Lesson 2-3: The IEEE 802.11x MAC Layer

Lesson Overview

This lesson describes basic IEEE 802.11x MAC operation, beginning with an explanation of contention schemes in both Ethernet and wireless LANs. The lesson also describes the hidden node problem and discusses the two contention modes in IEEE 802.11x. The lesson concludes with an explanation of MAC-layer fragmentation.

Lesson Objectives

After completing this lesson, you will be able to:

- Describe the IEEE 802.11x MAC layer
- Explain the difference between the IEEE 802.3 and IEEE 802.11x contention schemes
- Explain how IEEE 802.11x deals with the problem of hidden nodes
- Describe DCF and PCF
- Describe fragmentation in IEEE 802.11

Introduction

There is more resemblance between the IEEE 802.11x and IEEE 802.3 MAC layers than between the IEEE 802.11x and IEEE 802.3 PHY layers. The IEEE 802.11x MAC layer uses a contention scheme for media access that is similar to the contention scheme of conventional Ethernet. Most important, however, is the fact that the packets passed from the IEEE 802.11x MAC layer to the LLC layer are identical in format to the packets passed from the IEEE 802.3 MAC layer to the LLC layer. An IEEE 802.11x wireless LAN, therefore, can interact transparently with a wired Ethernet at the LLC layer and with higher-protocol stacks such as TCP/IP.

IEEE 802.11 and IEEE 802.11b MAC Layers

MAC-layer operation for IEEE 802.11 and IEEE 802.11b is nearly identical. There are only two differences: IEEE 802.11b has three additional bits in the capability information fixed field and three new reason codes. (See Appendix C.) These differences are relevant only when a STA attempts to associate with an AP. If the STA cannot support required IEEE 802.11b PHY parameters, it will be denied association. For example, if the AP requires that STAs support 5.5 Mbps and 11 Mbps, a STA built to the IEEE 802.11 DSSS PHY standard (which cannot support these rates) will not be able to associate with the AP. However, an AP built to the IEEE 802.11b standard is compatible with STAs built to the IEEE 802.11 DSSS PHY standard when the AP is configured to support 1 Mbps and/or 2 Mbps data rates. The Intel PRO/Wireless 2011 LAN Access Point default setting supports the earlier standard.

CSMA/CA

In wired Ethernet, nodes contend for access to the physical medium by "listening" for traffic from other nodes. If a node detects that the medium is idle, it can transmit a frame. The frame usually reaches its destination, but occasionally it will collide with a frame that was sent from another node at the same time. When the nodes detect a collision, they cease transmitting their frames and "back off" before trying to resend the frames.

This process, called Carrier Sense Multiple Access with Collision Detection (CSMA/CD), works well in wired networks, but it cannot be implemented in wireless networks. CSMA/CD is possible only because a node on a wired network can transmit a frame and listen to the medium at the same time in order to determine whether the frame collided with another frame. Because the energy level of a signal transmitted over the wireless medium is much higher than that the energy level of an incoming signal, a STA cannot detect an incoming signal at the same time it is transmitting. To compensate for this difference, the IEEE 802.11x standard defines Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) as the medium access method.

Carrier Sense Mechanisms

IEEE 802.11x defines two carrier-sense mechanisms: physical and virtual. The physical carrier sense mechanism, provided by the PHY, is similar to the carrier sense mechanism used by Ethernet. The STA listens to the medium by sensing the power level of the medium: if the power level is below a particular threshold, the medium is idle.

The virtual carrier sense mechanism is implemented by the MAC using a network allocation vector (NAV). The NAV is a value that a STA continuously updates using the value in the duration field of the last frame transmitted. The value of the NAV is then decremented until it reaches zero. When the value of the NAV is zero and the physical carrier sense mechanism registers no signal, the medium is considered idle.

Hidden Nodes

Using a virtual carrier sense mechanism solves several problems. The first is the problem mentioned above: wireless transmitters can rarely receive and transmit at the same time, so a physical carrier sense mechanism will not always be sufficient. In wireless networks there is also another problem called the "hidden node" problem. *Figure 2-19* illustrates a hidden node.

