UNIT -3
Mobile Transport Layer

VIII SEMESTER

Mobile Computing

ETIT-402

Mobile Transport Layer

- Traditional TCP/IP
- Transport Layer Protocols-
  - Indirect
  - Snooping
  - Mobile TCP

Book to be Referred

1) Mobile Computing by Rak kamal ,Chapter 6, Page No.-271-291

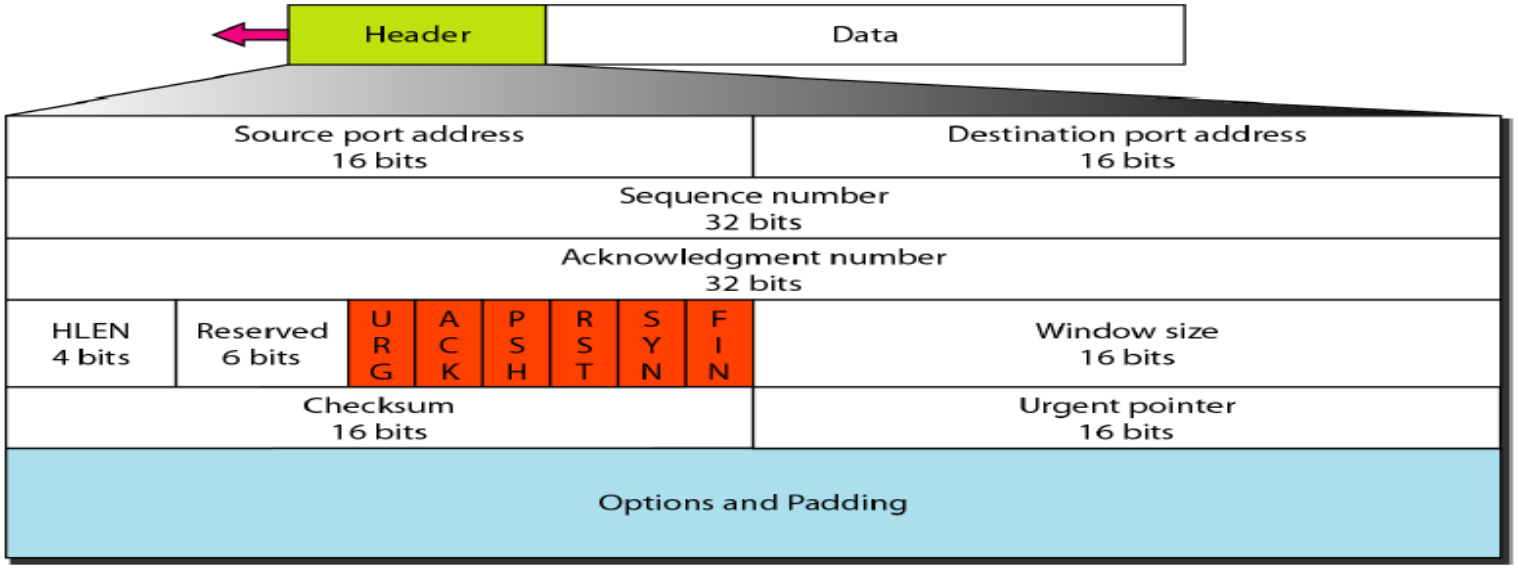
2) Mobile Communication By Jochen Schiller,Chapter 9, Page No. 351-365

Traditional TCP/IP

- Two Transport layer protocol :-
  - UDP (User Data Gram)
  - TCP (Transmission Control Protocol)
- UDP (User Data Gram Protocol)
  - Connection less Protocol
  - Doesn't require session establishment, data flow, congestion control, session termination
  - UDP header used for encapsulation at L_4 during transmission of port data.
  - UDP Header - 4 octet or Two word
    - First Word :-Consist 16 bit source and 16 bit destination address
    - Second word:- 16 bit length of datagram and 16 bit header check sum
  - Pseudo header is used as prefix.
  - Use full in transmitting datagram for multicasting, registration request etc.

Traditional TCP/IP Contd.

