UNIT II

Fundamentals of (wireless) MAC protocols

- The fundamental task of any MAC protocol is to regulate the access of a number of nodes to a shared medium in such a way that certain application-dependent performance requirements are satisfied.
- The MAC protocol determines for a node the points in time when it accesses the medium to try to transmit a data, control, or management packet to another node (unicast) or to a set of nodes (multicast, broadcast).
- Collisions can happen if the MAC protocol allows two or more nodes to send packets at the same time.
- Collisions can result in the inability of the receiver to decode a packet correctly, causing the upper layers to perform a retransmission.

MAC (Medium Access Control)protocol issues

- There are many issues that need to be addressed in order to design an efficient MAC protocol in a wireless adhoc network environment.
- The main question in connection with MAC in the wireless it is possible to use complicated MAC schemes from wired networks.
- For example: CSMA/CD
- Let us consider carrier sense multiple access with collision detection(CDMA/CD) which works as follow:
- A sender sense the medium (a wire) to see if it is free . If the medium is busy , the sender waits until it is free.
- If the medium is free sender starts transmitting data continues to listen into the medium.
- If sender now detects a collision while sending it stops at once and sends jamming signal .
- Why does this scheme fail in wireless networks?

- CDMA/CD is not really interested in collision at sender, but rather in those at the receiver.
- The signal should reach the receiver without collisions.
- But the sender is one detecting the collisions.
- This is not problem using a wire , and if collision occurs somewhere in the wire , everybody will notice it.
- The situation is different in wireless networks.
- The strength of signal decreases proportionally to the square of the distance to a sender.
- The sender may now apply carrier senses and detect an idle medium.
- Thus the sender start sending but a collision happens at the receiver due to a second sender.
- This is hidden terminal problem.
- The sender detects no collision assumes that the data has been transmitted without errors, but actually a collision which destroyed the data at the receiver.

Hidden terminal problem

Hidden terminals

- Consider the situation as show in figure.
- A sends to B, C cannot receive A
- C wants to send to B, C senses a “free” medium (CS fails)
- collision at B, A cannot receive the collision (CD fails)
- A is “hidden” for C


A
B
C
A
B
C

- Exposed terminals
  - Consider the situation as show in figure.
  - B sends to A, C wants to send to another terminal (not A or B)
  - C has to wait, CS signals a medium in use
  - but A is outside the radio range of C, therefore waiting is not necessary
  - C is “exposed” to B

Important classes of MAC protocols

- MAC Protocols can be roughly classified into the following classes

1)fixed assignment protocols,

2)Demand assignment protocols

3) random access protocols.

fixed assignment protocols

- In this class of protocols, the available resources are divided between the nodes such that the resource assignment is long term, and each node can use its resources exclusively without the risk of collisions.
- Typical protocols of this class are TDMA, FDMA, CDMA, and SDMA.

The Time Division Multiple Access (TDMA)

- The Time Division Multiple Access (TDMA) scheme subdivides the time axis into fixed-length super frames and each super frame is again subdivided into a fixed number of time slots.
- These time slots are assigned to nodes exclusively and hence the node can transmit in this time slot periodically in every super frame.
- TDMA requires tight time synchronization between nodes to avoid overlapping of signals in adjacent time slots.

Frequency Division Multiple Access (FDMA)

- Frequency Division Multiple Access (FDMA), the available frequency band is subdivided into a number of sub channels and these are assigned to nodes, which can transmit exclusively on their channel.
- This scheme requires frequency synchronization.

Code Division Multiple Access (CDMA)

- In Code Division Multiple Access (CDMA) schemes , the nodes spread their signals over a much larger bandwidth than needed, using different codes to separate their transmissions.
- The receiver has to know the code used by the transmitter, all parallel transmissions using other codes appear as noise. Crucial to CDMA is the code management.

Space Division Multiple Access

- Space Division Multiple Access (SDMA), the spatial separation of nodes is used to separate their transmissions.
- SDMA requires arrays of antennas and sophisticated signal processing techniques.

Demand Assignment Protocols

- The main objective of demand assignment protocols is to improve channel utilization by allocating the capacity of the channel to contending nodes in an optimum or near-optimum fashion.
- Unlike fixed-assignment schemes, where channel capacity is assigned exclusively to the network nodes in a predetermined fashion regardless of their current communication needs, demand assignment protocols ignore idle nodes and consider only nodes that are ready to transmit.
- The channel is allocated to the node selected for a specified amount of time, which may vary from a fixed-time slot to the time it takes to transmit a data packet.

- Demand assignment protocols typically require a network control mechanism to arbitrate access to the channel between contending nodes.
- Furthermore, a logical control channel, other than the data channel, may be required for contending stations to dynamically request access to the communication medium.
- Demand assignment protocols may be further classified as centralized or distributed. Polling schemes are representative of centralized control, whereas token- and reservation-based schemes use distributed control.

polling

- A widely used demand assignment scheme is polling.
- In this scheme, a master control device queries, in some predetermined order, each slave node about whether it has data to transmit.
- If the polled node has data to transmit, it informs the controller of its intention to transmit.
- In response, the controller allocates the channel to the ready node, which uses the full data rate to transmit its traffic.
- If the node being polled has no data to transmit, it declines the controller’s request.
- In response, the controller proceeds to query the next network node.

- The main advantage of polling is that all nodes can receive equal access to the channel.
- The major drawback of polling is the substantial overhead caused by the large number of messages generated by the controller to query the communicating nodes.
- Efficiency of the polling scheme depends on the reliability of the controller.

Reservation

- The basic idea in a reservation-based scheme is to set some time slots for carrying reservation messages.
- Since these messages are usually smaller than data packets, they are called mini slots.
- When a station has data to send, it requests a data slot by sending a reservation message to the master in a reservation mini slot.
- In some schemes, such as in fixed-priority-oriented demand assignment, each station is assigned its own mini slot.
- In a reservation-based scheme, if each station has its own reservation mini slot, collision can be avoided.

- Moreover, if reservation requests have a priority field, the master can schedule urgent data before delay-insensitive data.
- Packet collisions can happen only when stations contend for the mini slot, which use only a small fraction of the total bandwidth.
- Thus, the largest part of the bandwidth assigned to data packets is used efficiently

Random Assignment Protocols

- In fixed-assignment schemes, each communicating node is assigned a frequency band in FDMA systems or a time slot in TDMA systems.
- This assignment is static, however, regardless of whether or not the node has data to transmit.
- In the absence of data to be transmitted, the node remains idle, thereby resulting in the allocated bandwidth to be wasted.
- Random assignment strategies attempt to address this shortcoming by eliminating pre allocation of bandwidth to communicating nodes.

