CHAPTER 3

Proposed Mechanisms

3.1 Problem

Both DCF and EDCA use random contention window to determine bakeoff intervals. Once a transmission is failed, it results in a larger contention window. This may cause two problems. First, video packets are still transmitting while deadlines were due. This will waste network bandwidth in transmitting invalid packets. Second, I-slice is the most critical slice for P/B-slice to refer to. As the number of stations increases, the number of collisions, average delay time and packet loss rate increase, too. Since the lower layer cannot distinguish the type of slices, no matter I-slice, P-slice, and B-slice (packet) would have the same packet loss rate. This may cause poor video quality.

3.2 Methodology

3.2.1 Get I/P/B information for slice to MAC layer

We propose a MAC-centric cross-layer architecture to resolve these problems. In the framework Transport layer retrieves slice type and deadline from APP layer and encapsulates them into packet header. MAC layer then retrieves this encapsulated information from the packet header (Figure 3.1).

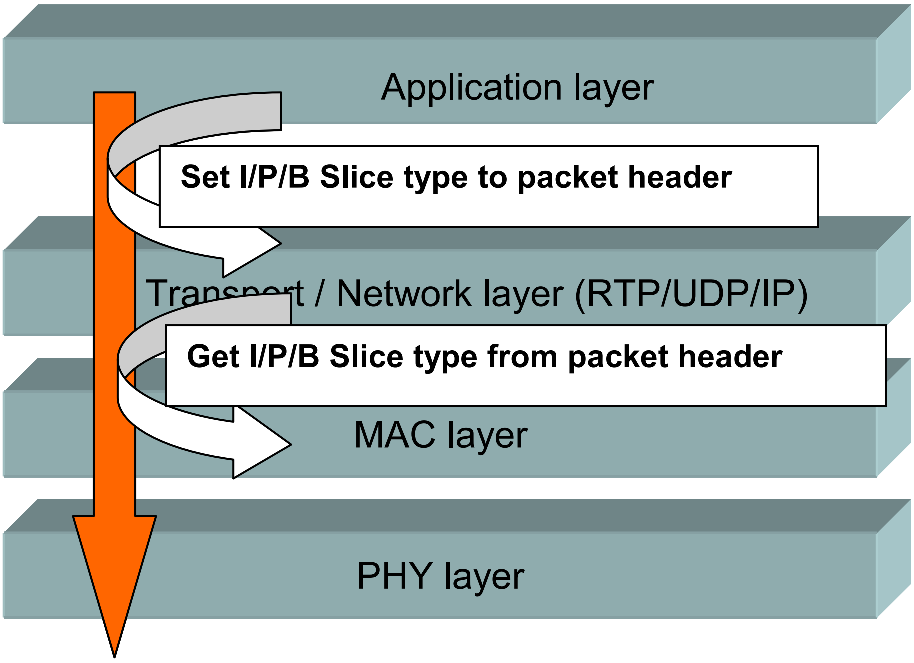

Figure 3.1: MAC-centric cross-layer architecture.

3.2.2 Don't waste too much time in backoff interval

The packet retransmission of IEEE 802.11/802.11e MAC layer is based on retry count as Figure 3.2 shows. If a packet transmission is failed then the retry count is increased by one. If the retry count is greater than retry limit then the packet is discarded. Otherwise, the packet is retransmitted. In this paper we propose to use retransmission deadline instead of retry count as Figure 3.3 shows. The packet waiting time in a queue can be derived by deducting sending time from current time. If waiting time is greater than retransmission deadline then the packet is discarded. Otherwise, the packet is retransmitted. This deadline is used to avoid sending overdue packets and deliver higher priority packets earlier. The sequence of video frames should be determined at the time of encoding. This sequence and packet priority are used to determine the retransmission deadlines of different types of frames. We assume the sequence is IPBBPBB and the time interval between video frames is λ then the retransmission deadlines of I-frame, P-frame, and B-frame are 7λ, 3λ, and λ, respectively. Figure 3.4 shows an example of retransmission intervals of frames.