- TCP:-
  - Connection oriented protocol
  - Feature:-
    - Transmission as data stream
    - Buffering and retransmission
    - Session start , data transmission and session termination fully acknowledged from end to end
    - In order delivery
    - Congestion control and avoidance
  - TCP Header



Traditional TCP/IP Contd.

- TCP Data Stream :-
  - Data Steam consist byte delivered using virtual connection between sockets
  - Each socket having port ID and IP address
  - No of byte in stream depend Transport PDU
- TCP Data Delivery
  - TCP specify number of acknowledgement sequence from one end to other
  - Checksum field for detecting the error. Takes in account the header as well data field
- TCP Data Flow Control
  - Window Size adjustment
  - Cumulative acknowledgement
  - Reverse Packet acknowledgement
  - Duplicate acknowledgement
  - Delayed Acknowledgment

Traditional TCP/IP Contd.

- Congestion Control
  - Method Employed for Congestion control
    - Slow Start and Congestion avoidance
    - Fast recovery after packet loss
    - Fast retransmit and fast recovery
    - Selective acknowledgement
    - Explicit congestion notification
- Slow Start Method
  - sender calculates a congestion window for a receiver
  - start with a congestion window size equal to one segment (packet)
  - Exponentially increase congestion window till congestion threshold, then linear increase
  - Timeout/missing acknowledgement causes reduction of congestion threshold to half of the current congestion window
  - congestion window starts again with one segment

Traditional TCP/IP Contd.

- TCP fast retransmit/fast recovery
  - TCP sends an ACK only after receiving a packet
  - If sender receives duplicate ACKs, this is due to gap in received packets at the receiver
  - Receiver got all packets up to the gap and is actually receiving packets
  - Conclusion: packet loss not due to congestion, retransmit, continue with current congestion window (do not use slow-start)

Methods For Wireless/Mobile Networks

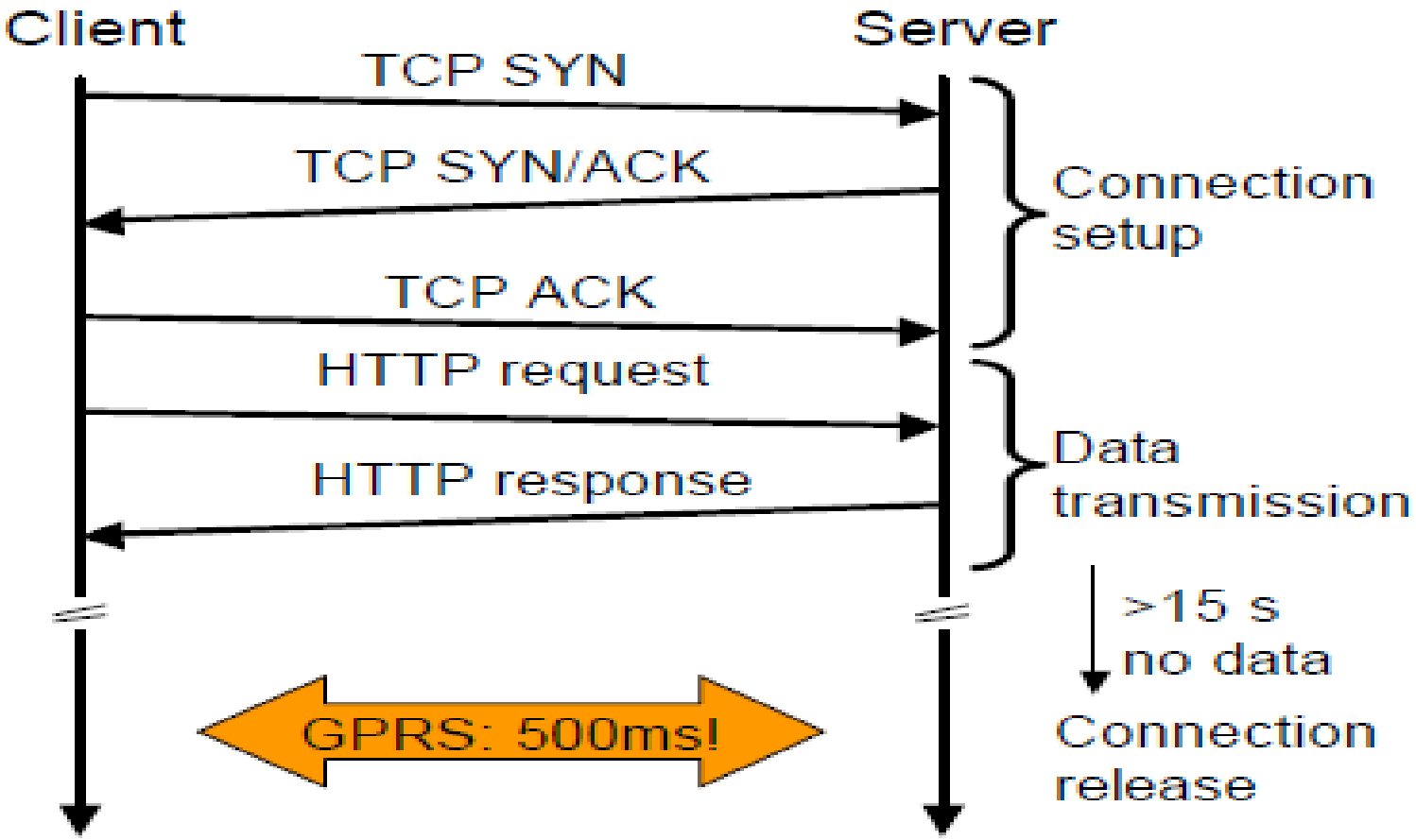
HTTP (used by web services) typically uses TCP

