PES UNIVERSITY ONLINE

COMPUTER COMMUNICATION NETWORKS

Department of Electronics and Communication Engineering

COMPUTER COMMUNICATION NETWORKS

Principles of reliable data transfer:
Stop and Wait protocols

Dr. Arpita Thakre

Department of Electronics and Communication Engineering

Introduction

Packet loss is said to occur due to:

  - Corrupt packet discarded at the receiver
  - Packet was discarded at a router due to lack of buffer space
  - Packets experience long queuing delays in a router

- Reliable data transfer (RDT)
  - It is a fool-proof mechanism for overcoming packet loss
  - It requires a connection oriented approach (i.e., sender and receiver must agree to some parameters for monitoring packets using some handshaking)
  - Throughput and delay may be compromised
  - By default, UDP does not guarantee reliable data transfer
  - TCP offers reliable rata transfer (the only QoS guaranteed)

Principles of reliable data transfer

System model

- Consider two hosts A (Sender) and B (Receiver) which wish to communicate over a network reliably
- Reliable data transfer between hosts A and B is achieved when they agree to monitor the packets exchanged and notify one another if packet losses are detected
- This is accomplished by some handshaking between A and B before data packets are exchanged
- We build the principles of reliable data transfer systematically
- We start with an ideal case and incrementally add complexity to the transport layer protocol
- Hereafter, the transport layer protocol is referred to as **RDT protocol**

Overview of the stages of development of the RDT protocol

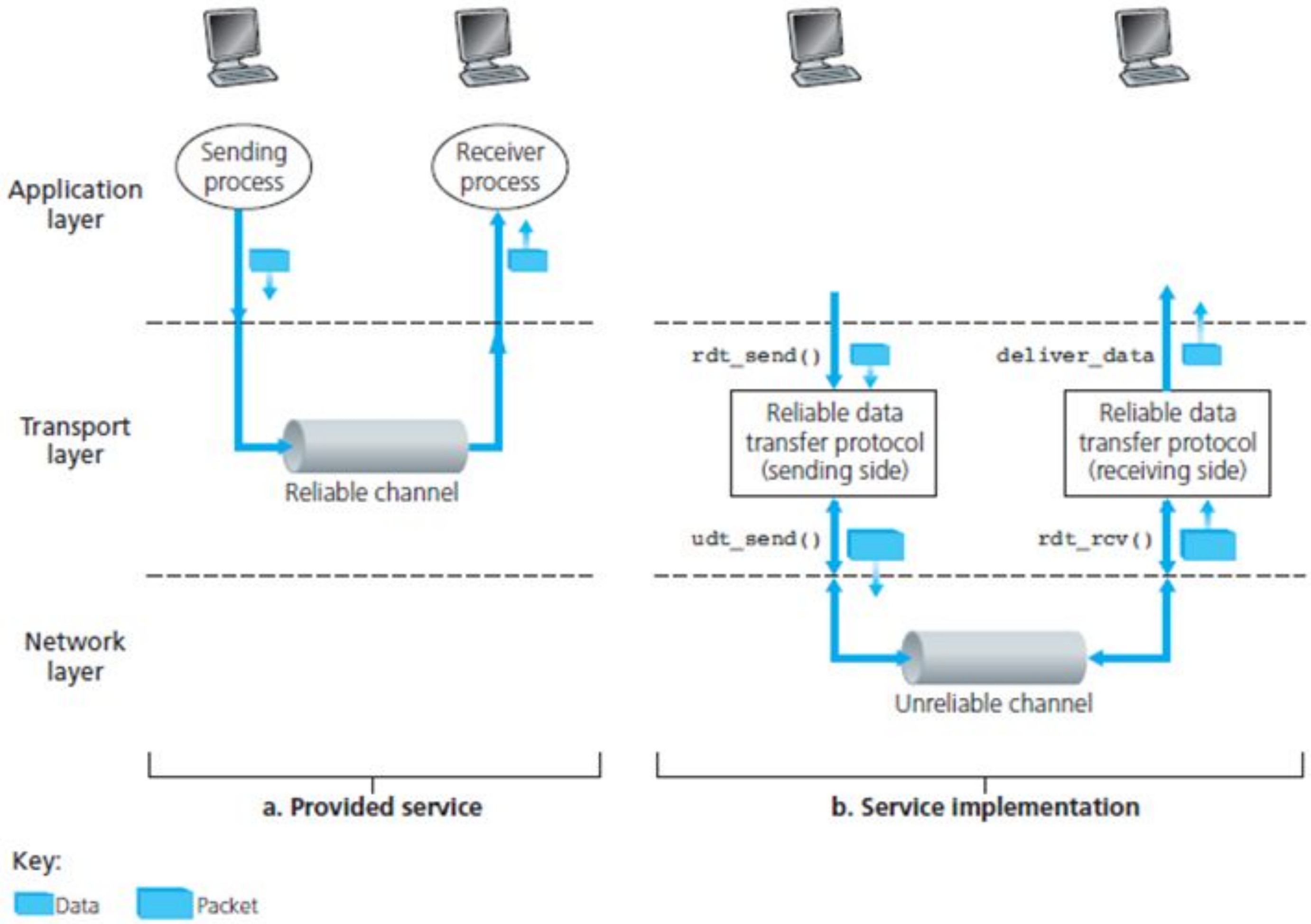
Stop and wait RDT protocols:

- Host A sends one packet at a time
- Host A waits for an acknowledgement from the host B to transmit the next packet
- We will see **four versions** of the stop and wait RDT protocols with each version addressing one limitation of its predecessor
- **Pipelining RDT protocols:**
- Host A sends multiple packets to host B at a time
- Host A waits for the acknowledgements from host B within a fixed time interval (referred to as a **Timeout**)
- Upon learning successfully delivery of the packets in previous round, the next batch of packets are transmitted

Principles of reliable data transfer

- **Pipelining RDT protocols (contd.):**
- We will see two versions of the pipelining RDT protocols (GBN and SR).
- Compared to the stop and wait RDT protocols, these pipelining RDT protocols provide better link utilization
- **Transmission control protocol (TCP):**
- It is a hybrid of the above pipelining RDT protocols but with sophistication of its own.
- TCP is robust compared the above pipelining RDT protocols.
- TCP makes host A adaptive to network congestion and packet overflow problems that may occur at host B.

Stop and wait RDT protocol – Version 1



Stop and wait RDT protocol – Version 1 (RDT1.0):

- No bit errors and no packet delays
- What should be the role of the RDT protocol in this context?
- An application in host A generates a message.
- Host A segments the message into several packets.
- Transport layer in host A inserts the source and destination port numbers (i.e., encapsulation) and passes it to network layer.
- The network layer protocol delivers the datagram to host B.

Stop and wait RDT protocol – Version 2 (RDT 2.0):

- Bit errors in packet transmissions from A to B but no packet delays.
- How should RDT1.0 be modified to handle the above issue?
- Host A introduces checksum (i.e., error detection code) into the packet before passing it to the network layer.
- Host B verifies if packet is corrupt or not.
- If packet is corrupt, then host B sends an NAK packet; otherwise host B sends an ACK packet.
- If host A receives NAK packet, it retransmits the old packet.
- If host A receives ACK packet, it retransmits next packet with new checksum.

Stop and wait RDT protocol – Version 3 (RDT2.2):

- Bit errors in two way packet transmissions but no packet delays.
- Now, it is difficult to distinguish between old packets and new packets exchanged between A and B.
- Two types of packets are used by host B (ACK and NAK) which is unnecessary given that they may get corrupted as well

- How should RDT2.0 be modified to handle the above issues?