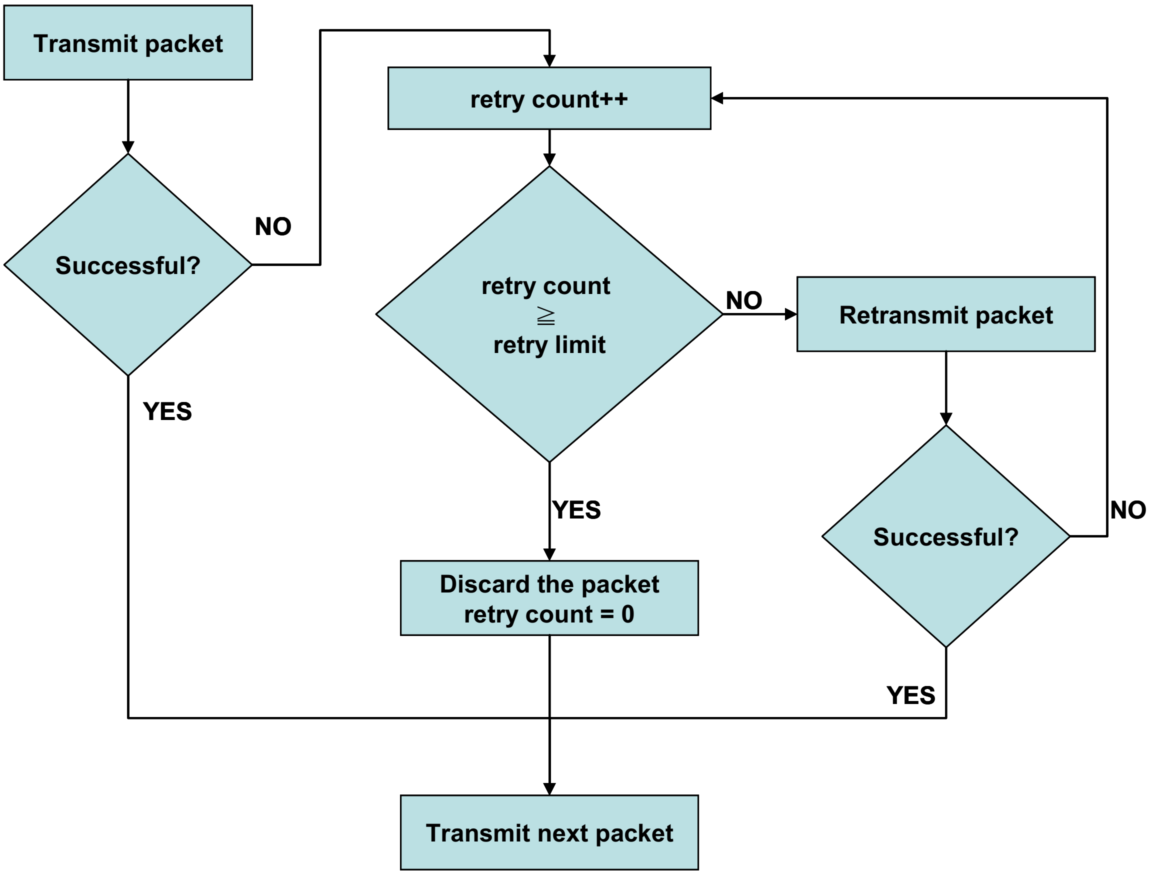

Figure 3.2: Packet retransmission for IEEE 802.11/802.11e.