- Reliable transport between client and server required

- TCP
  - Steam oriented, not transaction oriented
  - Network friendly: time-out
    - Congestion
    - slow down transmission

- Well known – TCP wrongly assumes congestion in wireless and mobile networks when
  - Packet losses due to transmission errors
  - Packet loss due to change of network Result
  - Severe ***performance degradation***

Methods For Wireless/Mobile Networks

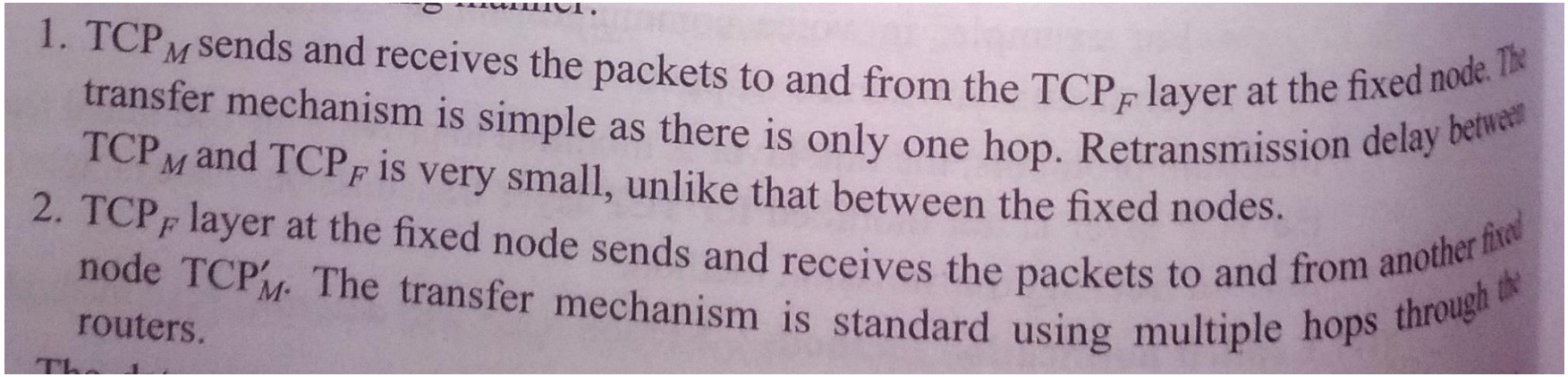


Methods For Wireless/Mobile Networks

- Problem Faced By Mobile network
  - Convention TCP presumes that packet loss is due to congestion only
- Problem associated with Mobile network for packet loss
  - Data linking transmission quality problem
  - High BER, leads to high retransmission rate
  - Duplicate acknowledge leads to reduced window size
- Methods Used By Transport layer
  - Split TCP
    - Split TCP in two layers
    - Upper layer for the requirement of Mobile network and send data stream to conventional TCP
    - Four Method :- Indirect, Selective repeat, mobile end TCP, and Mobile TCP
  - TCP aware link Layer Methods
  - Data Link Layer modification Methods
  - Explicit Notification Methods