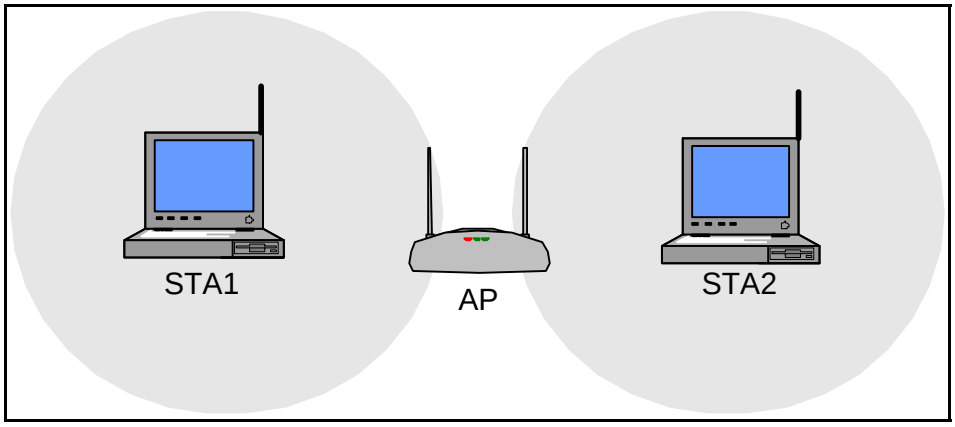

Figure 2-19: Hidden Nodes

In *Figure 2-19*, STA1 and STA2 are in range of the same AP, but they are out of range of each other; therefore, neither STA can detect signals the other transmits. In this case, STA1 could physically sense a free medium while STA2 is transmitting. If STA1 transmits a frame at the same time STA2 is transmitting, the frames will collide near the AP. To solve the hidden node problem, an optional feature of IEEE 802.11x allows network administrators to require that before transmitting, a STA will send a request to send (RTS) frame. In response, an AP will send a clear to send (CTS) frame.

With RTS/CTS enabled, STA1 transmits an RTS frame, which STA2 does not detect. STA2, however, detects the CTS frame sent by the AP. STA2 will then update the value in its NAV according to the value in the duration field of the CTS frame. This value is high enough that, by the time the NAV value of STA2 counts down to zero, STA1 will have finished its transmission.

To enable RTS, the RTS threshold must be configured. The default value of this parameter is 2347 bytes. All frames larger than the RTS threshold are required to be preceded by an RTS frame. Because 2347 is larger than the largest frame permitted, this default value disables RTS for all transmissions. When RTS is desired, the RTS threshold should be set to a number lower than 2347. It is not recommended that all frames be preceded by an RTS frame, however, because the increased overhead will significantly reduce wireless LAN performance.

You can expect to encounter a hidden node problem only when two or more stationary STAs consistently transmit from the edges of the AP's range. In this case, you may have placed your APs improperly. (See Module 5: Conducting a Site Survey.)

Frame Exchange Protocol

Because STAs cannot detect collisions directly and because of the unpredictability of the wireless medium, it is possible that the destination STA will not receive a frame sent to it. To prevent frames from being lost, the IEEE 802.11x standard requires that every sent frame be acknowledged by the recipient. If the sending STA does not receive an acknowledgement (ACK) frame from the recipient, the sending STA resends the same frame until an ACK is received or until the retry counter expires.

There are two frame exchange protocols in IEEE 802.11: the two-way handshake and the four-way handshake. The two-way handshake consists of a data frame and an ACK frame. Between these frames, no other STA can transmit. The four-way handshake consists of an RTS frame, a CTS frame, a data frame, and an ACK frame. Again, no other STA can transmit during this handshake.