- these strategies do not assign any predictable or scheduled time for any node to transmit.
- All backlogged nodes must contend to access the transmission medium.
- Collision occurs when more than one node attempts to transmit simultaneously.
- To deal with collisions, the protocol must include a mechanism to detect collisions and a scheme to schedule colliding packets for subsequent retransmissions.
- Random access protocols were first developed for long radio links and for satellite communications. The ALOHA protocol, also referred to as pure ALOHA, was one of the first such media access protocols.

Busy tone

- Several approaches have been proposed to eliminate, or at least reduce, the impact of the hidden- and exposed-node problems on the network throughput.
- The first approach is based on the use of a busy tone.
- The basic idea of the busy-tone approach stems from the observation that collisions occur at the receiving node whereas CSMA is performed at the transmission node.

- To address the disparity between the design goals of CSMA as originally specified and application of the protocol to wireless environments, the busy-tone approach requires the use of two separate channels: a data channel and a control channel.
- The data channel is used to transmit data exclusively.
- The control channel is used by the receiver to signal to the remaining nodes in the network that it is in the process of receiving data.
- Immediately after the node starts to receive a data packet, which carries its address in the destination address field, the node initiates the emission of an unmodulated wave on the control channel, indicating that its receiver is busy.

- The node continues to transmit the busy tone at the same time that it is receiving the data packet until the packet is fully received.
- Before transmitting a data packet, the sending node must first sense the control channel for the presence of a busy tone.
- The node proceeds to transmit the data packet only if the control channel is free.
- The busy-tone approach solves both the hidden- and exposed-node problems,
- assuming that the busy-tone signal is emitted at a level such that it is not too weak not to be heard by a node within the range of a receiver and not too strong to force more nodes than necessary to suppress their transmissions.
- The major drawback of the approach, however, is a node’s need to operate in duplex mode, to be able to transmit and receive simultaneously.

Ready-to-send (RTS), clear-to-send (CTS) handshake

- The second approach to deal with the hidden-node problem is based on collision avoidance.
- When a node intends to transmit a data packet, it first senses the carrier to determine if another node is already transmitting.
- If no other transmissions are sensed, the node sends a short RTS packet to the intended recipient of the data packet.
- If the recipient is, in fact, idle and senses that the medium is clear, it sends a short CTS packet in reply.
- Upon receiving the CTS packet, the transmitting node sends the actual data packet to its intended recipient.

- If after a predetermined period of time, the transmitting station does not receive a CTS packet in reply to its RTS packet, it waits a random period of time before repeating the RTS/CTS handshake procedure.
- The use of the RTS/CTS handshake procedure in CSMA/CA schemes to avoid collisions is depicted in Figure 5.4.

- In this scenario, node B intends to transmit a data packet to node C. It senses the carrier to determine if any other node is already transmitting. After it determines that the channel is free, it transmits a RTS packet.
- In addition to the destination address, the packet also contains the duration field, which indicates the time necessary to complete the transmission of the packet and the receipt of the corresponding acknowledgment.
- In response, the intended recipient of the packet, node C in this case, transmits a CTS packet, which contains the remaining time until the completion of the transmission.
- Upon receiving the RTS packet, station A sets an internal timer to the remaining time until completion of the data packet transmission and avoids transmitting any packet until the timer expires. When node B receives the CTS packet, it proceeds to transmit its data packet.
- Upon receiving the CTS packet, node D sets an internal timer to the remaining time until completion of the data and defers the transmission of any packets until the timer expires.