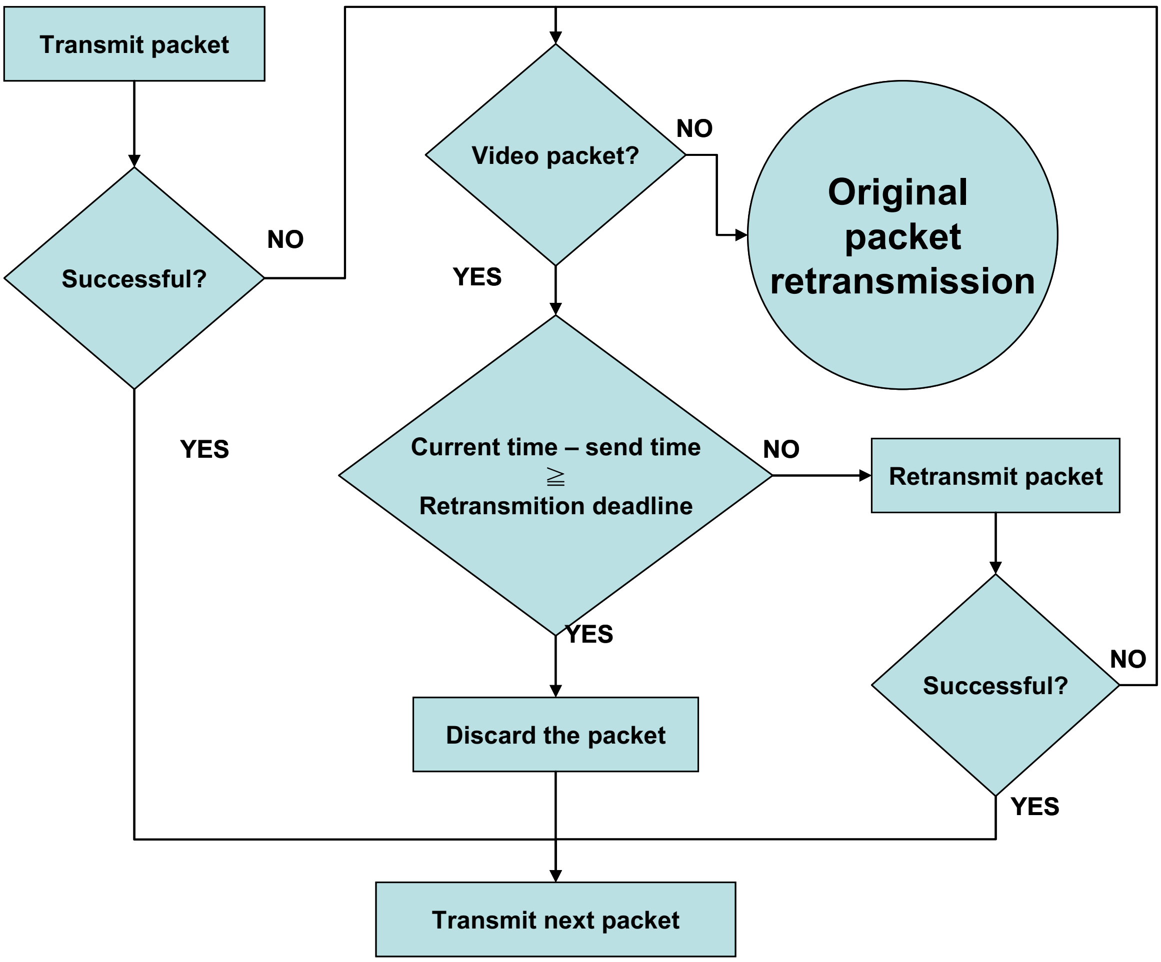

Figure 3.3: Proposed packet retransmission.