Timing Intervals

When a STA in a wired Ethernet network detects an idle medium, it must wait one "slot time" before transmitting. IEEE 802.11x also requires that STAs wait a certain amount of time after detecting an idle medium before transmitting. However, in IEEE 802.11x there are four "interframe spaces" (IFSs). The duration of each IFS is relative to two units: the short IFS (SIFS) and the slot time. The duration of each IFS also depends on the IEEE 802.11x PHY that is used. For the IEEE 802.11b PHY, the SIFS is 10 microseconds (μs), which is a function of four variables: the receiver delay, the delay in decoding the PLCP preamble and header, the transceiver turnaround time, and the MAC processing time. The slot time is 20 μs. *Figure 2-20* shows each IFS relative to the slot time and the SIFS.

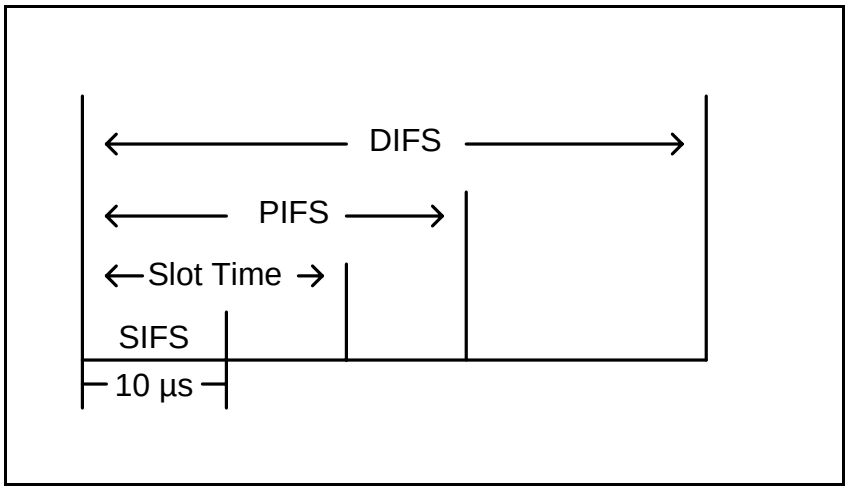

Figure 2-20: Interframe Spaces for IEEE 802.11b PHY

The following is an explanation of the IFSs.

- **Slot Time**
  The slot time represents six variables: the receiver delay, the delay in decoding the PLCP preamble and header, the carrier-sensing time, the transceiver turnaround time, the signal propagation delay, and the MAC processing delay. The backoff interval is defined in multiples of the slot time.
- **SIFS**
  The shortest of the IFSs, the SIFS allows some frames to be sent quickly before other STAs can detect an idle medium. ACK and CTS frames are sent one SIFS after the last frame received, for example. Fragmented frames are also sent with one SIFS between each fragment.

- **Point Coordination Function IFS (PIFS)**
  The PIFS is used by STAs during the contention-free period so that they can get priority access to the medium. The PIFS equals one slot time plus one SIFS.
- **Distributed Coordination Function IFS (DIFS)**
  The DIFS is used during the contention period. Because it is the longest IFS, frames sent after this interval have lower priority than those sent after PIFSs or SIFSs. The DIFS equals one SIFS plus two slot times.

Medium Access Methods

Unlike wired Ethernet, which operates only in a contention mode, IEEE 802.11x can operate in either a contention mode or an optional contention-free mode. The contention mode is called the distributed coordination function (DCF). During this mode, CSMA/CA is the protocol used to access the medium. The contention-free mode is called point coordination function (PCF). During this mode, access is controlled by a point coordinator (PC), which always resides in an AP. STAs send frames when polled by the PC instead of competing for the medium.

DCF Protocol

Frame transmission during DCF proceeds as follows:

- The MAC checks the virtual and physical carrier senses for an idle medium.
- If the medium is idle for one DIFS, the MAC begins to transmit the frame.
- If the medium is not idle for one DIFS, the MAC selects a random backoff interval and increments the retry counter.
- The backoff interval falls within the parameters of the “contention window,” which is similar to the contention window of Ethernet.
- The backoff interval begins to count down after the end of the next DIFS.
- Each time the MAC detects an idle medium for the space of one slot time, the backoff value is decremented by one.
- If the backoff interval expires and the medium is still idle, the MAC can begin transmission.
- If, after transmitting a frame, an ACK is not received, the MAC “detects” a collision; the contention window is then doubled, a new backoff interval is selected, and the backoff countdown begins again.