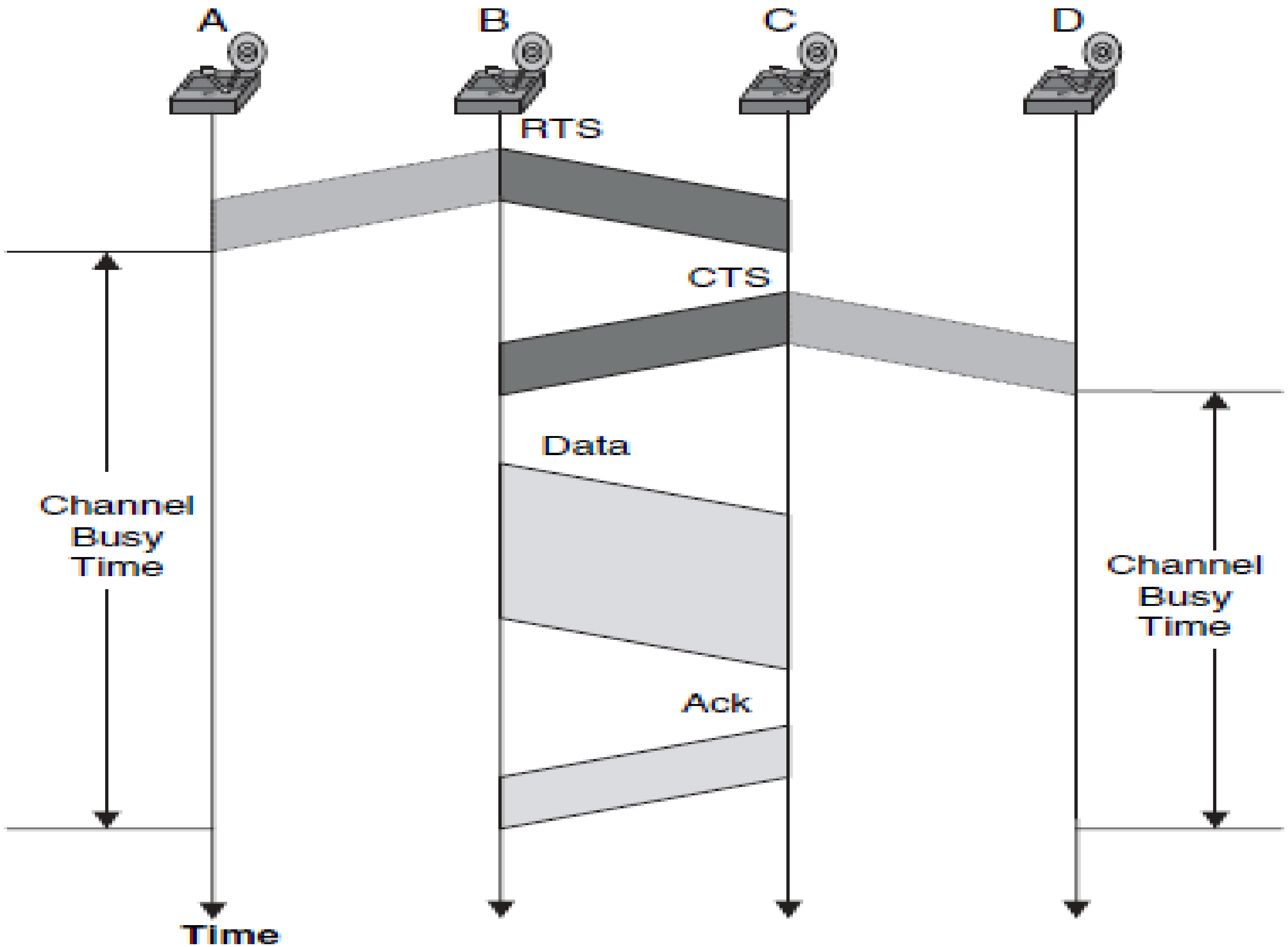
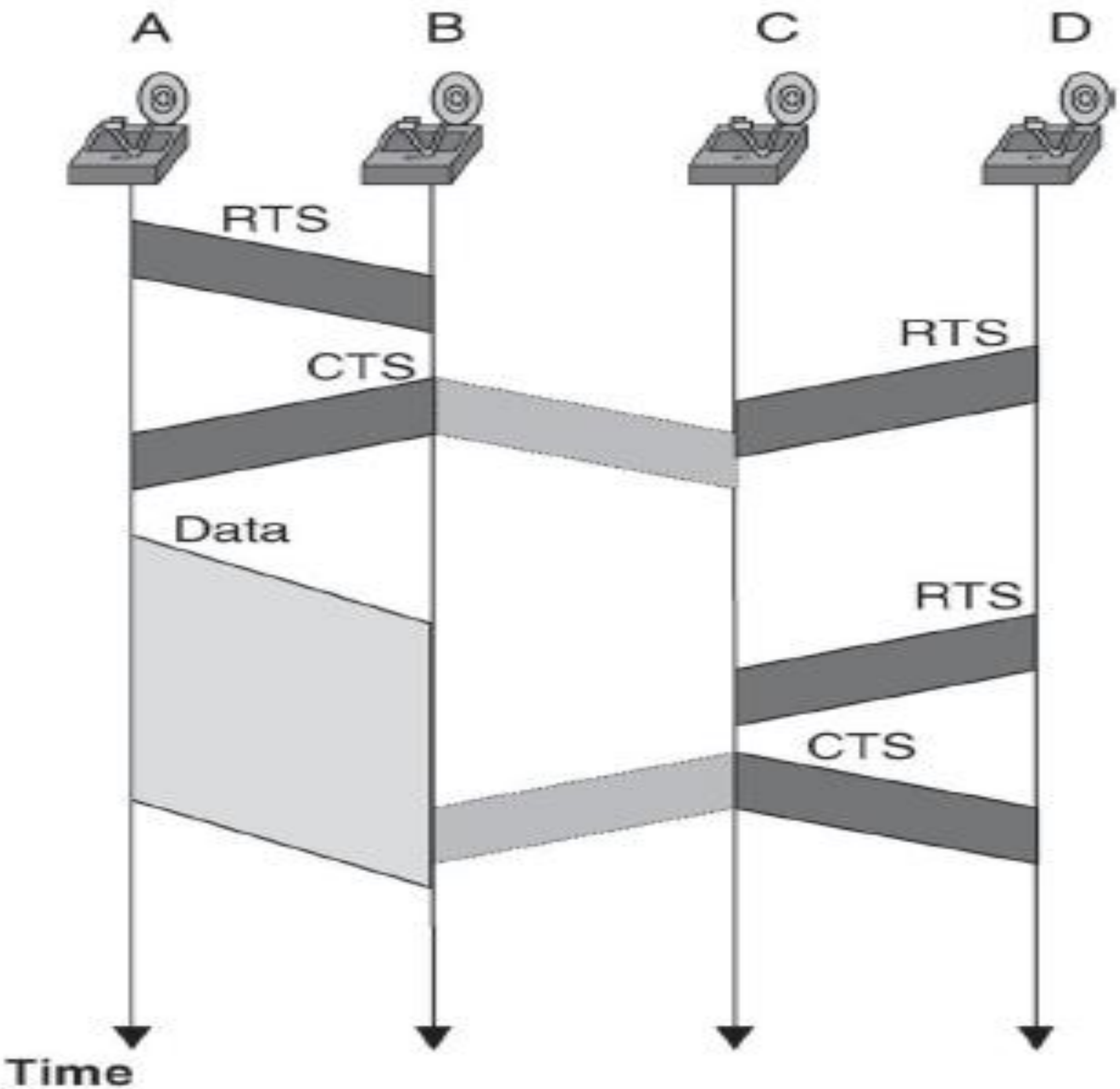


Figure 5.4 Collision avoidance using RTS/CTS handshake.

- Figure 5.5a, node A senses the channel to be free and sends an RTS packet to node B. In reply, node B sends a CTS packet.
- Node C, which is in the transmission range of node B, starts receiving the CTS packet.
- Before the reception of this packet is complete, however, node D, which is in the transmission range of node C, sends a RTS packet.
- The latter packet collides with the CTS packet sent by node B.
- Meanwhile, node A, which receives the CTS packet correctly, proceeds to transmit its data packet to node B.

