Channel Coding

ARQ and Reliability

November 22, 2021

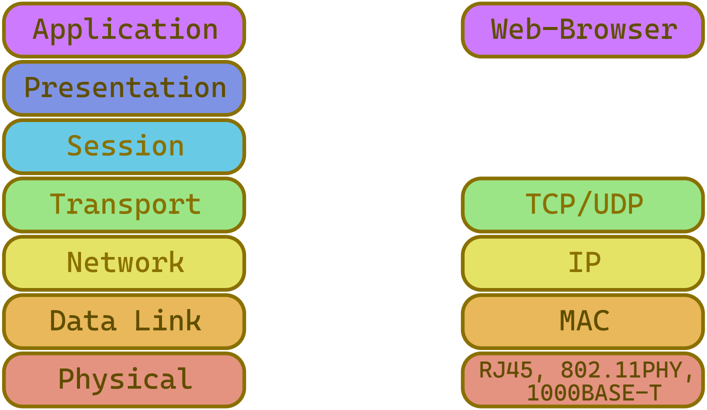
Recall

The Open Systems Interconnection model (OSI)

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

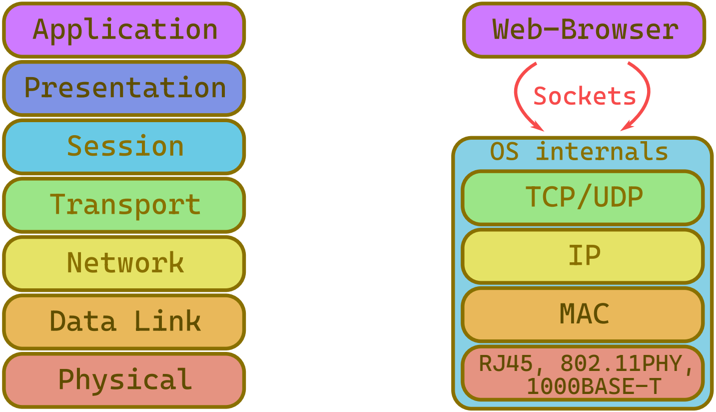
Recall

The Open Systems Interconnection model (OSI)



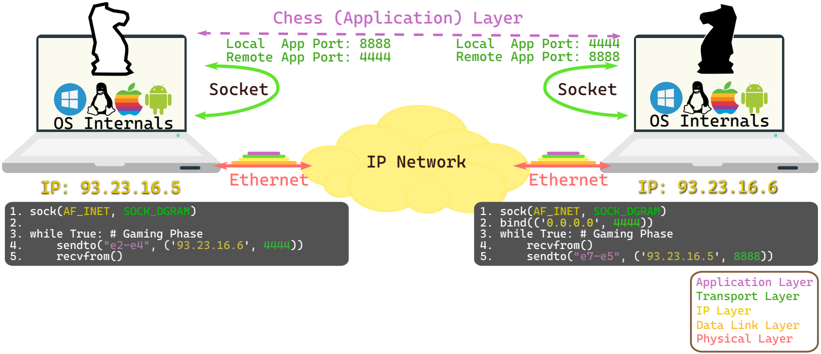
Recall

The Open Systems Interconnection model (OSI)