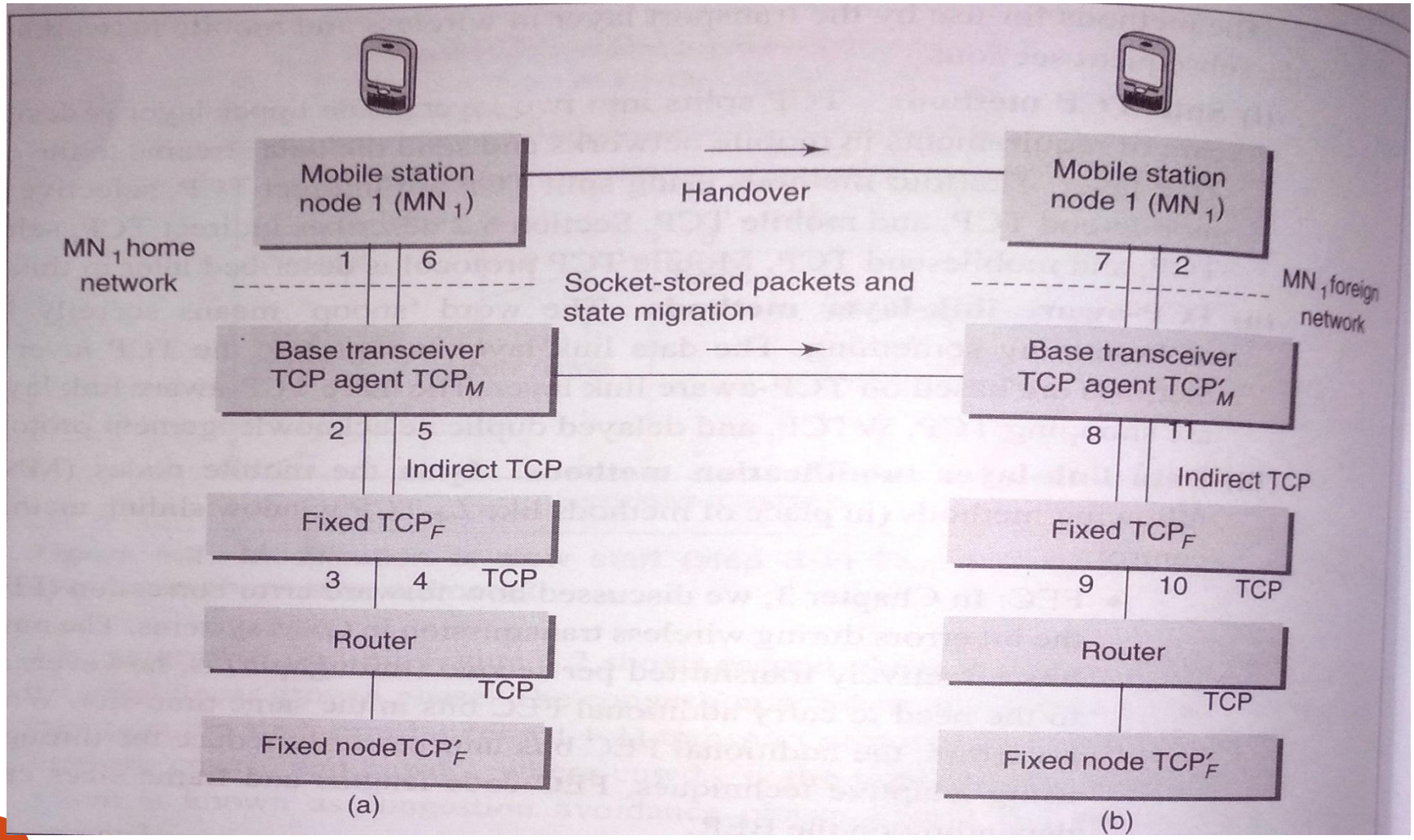
Indirect TCP

- Indirect TCP or I-TCP segments the connection
  - No changes to the TCP protocol for hosts connected to the wired Internet, millions of computers use (variants of) this protocol
  - Optimized TCP protocol for mobile hosts
  - Splitting of the TCP connection at, e.g., the foreign agent into 2 TCP connections, no real end-to-end connection any longer
  - hosts in the fixed part of the net do not notice the characteristics of the wireless part
  - Indirect TCP Function as mentioned below



1. TCP_M sends and receives the packets to and from the TCP_F layer at the fixed node. The transfer mechanism is simple as there is only one hop. Retransmission delay between TCP_M and TCP_F is very small, unlike that between the fixed nodes.
2. TCP_F layer at the fixed node sends and receives the packets to and from another fixed node TCP'_M. The transfer mechanism is standard using multiple hops through the routers.