- This process continues until the frame is transmitted successfully (an ACK is received) or the transmission is cancelled (the retry counter expires).

Figure 2-21 illustrates the basic DCF process when two STAs (STA1 and STA3) each transmit a frame to STA2.

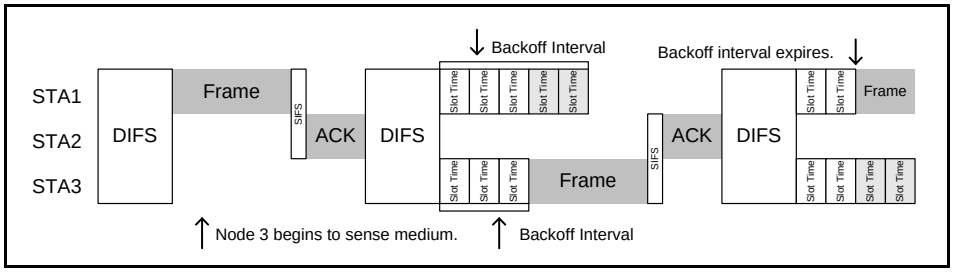

Figure 2-21: Successful Transmission of Two Frames to STA2

In *Figure 2-21*, STA1 transmits a frame after sensing an idle medium for one DIFS. When STA2 receives the frame, it transmits an ACK after one SIFS. Because STA2 has to wait only one SIFS from the end of the frame it received before transmitting the ACK, no other STA senses an idle medium before the ACK is sent. Also, the value in the duration field in the frame from STA1 sets the NAVs in all other STAs to a number high enough to allow the transmission of the ACK.

STA3 begins to sense the medium while the frame from STA1 is being transmitted. Because it registers the medium as busy, STA3 selects a random backoff interval, which is expressed as a multiple of slot times. After the end of the next DIFS, STA3 continues to wait for the duration of the backoff interval before attempting to transmit a frame. As each slot time passes, the backoff interval decreases by one if the medium is idle during that slot time.

STA1 must also select a random backoff interval after transmitting a frame successfully. At the end of the DIFS, both STA1 and STA3 begin the backoff countdown. Because STA3 chose a backoff interval that is shorter than the interval STA1 chose, STA3 is able to access the medium first. When STA3 begins to transmit, STA1 physically senses the transmission and discontinues its backoff countdown. After the next DIFS, STA1 resumes the backoff countdown until it reaches 0, at which time STA1 is free to transmit another frame.

Figure 2-22 shows an infrastructure BSS with one AP and three STAs. All STAs are within range of the AP. STA1 and STA2 are in range of each other, but STA3 is out of range of STA1 and STA2. To avoid the hidden node problem, the network administrator has enabled RTS/CTS.