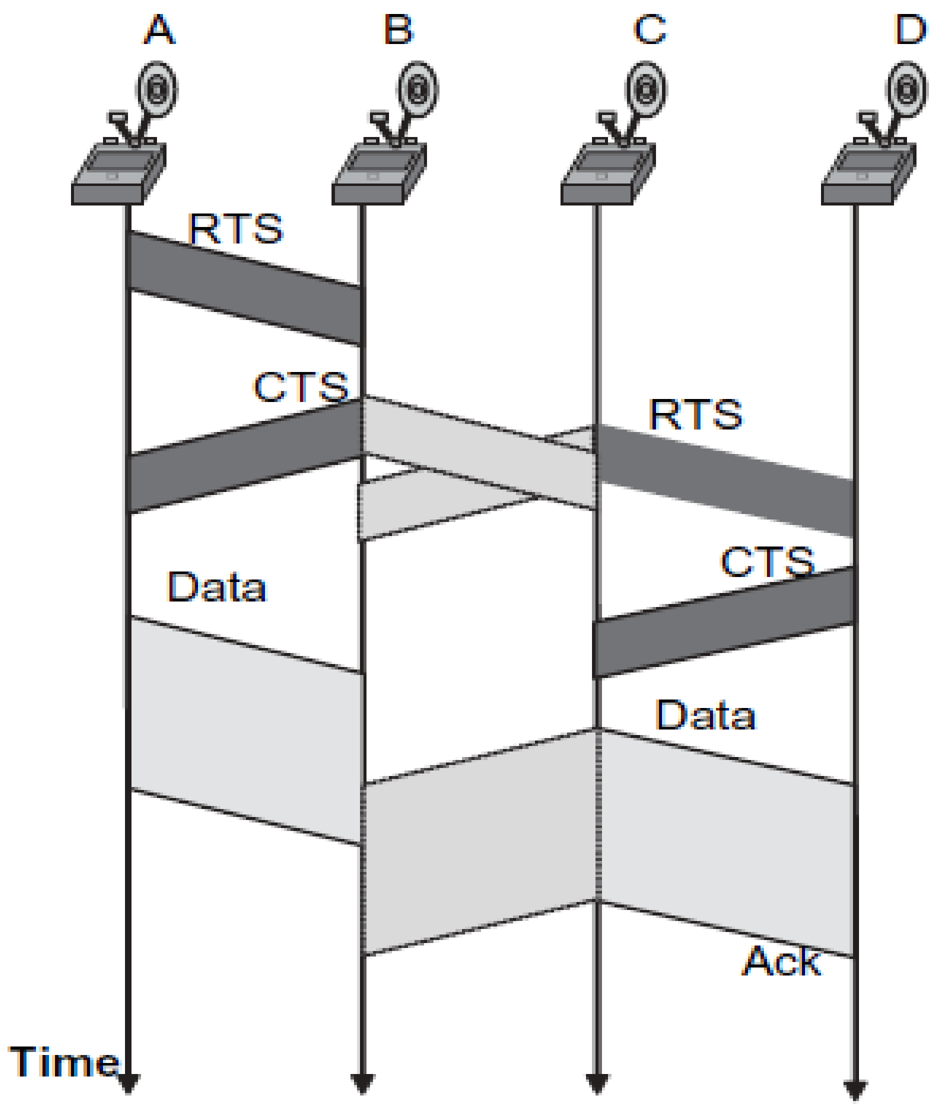
- Node D later times-out and retransmits its RTS packet.
- Since node C never received node B’s CTS packet, it assumes that the channel is free and replies with a CTS packet to node D.
- Since node B is within the transmission range of node C, the latter packet collides with the data packet being transmitted by node A.
- This procedure, however, does not completely solve the hidden-node problem.



(*a*)

- Figure 5.5b shows another case where collision avoidance fails, using the RTS/CTS handshake.
- In this scenario, node A senses the channel to be free and sends an RTS packet to node B.
- In reply, node B sends a CTS packet to node A.
- The CTS packet is received correctly by node A, which allows it to transmit its packet.
- The CTS packet is also received by node C, which is within the transmission range of node B.
- Since node C has started transmitting an RTS packet to node D, nearly at the same time that node B is transmitting its CTS packet; node C does not receive correctly the CTS packet sent by B.

- Node D, however, receives correctly the RTS packet sent by node C.
- In response, it sends a CTS packet to node C, thereby allowing it to start transmitting its data packet.
- Since node A did not complete transmission of its data packet to node B, node C’s data transmission causes a collision at node B.
- Despite its failure to solve the hiddennode problem completely, the RTS/CTS handshake is used widely in wireless networks to avoid packet collisions and increase network throughput.



(b)

Figure 5.5 Collision avoidance failure using RTS/CTS handshake.

MAC PROTOCOLS FOR WSNs

- The need to conserve energy is the most critical issue in the design of scalable and stable MAC layer protocols for WSNs.
- The main objective of most MAC-layer protocols is to reduce energy waste caused by collisions, idle listening, overhearing, and excessive overhead.
- These protocols can be categorized into two main groups: schedule- and contention-based MAC-layer protocols.

- Schedule-based protocols: It is class of deterministic MAC layer protocols in which access to the channel is based on a schedule. Channel access is limited to one sensor node at a time. This is achieved based on pre allocation of resources to individual sensor nodes.
- Contention-based MAC-layer protocols : avoid pre allocation of resources to individual sensors. Instead, a single radio channel is shared by all nodes and allocated on demand. Simultaneous attempts to access the communications medium, however, results in collision. The main objective of contention-based MAC layer protocols is to minimize, rather than completely avoid, the occurrence of collisions.

SENSOR-MAC CASE STUDY

- The sensor-MAC (S-MAC) protocol is designed explicitly to reduce energy waste caused by collision, idle listening, control overhead, and overhearing.
- The goal is to increase energy efficiency while achieving a high level of stability and scalability.
- S-MAC uses multiple techniques to reduce energy consumption, control overhead, and latency, in order to improve application-level performance.

Protocol Overview

- The protocol design assumes a large number of sensor nodes, with limited storage, communication, and processing capabilities.
- The nodes are configured in an ad hoc, self-organized, and self-managed wireless network.
- Data generated by sensors are processed and communicated in a store-and-forward manner.
- The applications supported by the network are assumed to alternate between long idle periods, during which no events occur, and bursty active periods, during which data flow toward the base station through message exchange among peer sensor nodes. data flows from source to sink

- Typical applications that fall into this category include surveillance and monitoring of natural habitats and protection of critical infrastructure.
- In these applications the sensors must be vigilant over long periods of time, during which they remain inactive until some event occurs.
- S-MAC exploits the bursty profile of sensor applications to establish low-duty-cycle operation on nodes in a multihop network and to achieve significant energy savings.

- During the long periods of time during which no sensing occurs, S-MAC nodes alternate periodically between listening and sleep modes.