Indirect TCP



Indirect TCP

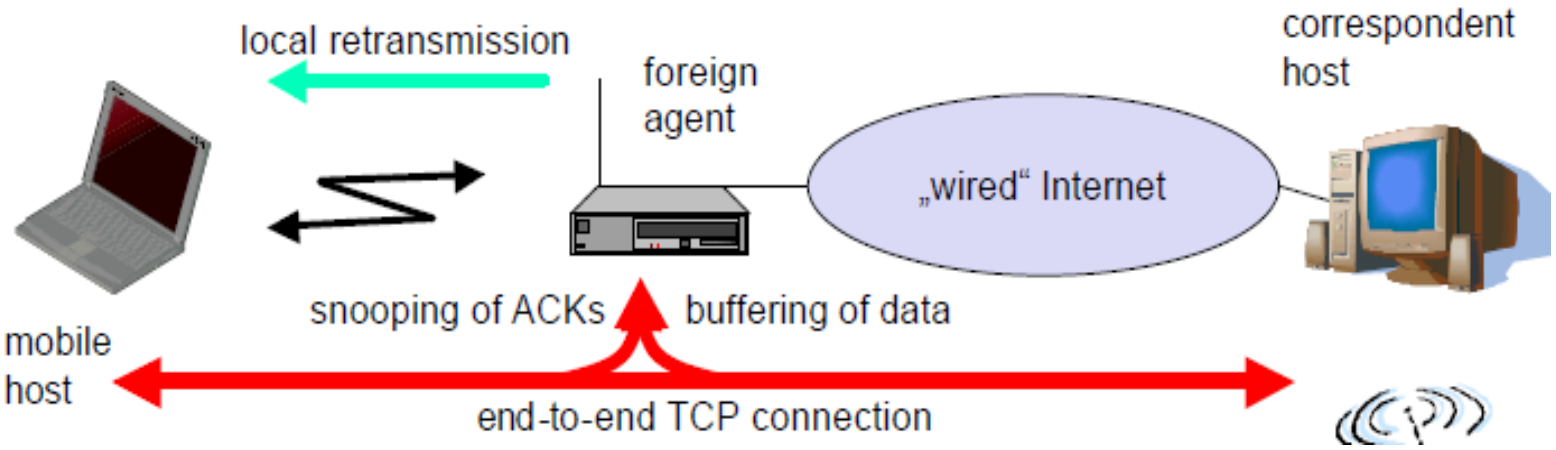
- Advantages
  - No changes in the fixed network necessary, no changes for the hosts (TCP protocol) necessary, all current optimizations to TCP still work
  - Wireless link transmission errors isolated from those in fixed network
  - Simple to control, mobile TCP is used only for one hop between, e.g., a foreign agent and mobile host
  - Very fast retransmission of packets is possible, the short delay on the mobile hop is known
- Disadvantages
  - loss of end-to-end semantics, an acknowledgement to a sender does now not any longer mean that a receiver really got a packet, foreign agents might crash
  - higher latency possible due to buffering of data within the foreign agent and forwarding to a new foreign agent