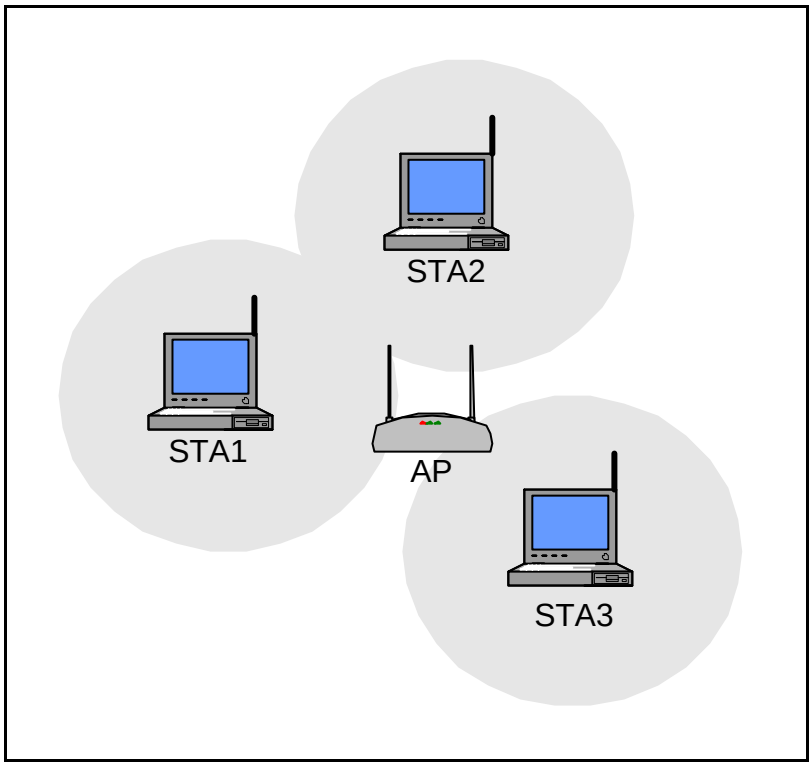

Figure 2-22: A Possible Hidden Node Problem

Figure 2-23 illustrates the DCF protocol for the example in *Figure 2-22*.

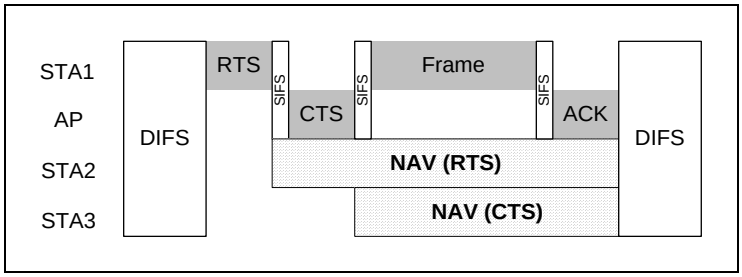

Figure 2-23: Successful Transmission of One Frame to the AP Using RTS/CTS

In *Figure 2-23*, STA1 sends an RTS frame to the AP after sensing an idle medium for one DIFS. The value in the duration field of the RTS is high enough to allow the transmission of a CTS, a data or management frame, an ACK, and three SIFSs. STA2, upon receiving the RTS, updates its NAV so that its virtual carrier sense mechanism will detect a busy medium until after the AP transmits the ACK. STA3, however, does not receive the RTS frame and does not update its NAV.

When the AP transmits the CTS, STA3 receives the CTS and updates its NAV to the duration value in the CTS frame. This value represents the amount of time it takes to transmit a data or management frame, an ACK, and two SIFS intervals. In this way, STA1 can send its frame with little chance of a collision with another frame because all other STAs have been "disabled" during that time. The ACK frame transmitted by the AP has a duration value of 0, which causes all the STAs to reset their NAVs to 0. Contention for the medium begins again after one DIFS.

It is possible that after the first DIFS, STA3 could begin to transmit an RTS frame at the same time as STA1. If this should happen, the two RTS frames would collide and become corrupted before reaching the AP. The AP would therefore not send a CTS frame to either STA. After waiting a preset amount of time for the CTS to arrive, each STA would select a random backoff interval before attempting to send another RTS frame. Because the intervals would probably be different, one of the STAs would gain control of the medium, and the other STA would have to defer transmission until after its NAV interval and backoff interval counted down to zero.

PCF Protocol

The nature of the DCF protocol makes it less than ideal for time-bounded services such as streaming video or audio. To accommodate time-bounded data, IEEE 802.11x includes the optional PCF, a centrally controlled access mechanism that allows STAs to send frames without having to contend for access to the medium. When they associate with the AP, STAs can request that they be polled during the contention-free period (CFP).