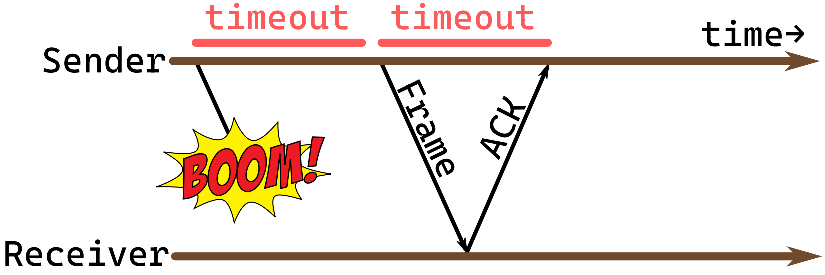
Sockets

Network stack from users perspective

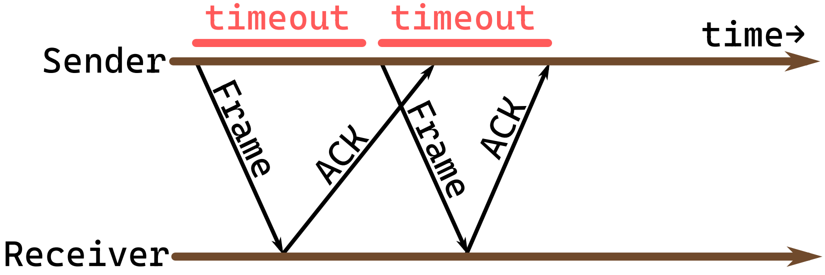


ARQ

Why do we need ARQ?

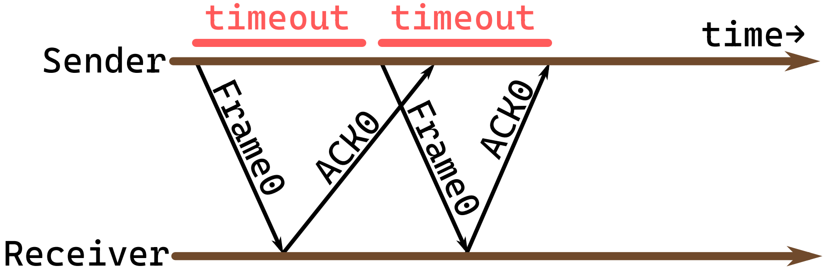

timeout
timeout
time→
Sender
Frame
ACK
BOOM!
Receiver

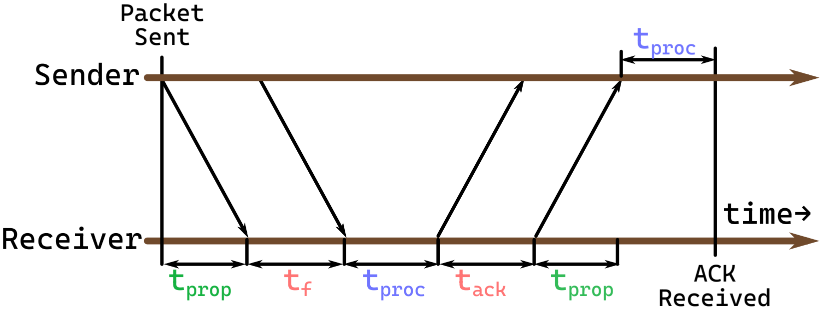
ARQ
Timeout problem



ARQ

Sequence number. Stop-and-wait.





$$t_0 = 2t_{prop} + t_f + 2t_{proc} + t_{ack} \approx RTT + 2t_{proc} + \frac{n_f + n_{ack}}{Rate}$$

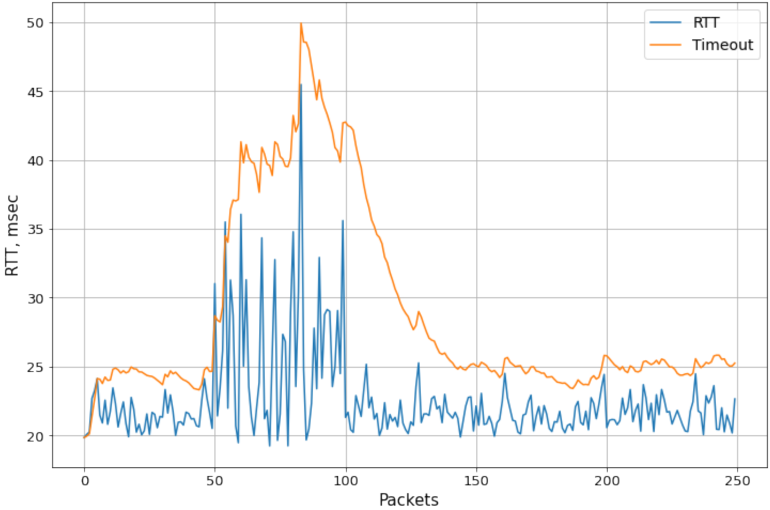
- Which timeout should we choose?
    - Not too big
    - Not too small
- Easy to define for specific LAN. Little variation.
- Difficult over the Internet. High variation.