Periodic Listen and Sleep Operations

- One of the S-MAC design objectives is to reduce energy consumption by avoiding idle listening.
- This is achieved by establishing low-duty-cycle operations for sensor nodes.
- Periodically, nodes move into a sleep state during which their radios are turned off completely.
- Nodes become active when there is traffic in the network.
- The basic periodic listen and sleep scheme is depicted in Figure 5.7. Based on this scheme, each node sets a wake-up timer and goes to sleep for the specified period of time.

- At the expiration of the timer, the node wakes up and listens to determine if it needs to communicate with other nodes.
- The complete listen- and-sleep cycle is referred to as a frame.
- Each frame is characterized by its duty cycle, defined as the listening interval-to-frame length ratio.
- Although the length of the listening interval can be selected independently by sensor nodes, for simplicity the protocol assumes the value to be the same for all nodes.

- Nodes are free to schedule their own sleep and listen intervals.
- It is preferable, however, that the schedules of neighboring nodes be coordinated in order to reduce the control overhead necessary to achieve communications between these nodes.
- Contrary to other protocols in which coordination is achieved through a master node such as a cluster head, S-MAC nodes form virtual clusters around schedules but communicate directly with their peers to exchange and synchronize their sleep and listen schedules.

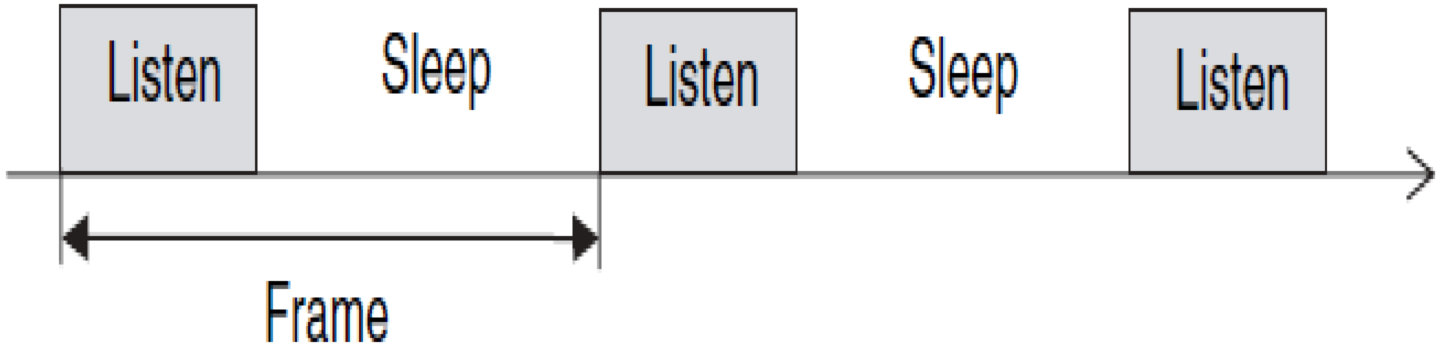


Figure 5.7 S-MAC period listen and sleep modes of operations [5.44].

Schedule Selection and Coordination

- The neighboring nodes coordinate their listen and sleep schedules such that they all listen at the same time and all sleep at the same time.
- To coordinate their sleeping and listening, each node selects a schedule and exchanges it with it neighbors during the synchronization period.
- Each node maintains a schedule table that contains the schedule of all its known neighbors.
- To select a schedule, a node first listens to the channel for a fixed amount of time, at least equal to the synchronization period.

- At the expiration of this waiting period, if the node does not hear a schedule from another node, it immediately chooses its own schedule.
- The node announces the schedule selected by broadcasting a SYNC packet to all its neighbors.
- It is worth noting that the node must first perform physical carrier sensing before broadcasting the SYNC packet.
- This reduces the likelihood of SYNC packet collisions among competing nodes.

- If during the synchronization period the node receives a schedule from a neighbor before choosing and announcing its own schedule, the node sets its schedule to be the same as the schedule received.
- The node waits until the next synchronization period to announce the schedule to its neighboring nodes.
- node may receive a different schedule after it chooses and announces its own schedule. This may occur if the SYNC packet is corrupted by either collision or channel interference.

- If the node has no neighbor with whom it shares a schedule, the node simply discards its own schedule and adopts the new one.
- On the other hand, if the node is aware of other neighboring nodes that have already adopted its schedule, the node adopts both schedules.
- The node is then required to wake up at the listen intervals of the two schedules adopted. This is illustrated in Figure 5.8.
- The advantage of carrying multiple schedules is that border nodes are required to broadcast only one SYNC packet.
- The disadvantage of this approach is that border nodes consume more energy, as they spend less time in the sleep mode.

- The neighboring nodes may still fail to discover each other, due to the delay or loss of a SYNC packet.
- To address this shortcoming, S-MAC nodes are required to perform frequent neighbor discovery, whereby a node listens periodically to the entire synchronization period.
- Nodes that currently do not have any neighbors are expected to perform neighbor discovery more frequently.

