SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

Course Name – 19IT401 Computer Networks

II Year / IV Semester

Unit 4 – Transport Layer

Topic 4 – Error Control

Error Control

- TCP is a reliable transport-layer protocol.
- An application program that delivers a stream of data to TCP relies on TCP to deliver the entire stream to the application program on the other end in order, without error, and without any part lost or duplicated.
- TCP provides reliability using error control.
- Error control includes mechanisms for
  - detecting and resending corrupted segments,
  - resending lost segments,
  - storing out-of order segments until missing segments arrive, and
  - detecting and discarding duplicated segments
- Error control in TCP is achieved through the use of three simple tools: checksum, acknowledgment, and time-out.

Error Control

Checksum

- Each segment includes a checksum field, which is used to check for a corrupted segment.
- If a segment is corrupted, as detected by an invalid checksum, the segment is discarded by the destination TCP and is considered as lost.
- TCP uses a 16-bit checksum that is mandatory in every segment.

Acknowledgment

- TCP uses acknowledgments to confirm the receipt of data segments.
- Control segments that carry no data, but consume a sequence number, are also acknowledged.
- ACK segments are never acknowledged.

Acknowledgment Type

- In the past, TCP used only one type of acknowledgment: **cumulative acknowledgment**.
- Today, some TCP implementations also use **selective acknowledgment**.

Error Control

Cumulative Acknowledgment (ACK)

- TCP was originally designed to acknowledge receipt of segments cumulatively.
- The receiver advertises the next byte it expects to receive, ignoring all segments received and stored out of order.
- This is sometimes referred to as positive cumulative acknowledgment, or ACK.
- The 32-bit ACK field in the TCP header is used for cumulative acknowledgments.

Selective Acknowledgment (SACK)

- A SACK does not replace an ACK, but reports additional information to the sender.
- A SACK reports a block of bytes that is out of order, and also a block of bytes that is duplicated, i.e., received more than once.
- However, since there is no provision in the TCP header for adding this type of information, SACK is implemented as an option at the end of the TCP header.

Error Control

Generating Acknowledgments

1. When end A sends a data segment to end B, it must include (piggyback) an acknowledgment that gives the next sequence number it expects to receive. This rule decreases the number of segments needed and therefore reduces traffic.

1. When the receiver has no data to send and it receives an in-order segment (with expected sequence number) and the previous segment has already been acknowledged, the receiver delays sending an ACK segment until another segment arrives or until a period of time (normally 500 ms) has passed. In other words, the receiver needs to delay sending an ACK segment if there is only one outstanding in-order segment. This rule reduces ACK segments.

2. When a segment arrives with a sequence number that is expected by the receiver, and the previous in-order segment has not been acknowledged, the receiver immediately sends an ACK segment. In other words, there should not be more than two in-order unacknowledged segments at any time. This prevents the unnecessary retransmission of segments that may create congestion in the network.

Error Control

Generating Acknowledgments

4. When a segment arrives with an out-of-order sequence number that is higher than expected, the receiver immediately sends an ACK segment announcing the sequence number of the next expected segment. This leads to the fast retransmission of missing segments

5. When a missing segment arrives, the receiver sends an ACK segment to announce the next sequence number expected. This informs the receiver that segments reported missing have been received.

6. If a duplicate segment arrives, the receiver discards the segment, but immediately sends an acknowledgment indicating the next in-order segment expected. This solves some problems when an ACK segment itself is lost

Error Control

Retransmission

- The heart of the error control mechanism is the retransmission of segments.
- When a segment is sent, it is stored in a queue until it is acknowledged.
- When the retransmission timer expires or when the sender receives three duplicate ACKs for the first segment in the queue, that segment is retransmitted

Retransmission after RTO

- The sending TCP maintains one retransmission time-out (RTO) for each connection.
- When the timer matures, i.e. times out, TCP resends the segment in the front of the queue (the segment with the smallest sequence number) and restarts the timer.

Retransmission after Three Duplicate ACK Segments

- This feature is called fast retransmission.
- In this version, if three duplicate acknowledgments (i.e., an original ACK plus three exactly identical copies) arrive for a segment, the next segment is retransmitted without waiting for the time-out.