Simple Timeout calculation scheme[1]. Smoothed RTT + variance.

- $SRTT_{N+1} = 0.9 \cdot SRTT_N + 0.1 \cdot RTT_{N+1}$
- $Svar_{N+1} = 0.9 \cdot Svar_N = 0.1 \cdot |RTT_{N+1} - SRTT_{N+1}|$
- $Timeout_N = SRTT_N + 4 \cdot Svar_N$

[1] rfc2988

ARQ
Adaptive Timeout



- Probability of Failure[1]:

$$P_f = 1 - (1 - plr)^2$$

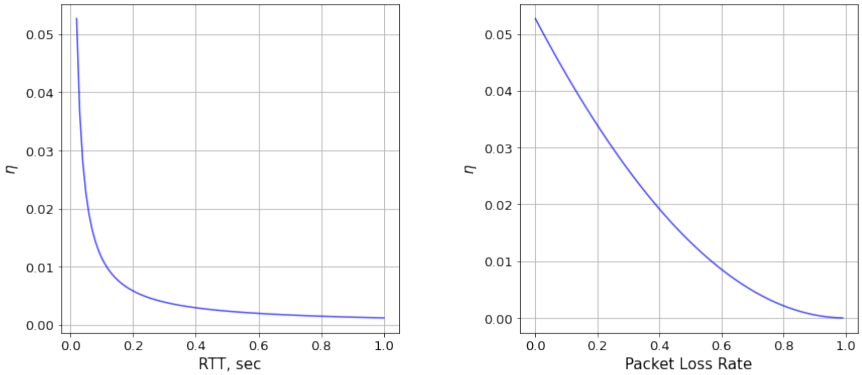
- Average total time to transmit a packet [**1**]:

$$E[t_{packet}] = t_0 + \frac{t_{out} P_f}{1 - P_f}$$

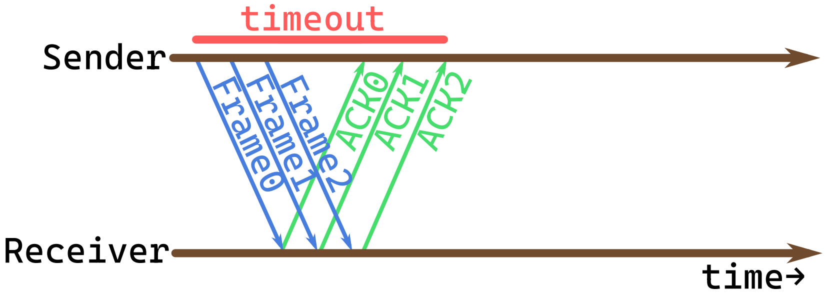
- Effective information transmission rate: $R_{eff} = \dfrac{n_f - n_{headers}}{E[t_{packet}]}$
- Associated transmission efficiency: $\eta = \dfrac{R_{eff}}{Rate}$