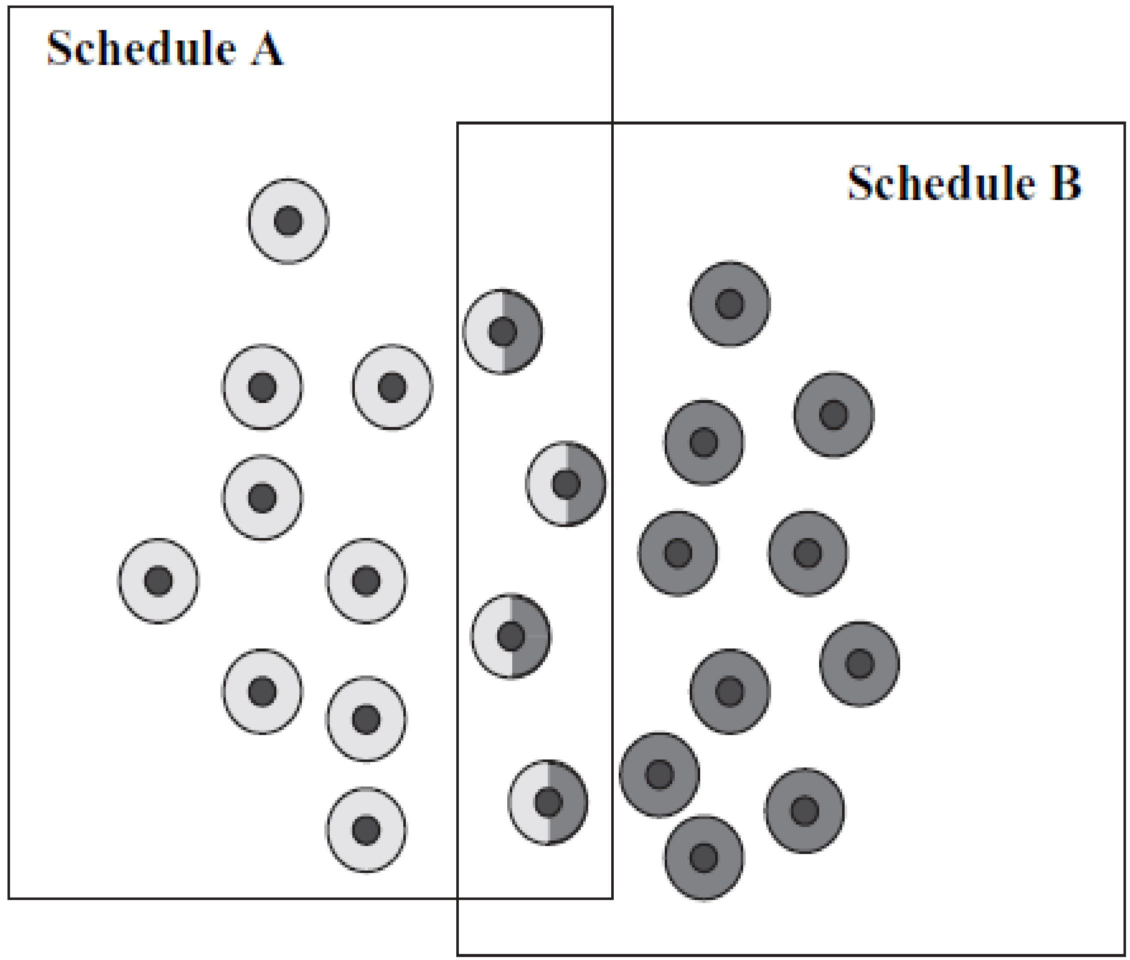


Figure 5.8 Border node schedule selection and synchronization.

Schedule Synchronization

- Neighboring nodes need to synchronize their schedules periodically to prevent long-term clock drift.
- Schedule updating is accomplished by sending a SYNC packet.
- For a node to receive both SYNC packets and data packets, the listen interval is divided into two subintervals as depicted in Figure 5.9.
- This figure illustrates three cases. In the first case the sender sends only a SYNC packet; in the second the sender sends only a data packet; and in the third the sender sends a SYNC packet in addition to the data packet.

- Access to the channel by contending nodes during these subintervals is regulated using a multi slotted contention window.
- The first subinterval is dedicated to the transmission of SYNC packets; the second subinterval is used for the transmission of data packets.
- In either of these subintervals, a contending station randomly selects a time slot, performs carrier sensing, and starts sending its packet if it detects that the channel is idle.

- Transmission of data packets uses the RTS/CTS handshake to secure exclusive access to the channel during transmission of the data.
- This access procedure guarantees that the neighboring nodes receive both the synchronization and data packets.

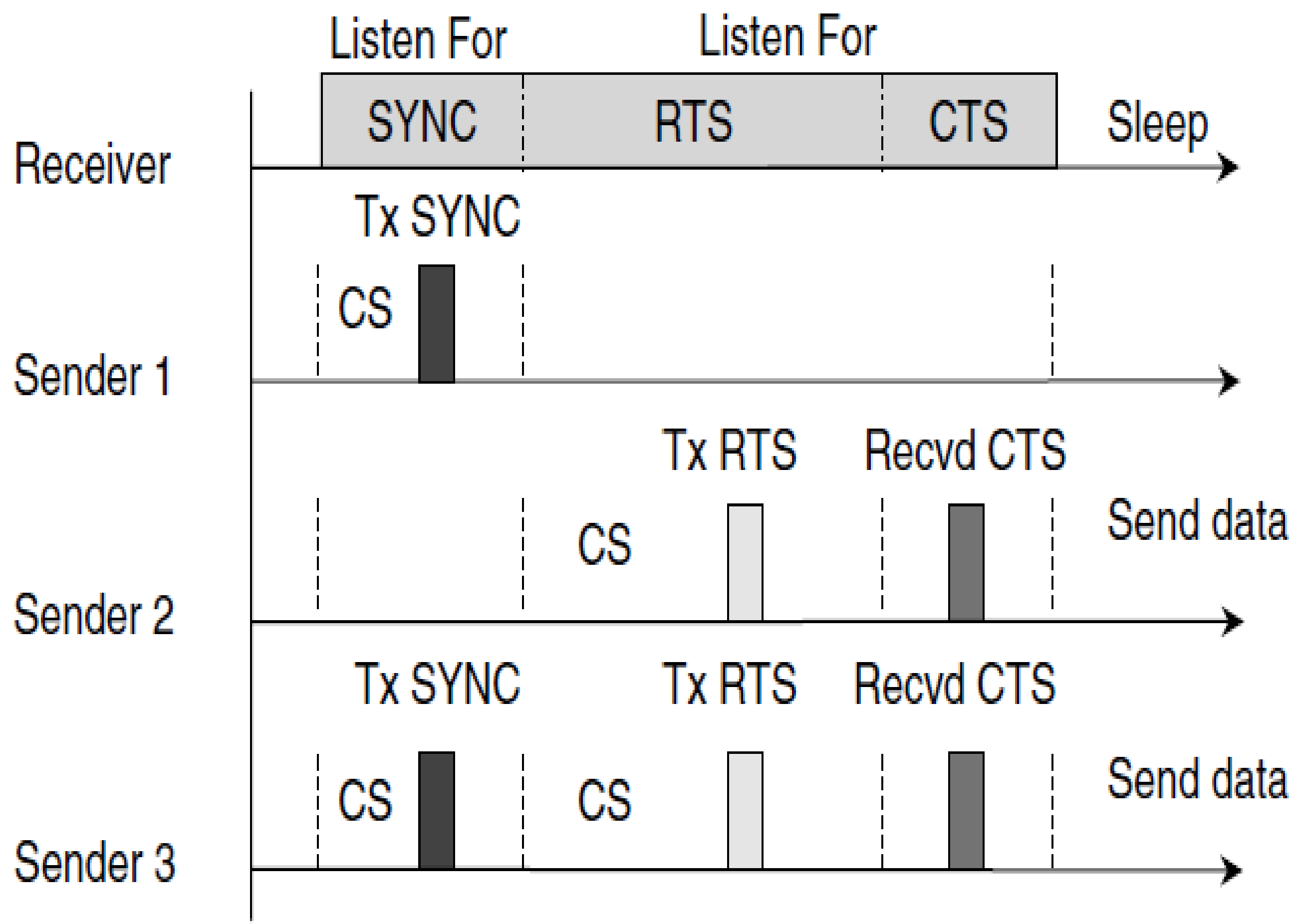


Figure 5.9 Timing relationship between a receiver and a variety of senders [5.44].