Indirect TCP

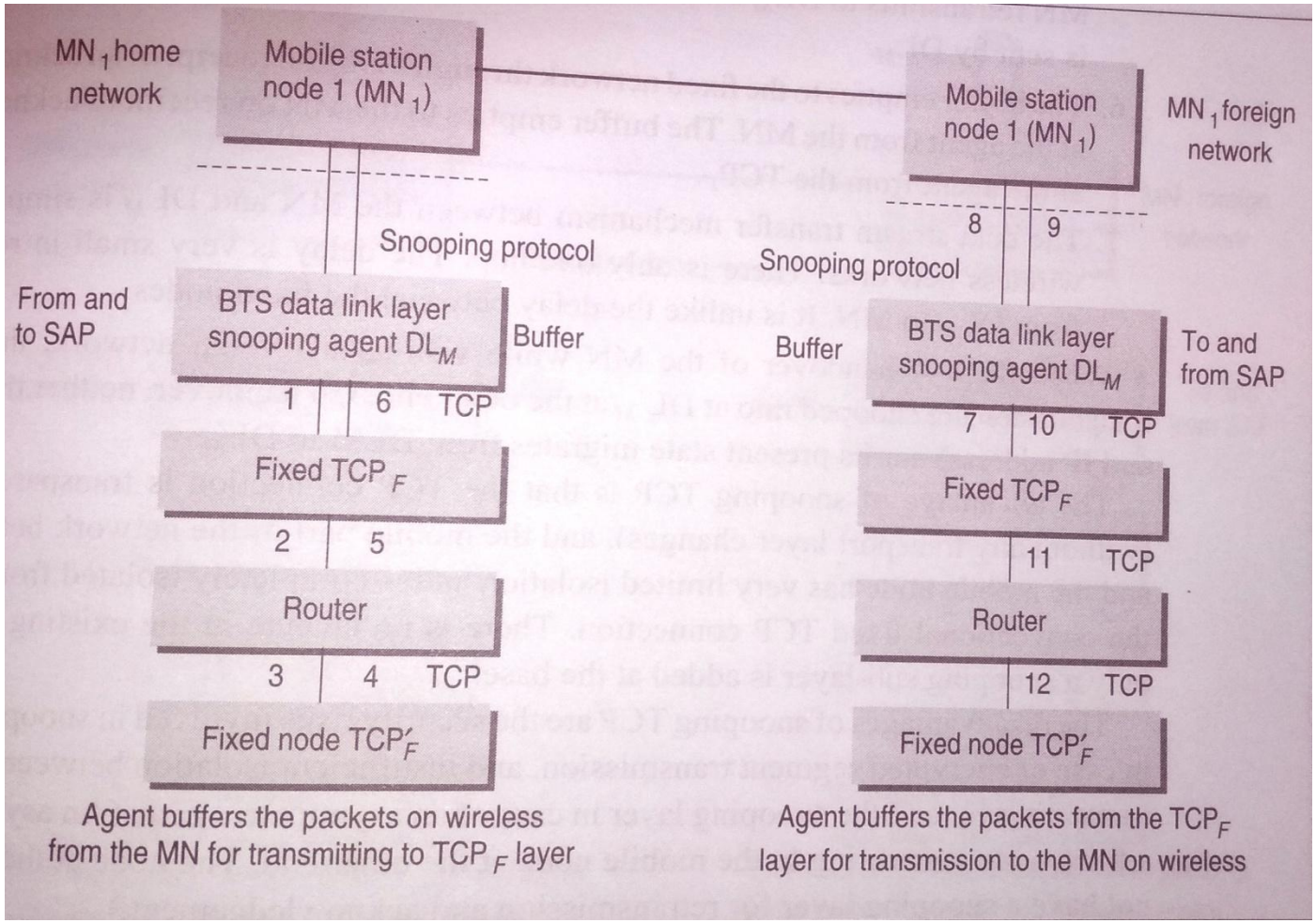
- Selective repeat protocol :-
  - Modification of Indirect TCP
  - Uses UDP between BTS and MN
  - Doesn't guarantee in order delivery between MN and BTS , unlike TCP
- Mobile End Transmission Protocol:-
  - Another Modification of Indirect TCP
  - Guarantee in order delivery between MN and BTS , like TCP
  - Uses Mobile end transmission protocol between MN and BTS
  - Rest Similar to Indirect TCP