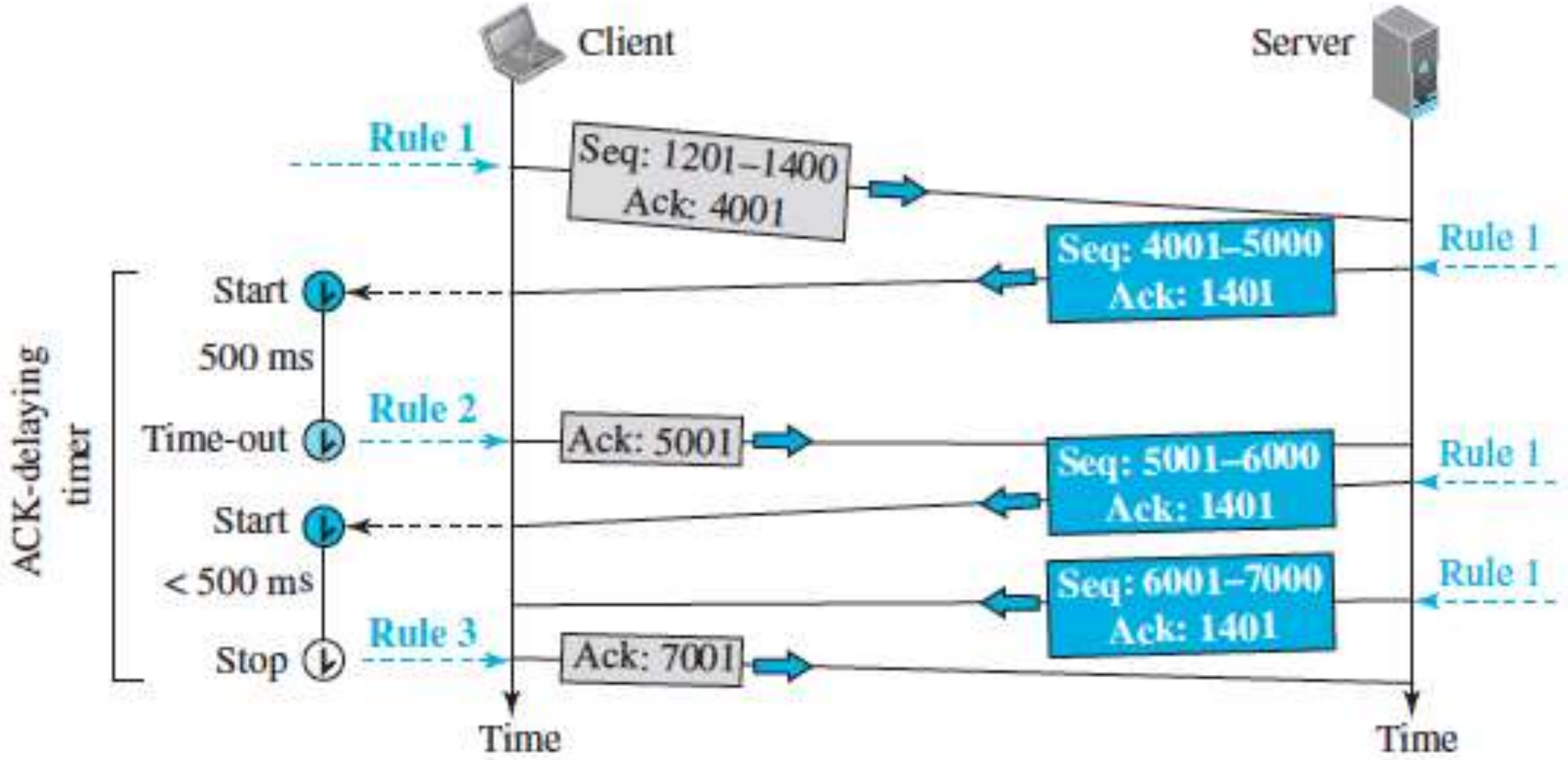
Error Control

Out-of-Order Segments

- TCP implementations today do not discard out-of-order segments.
- They store them temporarily and flag them as out-of-order segments until the missing segments arrive.
- Note, however, that out-of-order segments are never delivered to the process. TCP guarantees that data are delivered to the process in order.