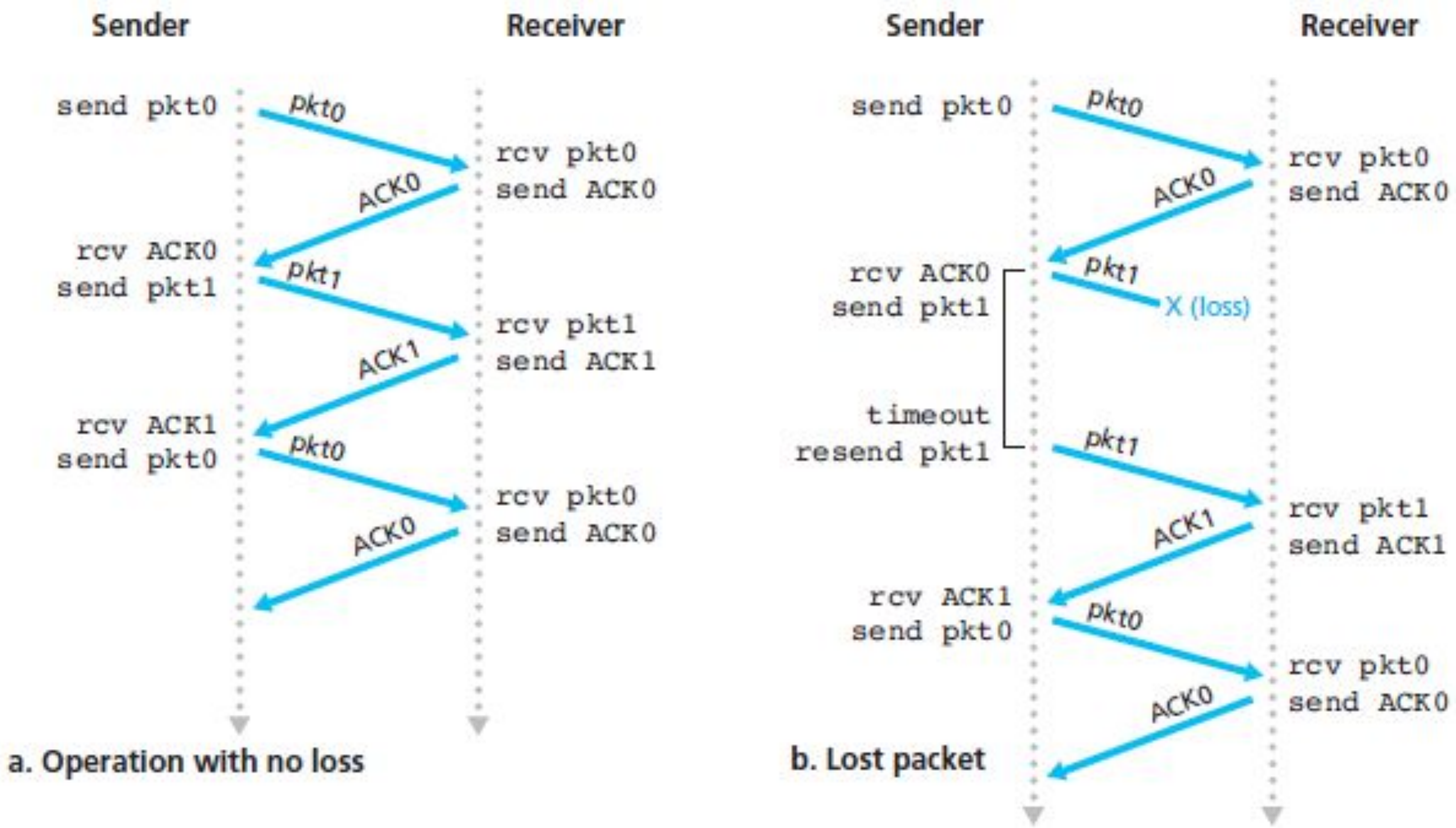
- Hosts introduce sequence numbers to identify packets (0 and 1 used alternatively) besides using the checksum to detect errors.
- Host B just uses ACK packets with sequence number (0 or 1)

- Example: Host A sent packet 0, then it deems the transmission successful only when an ACK packet with sequence number 0. Otherwise, Host A retransmits the packet 0.

Stop and wait RDT protocol – Version 4 (RDT3.0):

- Bit errors and packet delays occur in two way packet transmissions.
- Now, host A may end up waiting endlessly hoping the packet is stuck in some intermediate router’s queue
- How should RDT2.2 be modified to handle the above issues?
- Host A starts a timer as soon as it transmitted a packet using a sequence number and checksum.
- Host B replies just as in RDT2.2 depending on whether the received packet is corrupt, new or old.In case ACK packet is lost/corrupt/old, host A retransmits after timer expires.
- Otherwise, host A sends the next packet and starts timer