Snooping TCP

- Transparent“ extension of TCP within the foreign agent
  - Buffering of packets sent to the mobile host
  - Lost packets on the wireless link (both directions!) will be retransmitted immediately by the mobile host or foreign agent, respectively (so called “local” retransmission)
  - The foreign agent therefore “snoops” the packet flow and recognizes acknowledgements in both directions, it also filters ACKs
  - Changes of TCP only within the foreign agent



Snooping TCP



Snooping TCP

- Data transfer to the mobile host
  - FA buffers data until it receives ACK of the MH, FA detects packet loss via duplicated ACKs or time-out
  - fast retransmission possible, transparent for the fixed network
- Data transfer from the mobile host
  - FA detects packet loss on the wireless link via sequence numbers, FA answers directly with a NACK to the MH
  - MH can now retransmit data with only a very short delay
- Integration with MAC layer
  - MAC layer often has similar mechanisms to those of TCP
  - thus, the MAC layer can already detect duplicated packets due to retransmissions and discard them

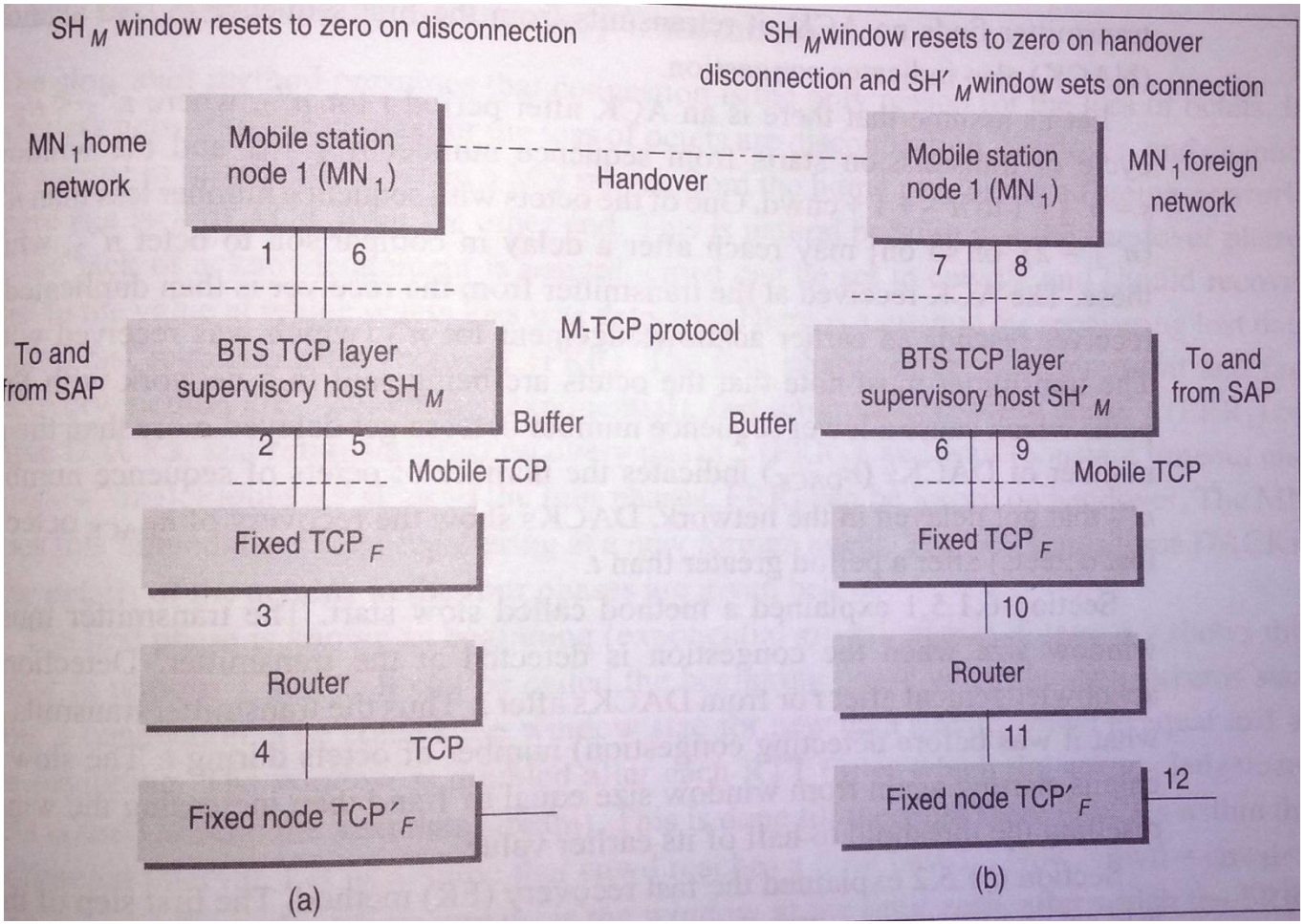
Snooping TCP

- Advantage
  - Transparent End to End Connection
  - No Change in existing TCP network, only snooping sub layer is added
- Disadvantage
  - snooping TCP does not isolate the wireless link as good as I-TCP
  - snooping might be tough if packets are encrypted
- Wireless TCP
  - Modified Form of snooping TCP
  - Modifies Time stamp on packet while returning ACK to compensate increased RTT
  - Cannot be used with shared LAN

Mobile TCP

- Special handling of lengthy and/or frequent disconnections
- M-TCP splits as I-TCP does
  - unmodified TCP fixed network to supervisory host (SH)
  - optimized TCP SH to MH
- Supervisory host
  - no caching, no retransmission
  - monitors all packets, if disconnection detected
    - set sender window size to 0
    - sender automatically goes into persistent mode