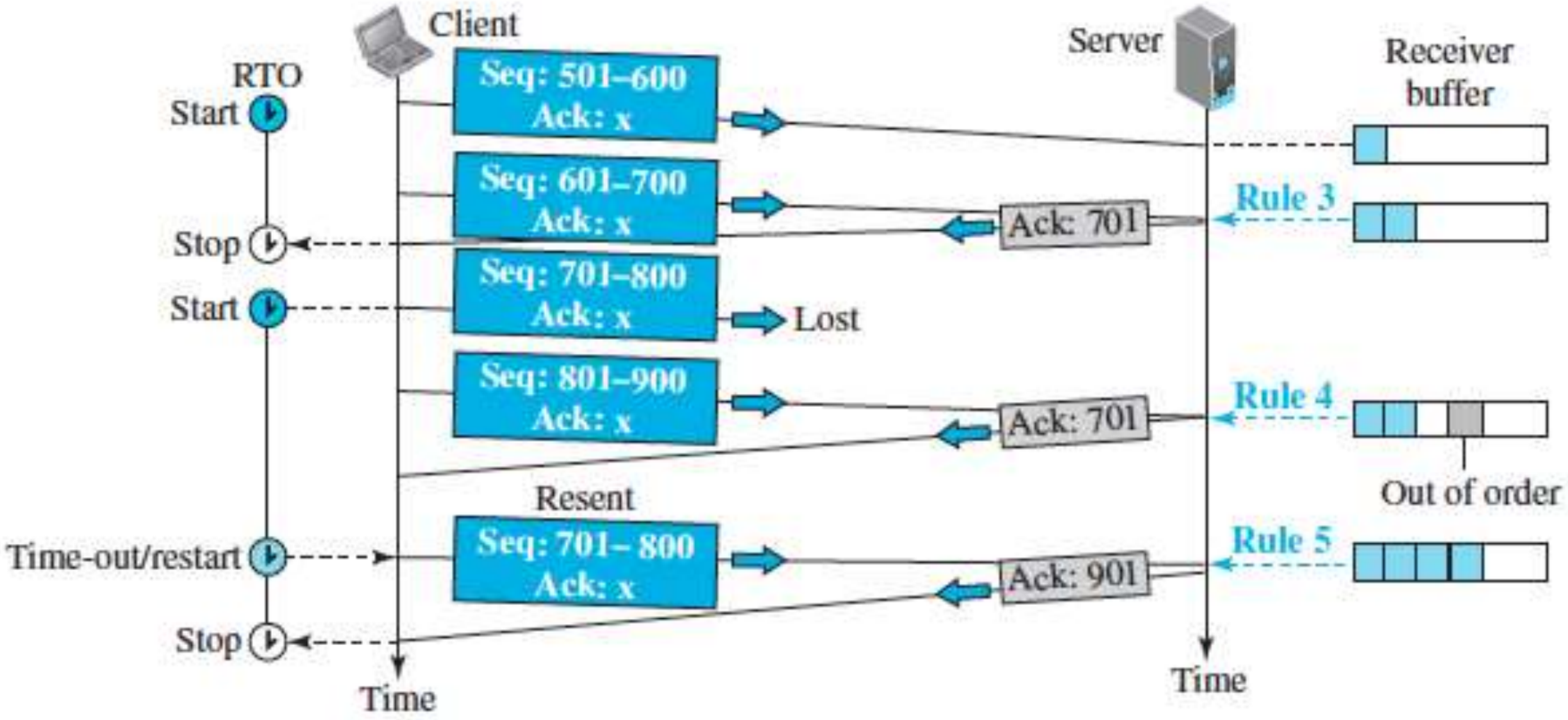
Error Control

Normal Operation



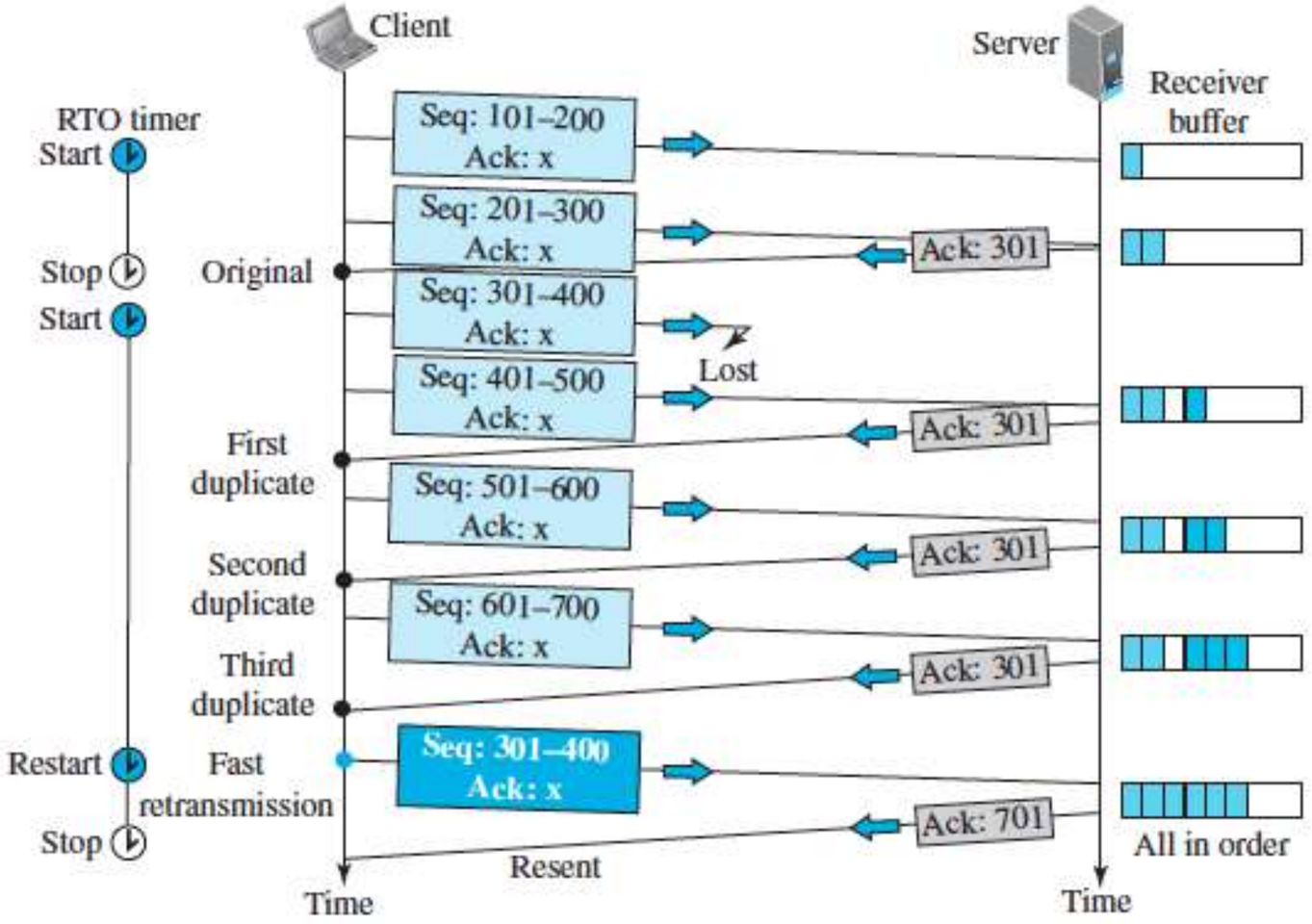
Error Control