Adaptive Listening

- A closer look at the periodic listen and sleep scheme reveals that a message may incur increased latency as it is stored and forwarded between adjacent network nodes.
- If a sensor is to follow its sleep schedule strictly, data packets may be delayed at each hop.
- To address this shortcoming and improve latency performance, the protocol uses an aggressive technique referred to as adaptive listening.

- Based on this technique, a node that overhears, during its listen period, the exchange of a CTS or RTS packet between a neighboring node and another node assumes that it may be the next hop along the routing path of the overheard RTS/CTS packet, ignores its own wake-up schedule, and schedules an extra listening period around the time the transmission of the packet terminates.
- The overhearing node determines the time necessary to complete the transmission of the packet from the duration field of the overheard CTS or RTS packet.

- Immediately upon receiving the data packet, the node issues an RTS packet to initiate an RTS/CTS handshake with the overhearing node.
- Ideally, the latter node is awake, in which case the packet forwarding process proceeds immediately between the two nodes.
- If the overhearing node does not receive an RTS packet during adaptive listening, it reenters its sleep state until the next scheduled listen interval.

Access Control and Data Exchange

- To regulate access to the communication channel among contending sensor nodes, S-MAC uses a CSMA/CA-based procedure, including physical and virtual carrier
- sensing and the use of RTS/CTS handshake to reduce the impact of the hidden and exposed terminal problems.
- Virtual carrier sensing is achieved through use of the network allocation vector (NAV), a variable whose value contains the remaining time until the end of the current packet transmission.

- Initially, the NAV value is set to the value carried in the duration field of the packet transmitted.
- The value is decremented as time passes and eventually reaches zero.
- A node cannot initiate its own transmission until the NAV value reaches zero.
- Physical carrier sensing is performed by listening to the channel to detect ongoing transmission.
- Carrier sensing is randomized within a contention window to avoid collisions and starvation.

- A node is allowed to transmit if both virtual and physical carrier sensing indicate that the channel is free.
- To perform virtual carrier sensing effectively, nodes may be required to listen to all transmissions from their neighbors.
- As a result, nodes may be required to listen to packets that are destined for other nodes.
- Packet overhearing may lead to significant energy waste. To avoid overhearing, S-MAC allows nodes to move into sleep mode after they hear the exchange of an RTS or a CTS packet between two other nodes.

- The node initializes its NAV with the value contained in the duration field of the RTS or CTS packets and enters the sleep state until the NAV value reaches zero.
- Since data packets are typically larger than control packets, the overhearing avoidance process may lead to significant energy savings.
- The scheme used by S-MAC to avoid collisions is illustrated in Figure 5.10.
- A node attempting to transmit a message must first sense the channel. If the channel is busy, the node goes to sleep and wakes up when the channel becomes free again.

- If the channel is idle, a node, sending a data packet, first issues an RTS packet and waits for a CTS packet from the receiver.
- When it receives the CTS packet, the node sends its data packet. The transaction is completed when the node receives an acknowledgment from the receiver.
- After successful exchange of the RTS and CTS packets, the communicating nodes use their normal sleep time to exchange data packets.
- The nodes do not resume their regular sleep schedule until the data transmission is completed.

- Furthermore, the transmission of a broadcast packet, such as a SYNC packet, does not require the exchange of the RTS and CTS packet.