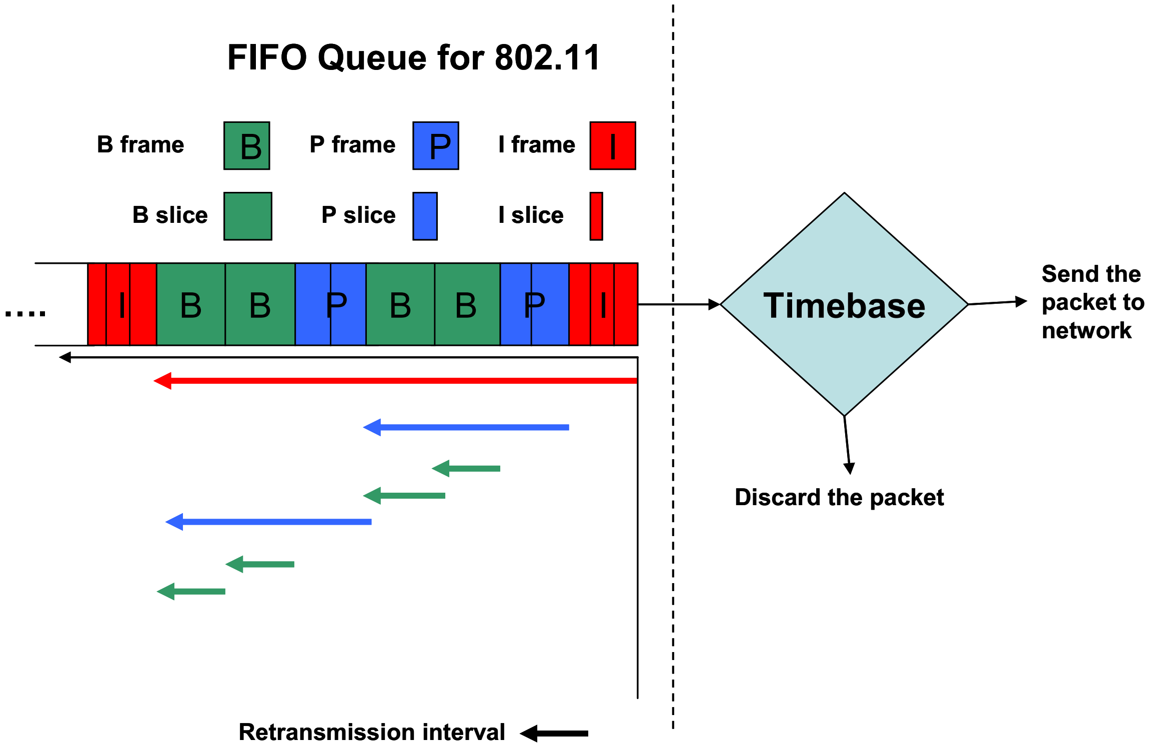

Figure 3.4: Timebase for 802.11.

For example, we analyze the H.264 video file of Foreman and Stefan for 30 FPS (Frame Per Second) and 352×288 CIF (Common Intermediate Format). Figure 3.6 and Figure 3.7 show the numbers of packets per video frame for I//P/B frames, respectively. Since I frame has the mort reference information, we see that I frame has the largest number of packets. In Forman, numbers of packet are around 10 to 20 for I frame, 2 to 7 for P frame, and 1 to 2 for B frame. And in Stefan, numbers of packet are around 21 to 23 for I frame, 4 to 13 for P frame, and 2 to 4 for B frame.

Figure 3.5: Foreman and Stefan [17].