Principles of reliable data transfer



send pkt0

rcv pkt0
send ACK0

rcv ACK0
send pkt1

rcv pkt1
send ACK1

rcv ACK1
send pkt0

rcv pkt0
send ACK0

a. Operation with no loss

send pkt0

rcv pkt0
send ACK0

rcv ACK0
send pkt1

timeout
resend pkt1

rcv pkt1
send ACK1

rcv ACK1
send pkt0

rcv pkt0
send ACK0

b. Lost packet

Principles of reliable data transfer

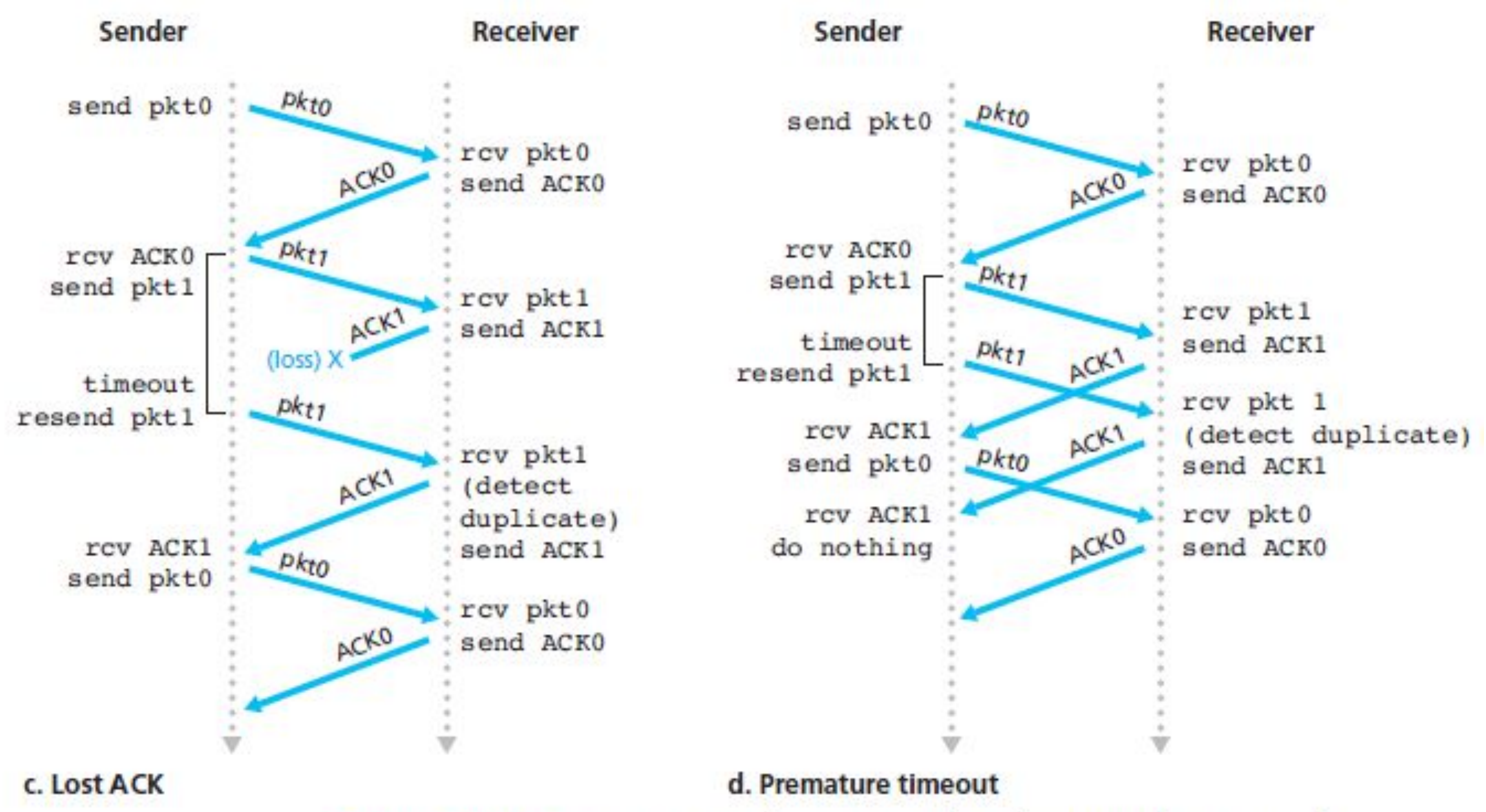


Figure 3.16 ♦ Operation of rdt3.0, the alternating-bit protocol

PES UNIVERSITY ONLINE

THANK YOU

Dr. Arpita Thakre

Department of Electronics and Communication Engineering