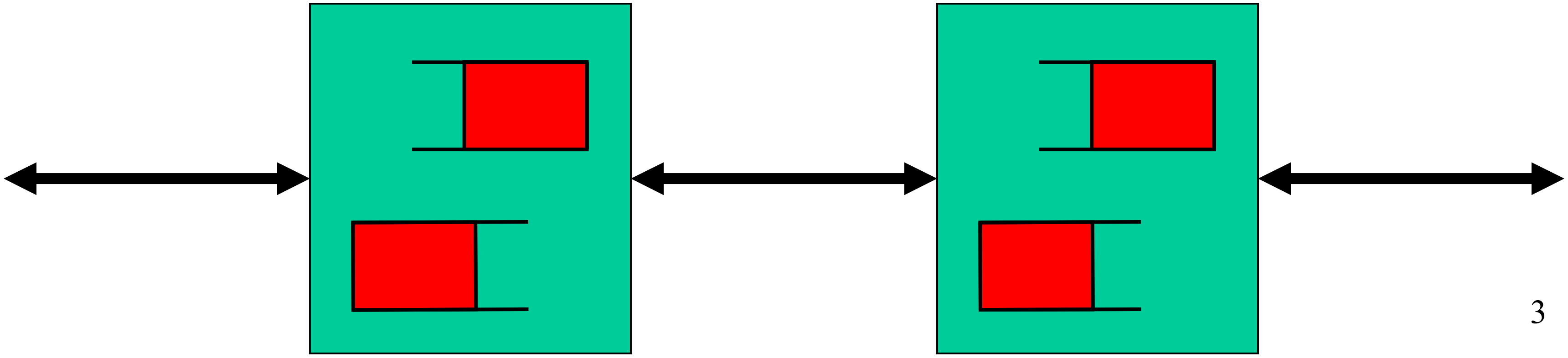
Network delay

- Components:
  - Transmission delay: time to place bits on the medium
  - Propagation delay: time for bits to propagate on the medium
  - Processing delay: additional time for processing packet, including buffering



Network delay

- Packet size: L
- Link capacity: c
- Link length: d
- Signal propagation speed: v
- Processing delay: e
- Packet delay through a link: $L/c + d/v + e$



Tools: ping

- Internet Control Message Protocol (RFC792)
- Goals: report IP connectivity problems, e.g. host reachability, unreachable ports, fragmentation,…
- Ping: ICMP-echo request-reply
  - Advantage: wide availability (in principle, any IP address)
  - Drawbacks
    - pinging routers is evil! (except for troubleshooting)
      - load on host part of router: scarce resource, slow
      - delay measurements very unreliable/conservative
      - availability measurement very unreliable: router state tells little about network state
      - pinging hosts: ICMP not representative of host performance

ping

```
ping [-LRUbdfnqrvVaAB] [-c count] [-i interval] [-l preload] [-p
   pattern] [-s packetsize] [-t ttl] [-w deadline] [-F flowlabel] [-I
   interface] [-M hint] [-Q tos] [-S sndbuf] [-T timestamp option] [-W
   timeout] [hop ...] destination
```

OPTIONS

-c count *Stop after sending count ECHO_REQUEST packets. With deadline option, ping waits for count ECHO_REPLY packets, until the timeout expires.*

-f *Flood ping. For every ECHO_REQUEST sent a period ``.'' is printed, while for ever ECHO_REPLY received a backspace is printed. This provides a rapid display of how many packets are being dropped. If interval is not given, it sets interval to zero and outputs packets as fast as they come back or one hundred times per second, whichever is more. Only the super-user may use this option with zero interval.*

-i interval *Wait interval seconds between sending each packet. The default is to wait for one second between each packet normally, or not to wait in flood mode. Only super-user may set interval to values less 0.2 seconds.*

-I interface address *Set source address to specified interface address. Argument may be numeric IP address or name of device.*

ping

```
> ping -c 10  www.tu-berlin.de

PING www.tu-berlin.de (130.149.7.201) 56(84) bytes of data.
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=1 ttl=126 time=0.583 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=2 ttl=126 time=0.579 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=3 ttl=126 time=0.531 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=4 ttl=126 time=0.594 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=5 ttl=126 time=0.634 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=6 ttl=126 time=0.570 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=7 ttl=126 time=0.535 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=8 ttl=126 time=0.475 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=9 ttl=126 time=0.545 ms
64 bytes from www.TU-Berlin.DE (130.149.7.201): icmp_seq=10 ttl=126 time=0.491 ms

--- www.tu-berlin.de ping statistics --- 10 packets transmitted, 10 received, 0%
    packet loss, time 10061ms rtt min/avg/max/mdev = 0.475/0.553/0.634/0.053 ms
```

Statistics with R

- What is R?
  - Open-source statistical environment
  - A framework for data analysis
  - A programming language
  - Free software
- What can you do with R?
  - Data manipulation:
    - Loading data
    - Selecting and modifying
    - Computing functions on data, e.g. statistics

Statistics with R: data

- Loading data
  - vector.small = read.table(“ping-10“,nrows=10)
  - vector.medium = read.table(“ping-100“,nrows=10)
  - vector.large = read.table(“ping-1000“,nrows=10)
- Playing with data
  - Vector Length: length(vector)
  - Element n: vector[n]
  - First k elements: vector[1:k]
  - Last k elements: vector[(length(vector)-k+1): length(vector)]
  - All elements larger than x: vector[vector>x]
  - Trimming: mean(vector,trim=1/x)

Statistics with R: basic stats

- Summary statistics:
  - Mean: mean(vector)
  - Median: median(vector)
  - Standard deviation: sd(vector)
  - Summary: summary(vector)
- Plotting densities:
  - Basic plot: plot(table(vector))
  - Histogram: hist(vector,x)
  - CDF: plot(ecdf(vector))
  - Quantile plot: boxplot(vector)

Statistics with R: distributions

- Normal distribution: centered limit of most distributions
- Generating a normal distribution with same mean and standard deviation as data:
    - x.normal= rnorm(n=1000,m=mean(vector),sd=sd(vector))
    - Impact of n on the distribution
  - Plotting the CDF:
    - hist(x.normal)
    - plot(ecdf(x.normal))

Statistics with R: distributions (2)

- Poisson distribution: arrivals of large numbers of independent sources
  - x.poisson = rpois(n=1000,lambda=mean(vector))
  - hist(x.poisson)
  - plot(ecdf(x.poisson))

Statistics with R: distributions (3)

- Normal and Poisson have limited variations: nothing like delay measurements
- Heavy-tailed distributions:
  - Weibull: distribution of failures (extreme events)
    - x.weibull=rweibull(n=1000,shape=1.8,scale=mean(vector))
    - hist(x.weibull)
    - plot(ecdf(x.weibull))

Statistics with R: matching delay

- From what distribution do delay come?
  - Plotting sample quantiles (e.g. delay measurements) against theoretical quantiles
  - If distributions are similar, quantiles should fall on the diagonal
  - Are delay measurements normal or heavy-tailed?
    - qqplot(vector,x.norm)
    - qqplot(vector,x.weibull)
    - qqplot(vector,x.poisson)