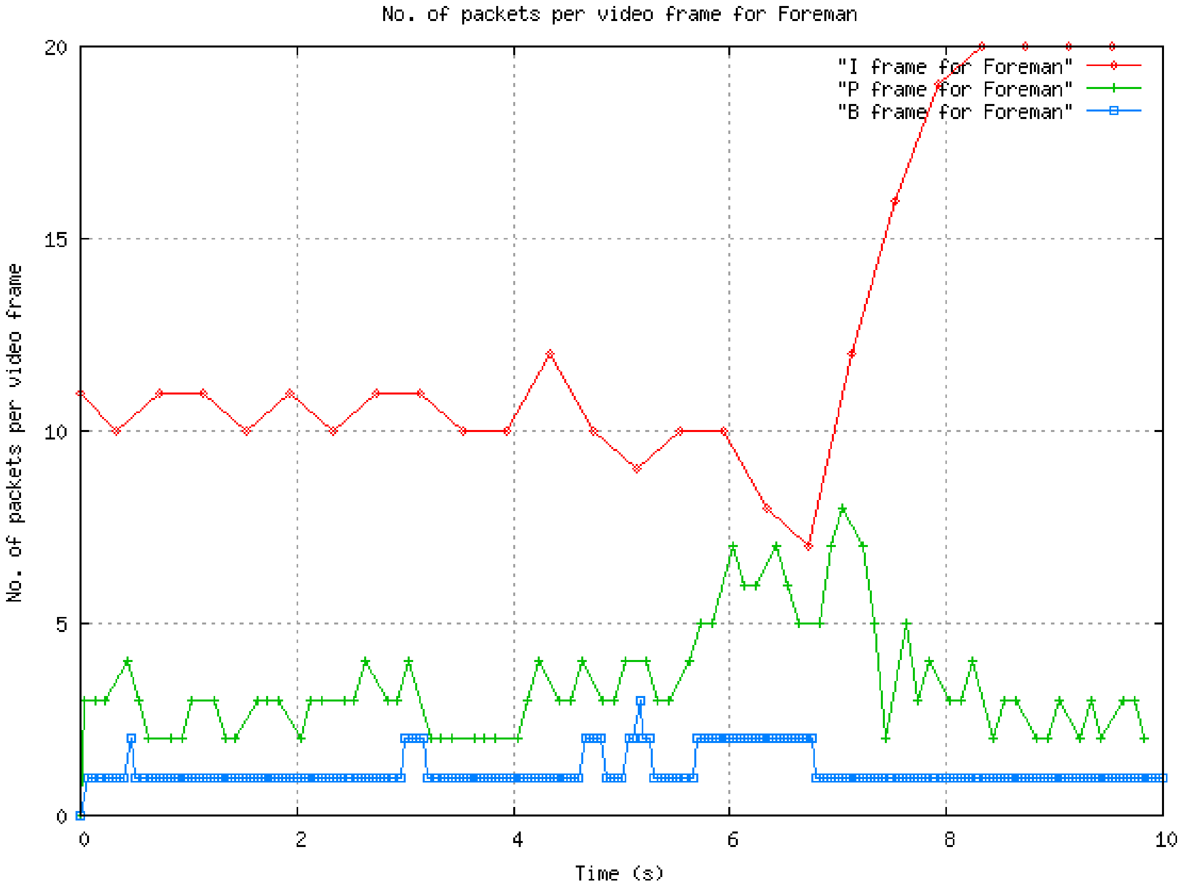

Figure 3.6: Number of packets per video frame for Foreman.