[1] plr stands for Packet Loss Rate

Stop And Wait
Efficiency

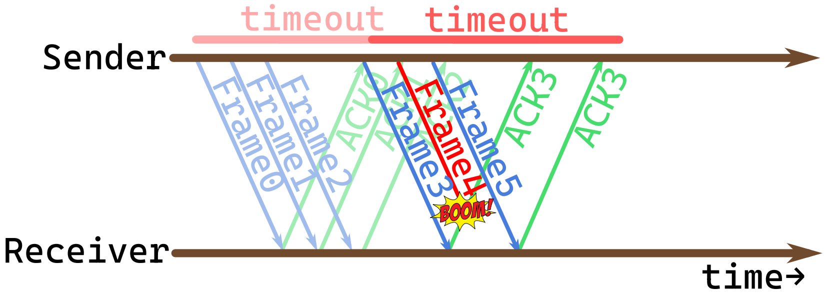


Sliding Window
Go Back N. Principle



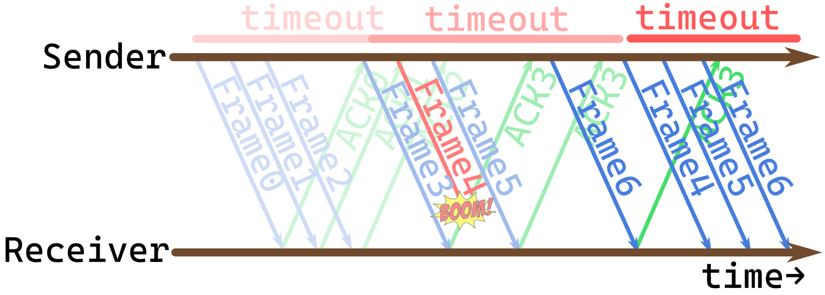
Sliding Window

Go Back N. Principle



Sliding Window

Go Back N. Principle



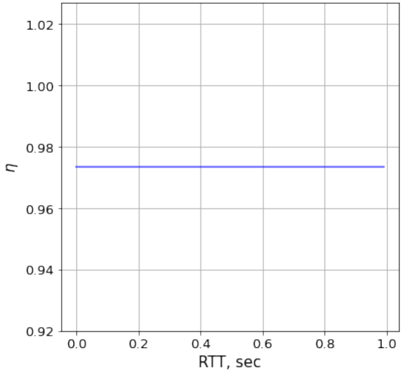
[1]Demo: 1, 2

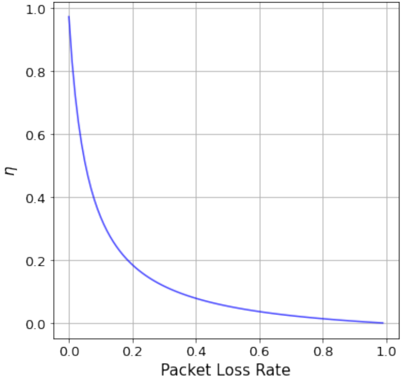
- Probability of Failure: $P_f = plr$
- Average total time to transmit a packet [**1**]. Windows size W_s should be selected so that the channel will be busy all the time.

$$E[t_{packet}] = t_f \frac{1 + (W_s - 1)P_f}{1 - P_f}$$

- Effective information transmission rate: $R_{eff} = \dfrac{n_f - n_{headers}}{E[t_{packet}]}$
- Associated transmission efficiency: $\eta = \dfrac{R_{eff}}{Rate}$