  - old or new SH reopen the window
- Advantages
  - maintains semantics, supports disconnection, no buffer forwarding
- Disadvantages
  - loss on wireless link propagated into fixed network
  - adapted TCP on wireless link

Mobile TCP


SH M window resets to zero on disconnection
SH M window resets to zero on handover
disconnection and SH′ M window sets on connection
MN 1 home network
Mobile station node 1 (MN 1)
Handover
Mobile station node 1 (MN 1)
MN 1 foreign network
1
6
7
8
M-TCP protocol
To and from SAP
BTS TCP layer supervisory host SH M
Buffer
Buffer
BTS TCP layer supervisory host SH′ M
To and from SAP
2
5
6
9
Mobile TCP
Mobile TCP
Fixed TCP F
Fixed TCP F
3
10
Router
Router
4
TCP
11
Fixed node TCP′ F
12
Fixed node TCP′ F
(a)
(b)

Comparison Between Different Approach

Approach	Mechanism	Advantages	Disadvantages
Indirect TCP	splits TCP connection into two connections	isolation of wireless link, simple	loss of TCP semantics, higher latency at handover
Snooping TCP	"snoops" data and acknowledgements, local retransmission	transparent for end-to-end connection, MAC integration possible	problematic with encryption, bad isolation of wireless link
M-TCP	splits TCP connection, chokes sender via window size	Maintains end-to-end semantics, handles long term and frequent disconnections	Bad isolation of wireless link, processing overhead due to bandwidth management
Fast retransmit/ fast recovery	avoids slow-start after roaming	simple and efficient	mixed layers, not transparent
Transmission/ time-out freezing	freezes TCP state at disconnect, resumes after reconnection	independent of content or encryption, works for longer interrupts	changes in TCP required, MAC dependant
Selective retransmission	retransmit only lost data	very efficient	slightly more complex receiver software, more buffer needed
Transaction oriented TCP	combine connection setup/release and data transmission	Efficient for certain applications	changes in TCP required, not transparent

Thank you