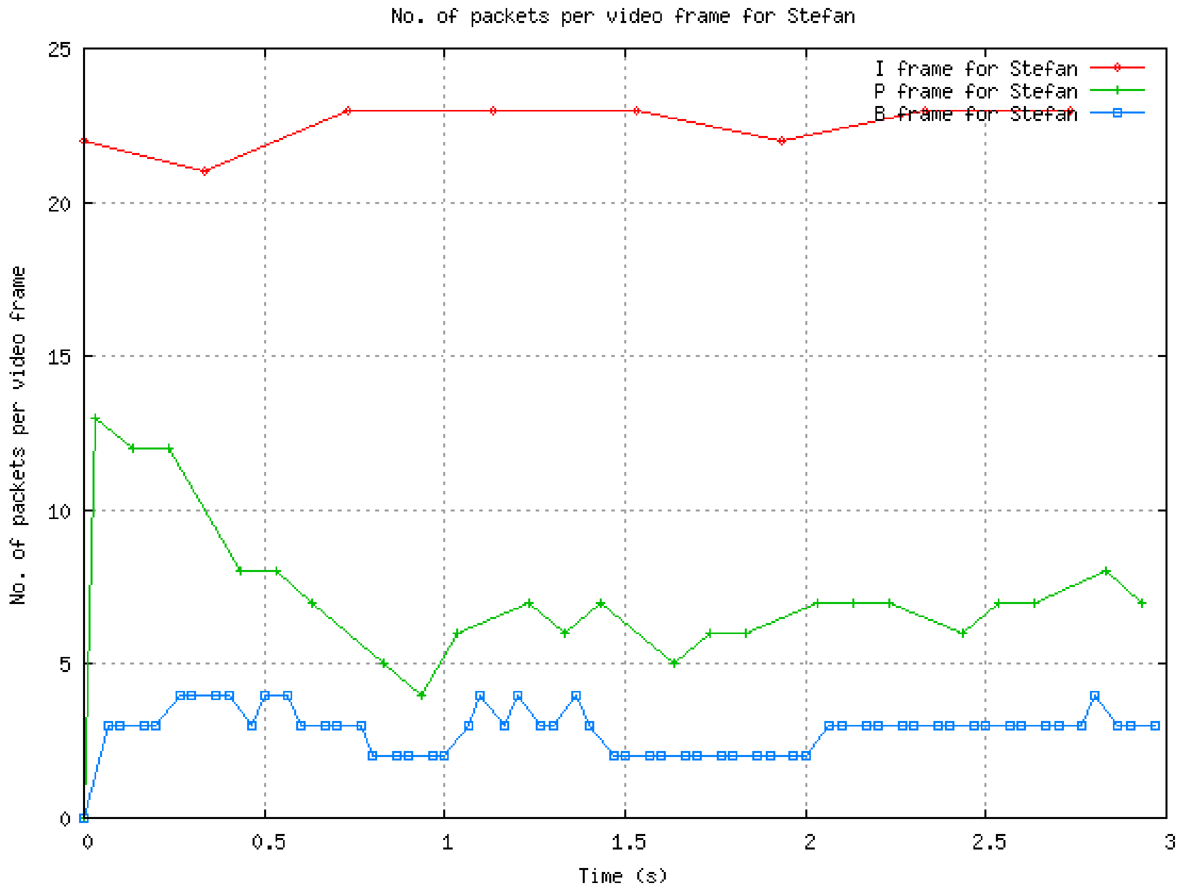

Figure 3.7: Number of packets per video frame for Stefan.

3.2.3 Single-video multi-level queue

IEEE 802.11e EDCA has priority multi-level queue, AC (Access Category). The priorities of ACs are AC_VO > AC_VI > AV_BE > AV_BK. If most of the transmissions are video packets of the same QoS level then the advantage of priority multi-level queue will no more exist. Hence we propose a single-video multi-level queue by creating four access categories, AC_1 (highest priority), AC_2 (second priority), AC_3 (third priority) and AC_4 (lowest priority). I-slice, P-slice and B-slice packets are assigned to AC_1, AC_2, and AC_3, respectively. AC_4 is dedicated for non-video traffic.

AC	Access Category
AC_BK	Background
AC_BE	Best Effort
AC_VI	Video
AC_VO	Voice

Table 3.1: Access Category for IEEE 802.11e.

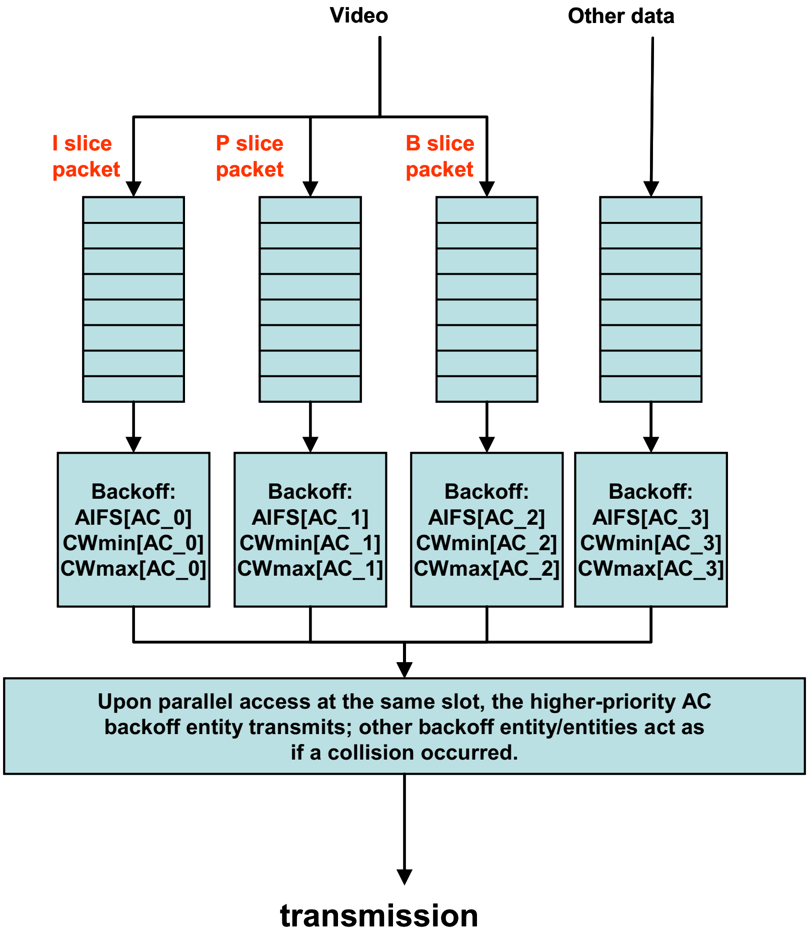

Figure 3.8: Single-video multi-level queue.