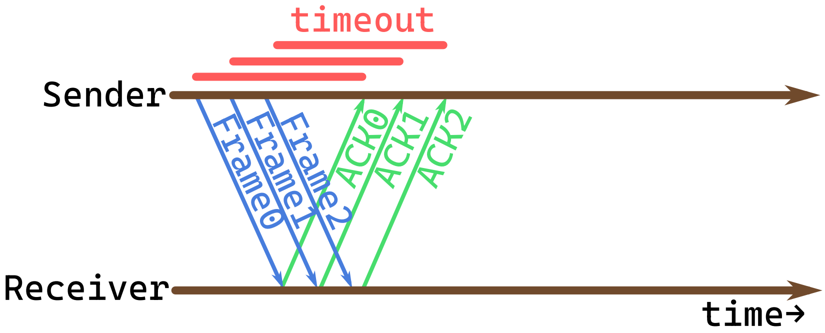
Sliding Window
Efficiency of GoBack-N





Sliding Window

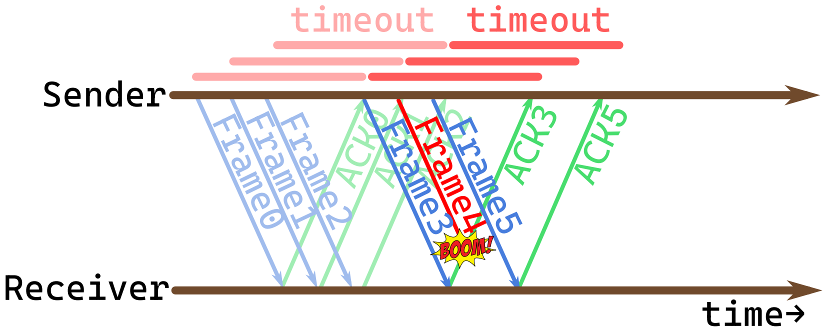
Selective Repeat. Principle



[1] Demo: 1, 2

Sliding Window

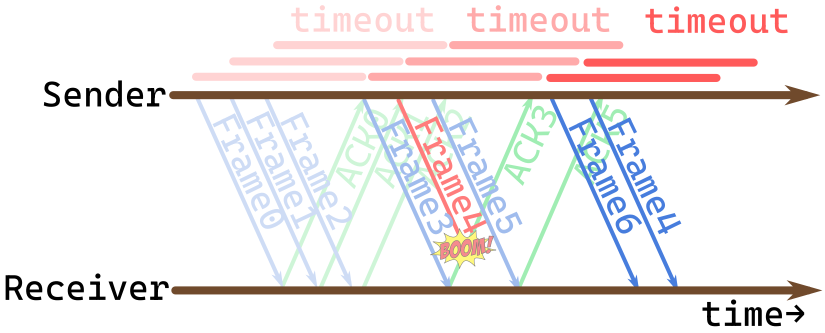
Selective Repeat. Principle



[1] Demo: 1, 2

Sliding Window

Selective Repeat. Principle



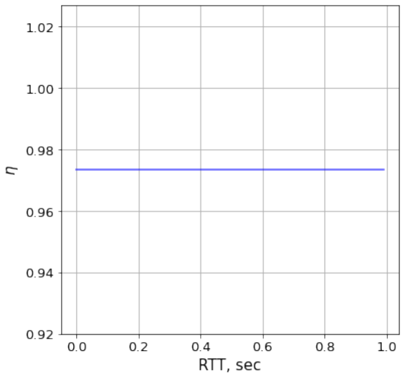
[1]Demo: 1, 2

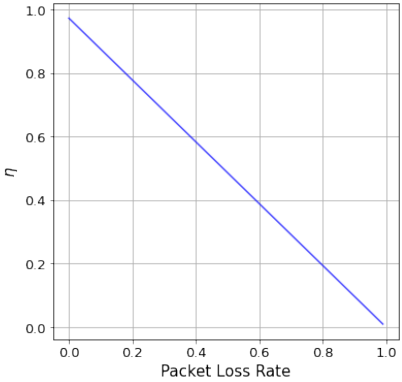
- Probability of Failure: $P_f = plr$
- Average total time to transmit a packet [**1**]. Windows size W_s should be selected so that the channel will be busy all the time.

$$E[t_{packet}] = t_f \frac{1}{1 - P_f}$$

- Effective information transmission rate: $R_{eff} = \dfrac{n_f - n_{headers}}{E[t_{packet}]}$
- Associated transmission efficiency: $\eta = \dfrac{R_{eff}}{Rate}$

Efficiency of Selective Repeat





Task 1

Echo Server

```
user@pc1:~$ ./echo-server -p 8888 --proto udp
Server is listenning on UDP port 8888....
Connection from 192.168.0.6...
Client message: Your mouth is deep as a cave!
Client message: I took it okay the first time mr dentist....
Client message: That was an echo))))
```

```
user@pc2:~$ ./echo-client -a 192.168.0.7 -p 8888 --proto udp
>>>>: Your mouth is deep as a cave!
echo: Your mouth is deep as a cave!
>>>>: I took it okay the first time mr dentist....
echo: I took it okay the first time mr dentist....
>>>>: That was an echo))))
echo: That was an echo))))
```

[1] Demo: 1, 2

Task 2

Network condition simulation

```
tc qdisc add dev eth0 root netem delay 10ms   loss 1.0%
tc qdisc add dev eth0 root netem delay 10ms   loss 10.0%
tc qdisc add dev eth0 root netem delay 100ms  loss 10.0%
tc qdisc add dev eth0 root netem delay 1000ms loss 1.0%
tc qdisc del dev eth0 root netem delay 100ms  loss 10.0%
```

References and further readings

[1] Leon-Garcia, A., & Widjaja, I. (2000). Communication networks: fundamental concepts and key architectures (Vol. 2). New York: McGraw-Hill.

[2] Computer Networking: A Top-Down Approach. / Interactive Animations

[3] TU Berlin. Computer Networks - An Animated Approach

Thanks for your attention