During the CFP, the following events occur:

- The PC obtains control of the medium by sending a beacon frame after one PIFS. The beacon contains a large value in the duration field. All STAs in the BSS set the value in their NAVs to this value, which prevents their virtual carrier sense mechanisms from detecting an idle medium during the CFP.
- The PC delivers frames to the STAs in its polling list, one at a time, and the STAs send ACK frames in return.
- The PC sends a contention-free poll (CF-Poll) frame to each STA that has requested contention-free service.
- For each CF-Poll frame received, a STA may send one frame to the PC. The PC will acknowledge this frame.
- If a polled STA does not have a frame to send, it will not respond to the poll.
- To increase efficiency during CFP, some frames can be combined. For example, a PC can combine the CF-Poll frame with an ACK frame and acknowledge the receipt of a frame from one STA while simultaneously polling another STA.
- During CFP, the maximum interval between frames is the PIFS, which prevents STAs operating under DCF from accessing the medium.
- When the PC sends a frame to a STA, it waits one SIFS for a response. If it does not receive a response, it sends its next frame after waiting one PIFS interval from the end of the last frame sent.
- The CFP ends when the PC sends a CF-End frame. This frame has 0 in its duration field, which causes all STAs to reset their NAVs so that DCF can resume.

Figure 2-24 illustrates the PCF protocol during CFP.

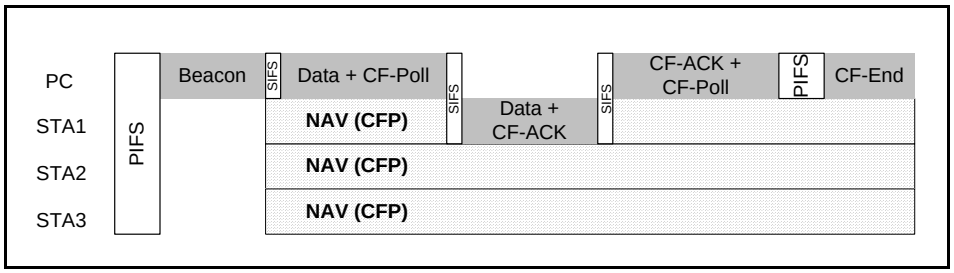

Figure 2-24: The PC polls STAs during CFP so that they do not have to contend for access to the medium.

In *Figure 2-24*, STA1 and STA2 requested that they be polled during CFP. STA3 requested that it not be polled during CFP. At the beginning of CFP, the PC gains control of the medium by transmitting a special beacon after one PIFS. All of the STAs reset their NAVs to the estimated length of the CFP, which is contained in the CF parameter set of the beacon frame. (See Appendix C.) Note that even though all the STAs register the medium as busy through their virtual carrier sense mechanisms, during CFP they are able to respond to polls from the PC.

The PC then transmits a data + CF-Poll frame to STA1 after one SIFS. STA1 responds after one SIFS with a data + CF-ACK frame, thereby acknowledging the data sent by the PC and simultaneously sending a data frame in response to the CF-Poll. One SIFS after receiving the frame from STA1, the PC transmits a CF-ACK + CF-Poll frame. This frame acknowledges the frame sent by STA1 and at the same time polls STA2. Because STA2 does not have any data to send, it does not respond to the poll. When the PC does not receive a response from STA2 after one SIFS, it transmits a CF-End frame after one PIFS. (If STA2 had sent a data frame in response to the poll, the PC could send a CF-End + ACK.) Because the CF-End frame contains a zero in the duration field, the STAs reset their NAVs to zero and resume DCF.

DCF and PCF alternate when the PCF option is set. The length of PCF can vary depending on how many STAs are on the polling list. Also, QoS options can be set to give some polled STAs more opportunities than others to transmit during PCF. Each DCF period must be long enough to send at least one maximum-length frame and its acknowledgement.

Fragmentation

IEEE 802.11x contains a provision for fragmenting MAC Service Data Units (MSDUs) prior to transmission. Any MSDU larger than the value indicated by the fragmentation threshold (a management information base [MIB] attribute) is fragmented according to the following rules:

- All MSDU fragments will be the same length except the last fragment, which may be shorter.
- The length of a fragment will always be an even number of octets except the last fragment, which may be either an even or an odd number of octets.
- The length of a fragment will not be longer than the fragmentation threshold, unless WEP is enabled. (If WEP is enabled, a fragment may be slightly larger than the fragmentation threshold.)
- Each fragment of the MSDU will be encapsulated in its own MAC-layer frame.
- Each fragment will be acknowledged by the recipient.
- The fragments of each MSDU will contain the same sequence number in the sequence control field of the MAC header.
- Each fragment will be numbered sequentially, using the fragment number field of the MAC header.
- Each fragment except the last will have the more fragments bit set in the MAC header. (See Appendix C.)
- Any fragment that is not acknowledged will be retransmitted.