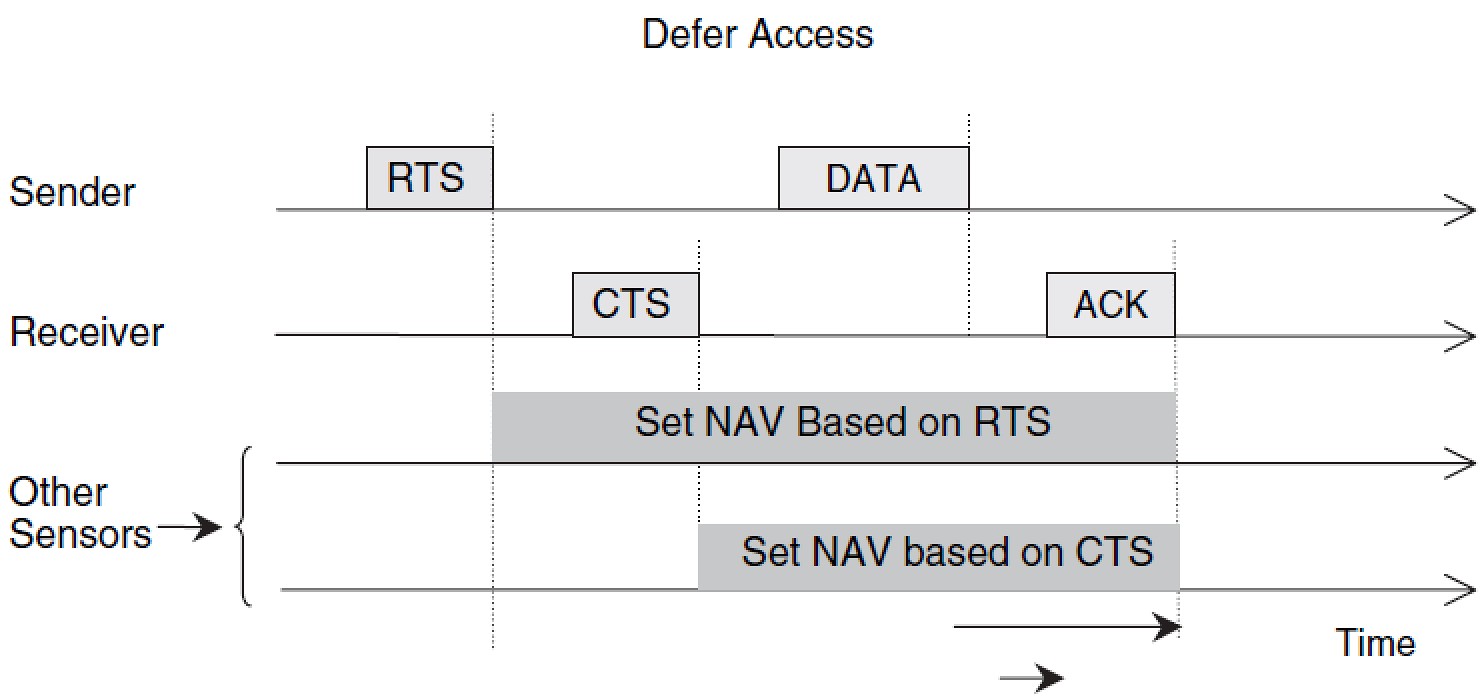


Figure 5.10 S-MAC collision avoidance scheme [5.46].

Message Passing

- To improve application-level performance, S-MAC introduces the concept of message passing, where a message is a meaningful unit of data that a node can process.
- Messages are divided into small fragments. These fragments are then transmitted in a single burst.
- The fragments of a message are transmitted using only one RTS/CTS exchange between the sending and receiving nodes.

- At the completion of this exchange, the medium is reserved for the time necessary to complete the transfer of the entire message successfully.
- Furthermore, each fragment carries in its duration field the time needed to transmit all the subsequent fragments and their corresponding acknowledgments. This procedure is depicted in Figure 5.11.
- Upon transmitting a fragment, the sender waits for an acknowledgment from the receiver.
- If it receives the acknowledgment, the sender proceeds with transmission of the next fragment.

- If it fails to receive the acknowledgment, however, the sender extends the time required to complete transmission of the segment to include the time to transmit one more fragment and its corresponding acknowledgment and retransmits the unacknowledged frame immediately.
- Sleeping nodes can hear about this extension only if they hear extended fragments or their corresponding acknowledgments.
- Nodes that only heard the initial RTS and CTS packet exchange remain unaware of the transmission extension.

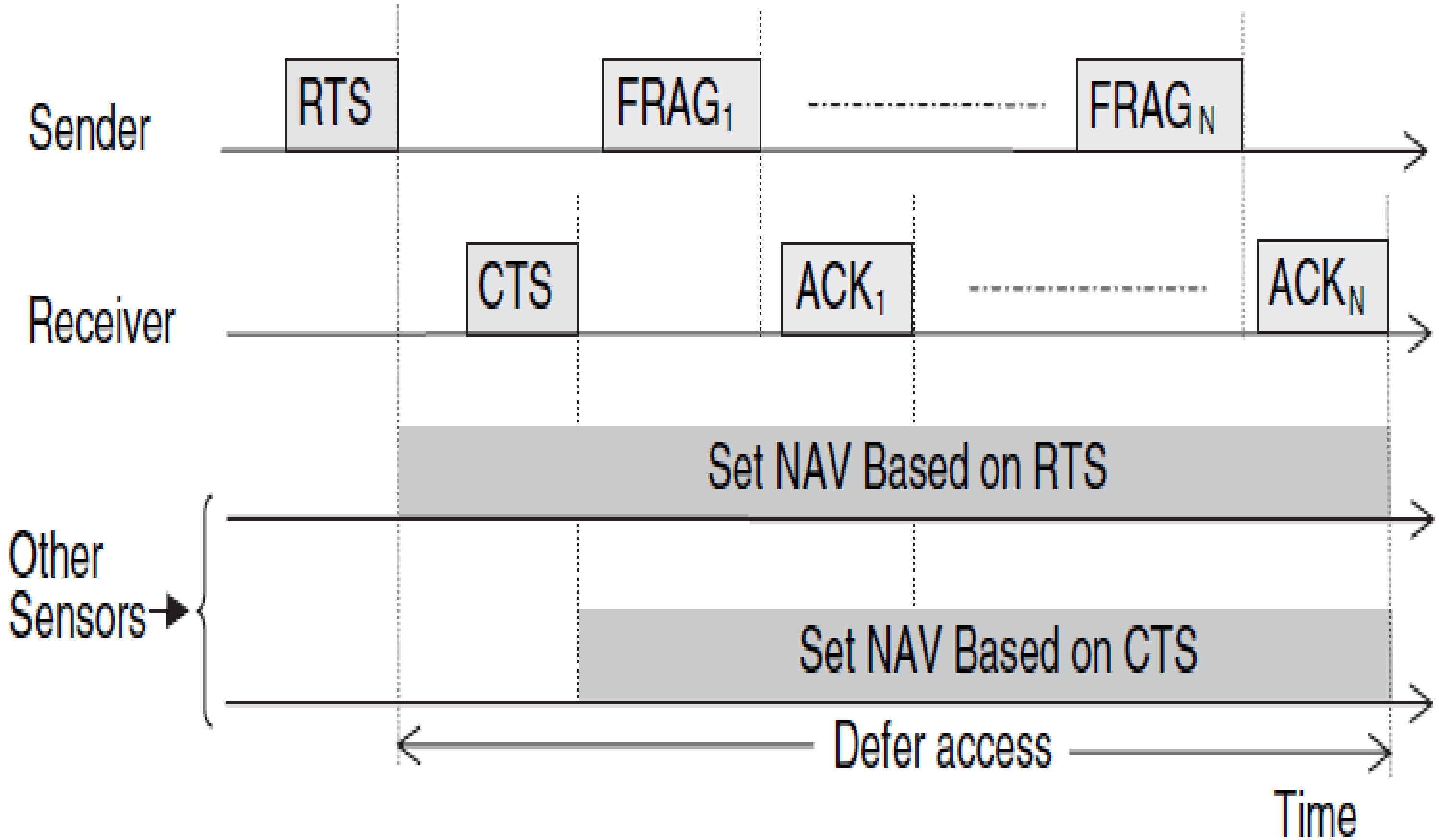


Figure 5.11 S-MAC message passing [5.46].