Lost Segment



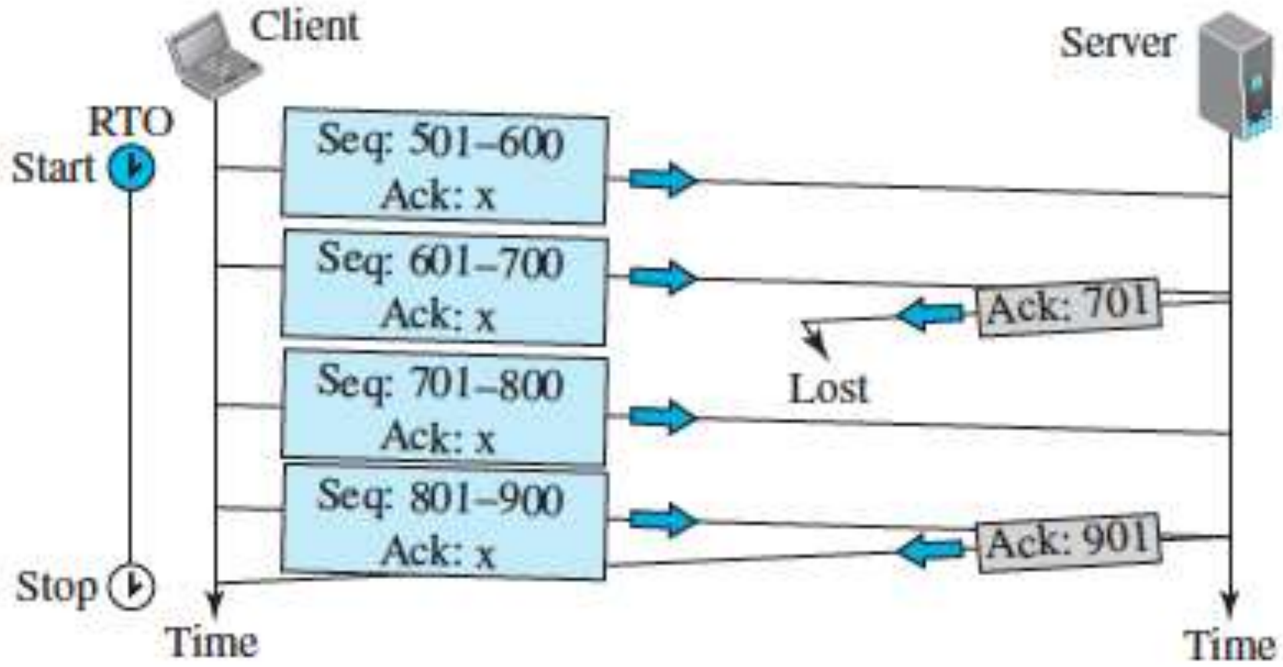
Error Control

Fast Retransmission



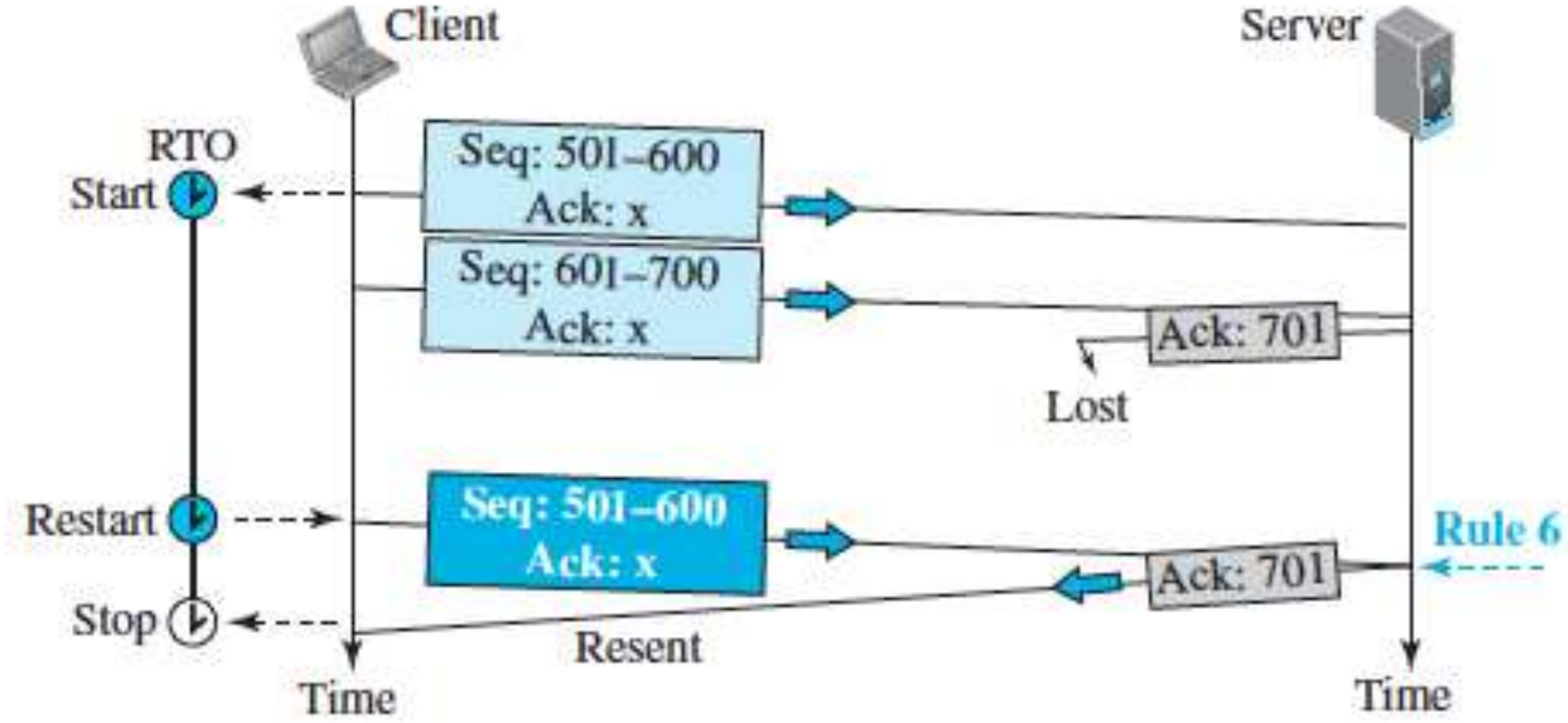
Error Control

Automatically Corrected Lost ACK



Error Control

Lost Acknowledgment Corrected by Resending a Segment



THANK YOU