When a STA transmits a fragmented MSDU, it transmits the fragments one after the other in a "fragment burst." A STA that needs to transmit a fragmented MSDU contends for the medium only once: each fragment and its acknowledgement is separated by one SIFS, thereby ensuring that no other STA can interrupt the fragment burst, as shown in *Figure 2-25*.

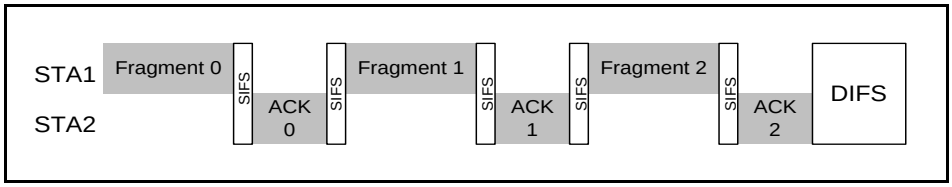

Figure 2-25: Fragment Burst

When an ACK frame follows a fragment with the more fragments bit set, the value in the duration field of the ACK is a nonzero value. (When the more fragments bit is not set, the value in the duration field of an ACK frame is zero.) This nonzero value causes all other STAs to update their NAVs so that one more frame and its acknowledgement can be transmitted, as if the ACK frame were a CTS frame, as shown in *Figure 2-26*.

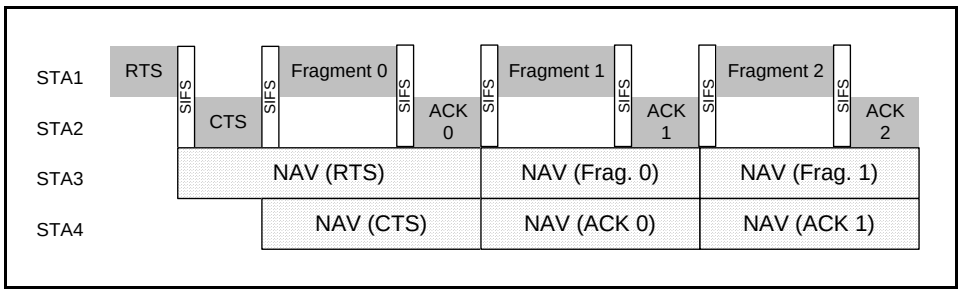

Figure 2-26: Fragment Burst with RTS/CTS

In *Figure 2-26*, RTS/CTS is enabled. STA1 is transmitting a fragmented MSDU to STA2. STA3 is in range of STA 1; STA4 is in range of STA2. The value in the duration field of the RTS frame is always large enough to allow transmission of a CTS frame, a data frame, and an ACK. When there are multiple fragments to transmit, the values in the duration fields of the fragments and their acknowledgements (Fragment 0, ACK 0, Fragment 1, ACK 1, etc.) are large enough to allow the next frame and its acknowledgement to be sent. For example, the value in the duration field of Fragment 0 is long enough to allow ACK 0, Fragment 1, and ACK 1 to be transmitted. STA3 updates its NAV using the duration value in Fragment 0 so that it will not attempt to transmit during the fragment burst. Likewise, the value in the duration field of ACK 0 is large enough for Fragment 1 and ACK 1 to be transmitted. STA4 therefore updates its NAV with the duration value in ACK 0 so that it will not attempt to transmit during the fragment burst.

The receiving STA reassembles the fragments by assembling all the frames with the same sequence number according to the value in the fragment number field. Any duplicate fragments are discarded.