3.2.4 Hybrid mode

We propose a hybrid design framework for IEEE 802.11e as shown in Figure 3.10. This framework is based on the following three steps (Figure 3.9). At the first step, we retrieve the information of I/P/B slice type through cross layer. These information is used for testing in both IEEE 802.11e Timebase and IEEE 802.11e single-video multi-level queue methods in the second step. In the third step we test the hybrid mode of the above two methods for IEEE 802.11e. In Figure 3.11 we propose a QoS framework of video streaming for IEEE 802.11e. This framework is an integration of retransmission deadline, retry limit and single-video multi-level queue for IEEE 802.11e. The advantages stem from saving waiting time in backoff interval and prioritizing packet (slice) delivery.

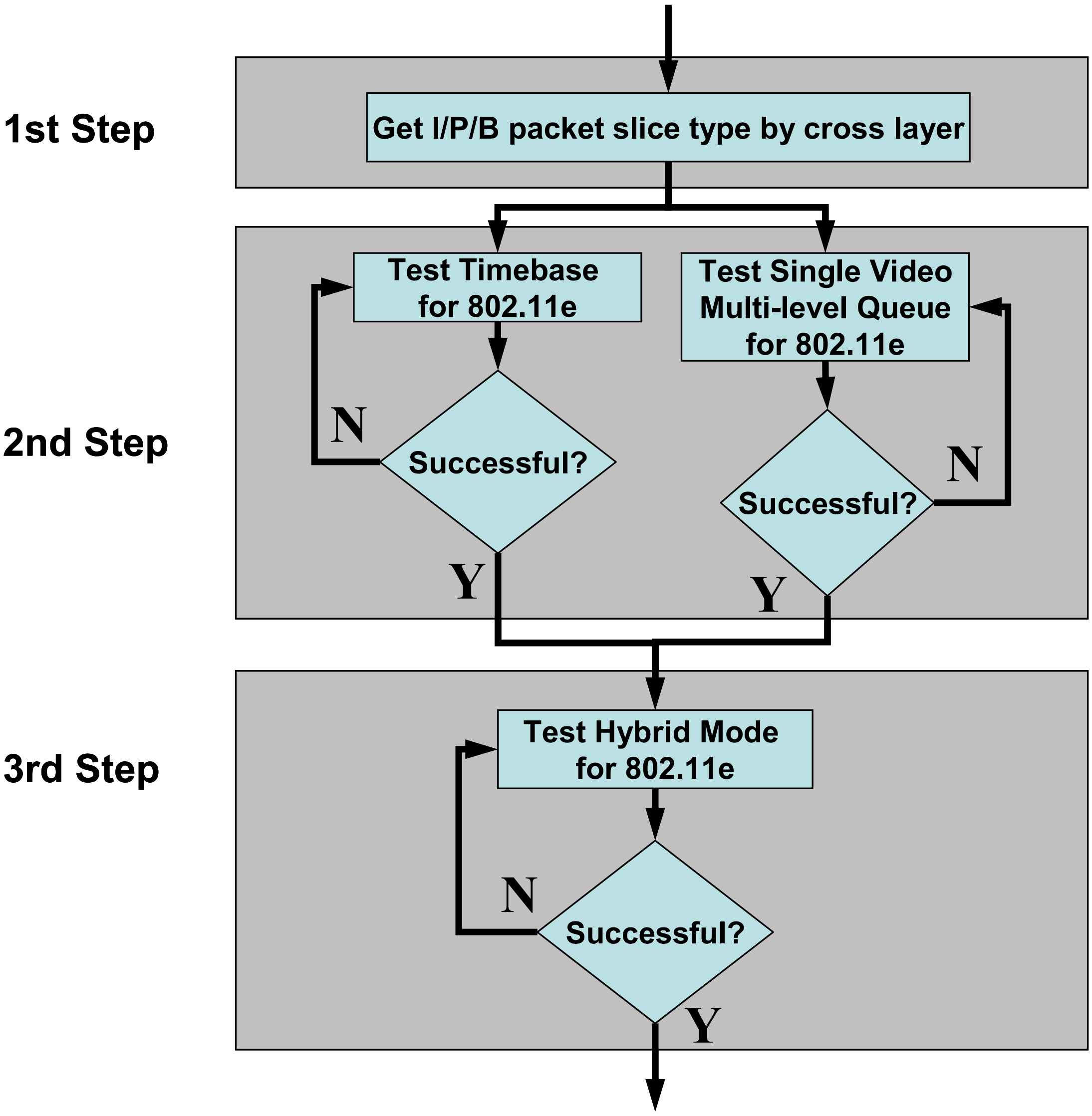

Figure 3.9: Hybrid design framework for IEEE 802.11e.

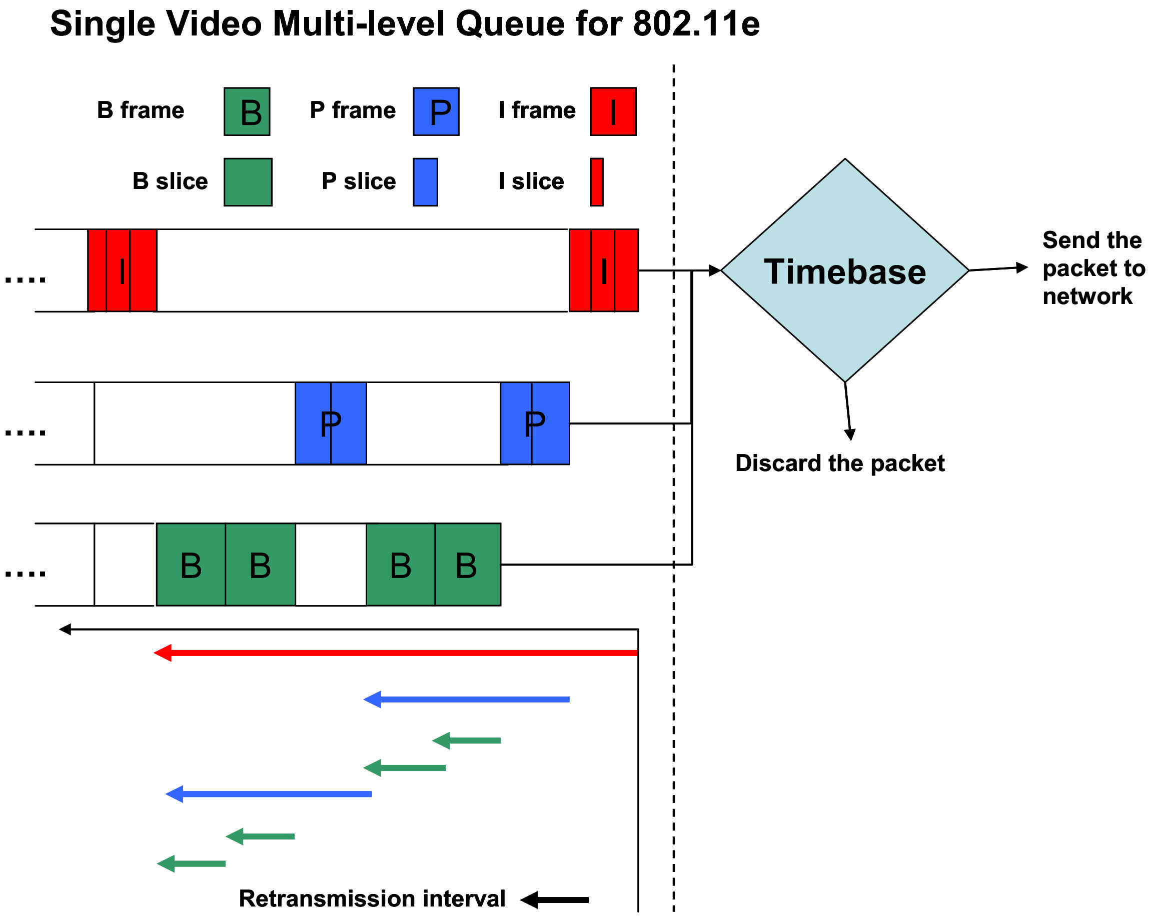

Figure 3.10: Hybrid mode for 802.11e.

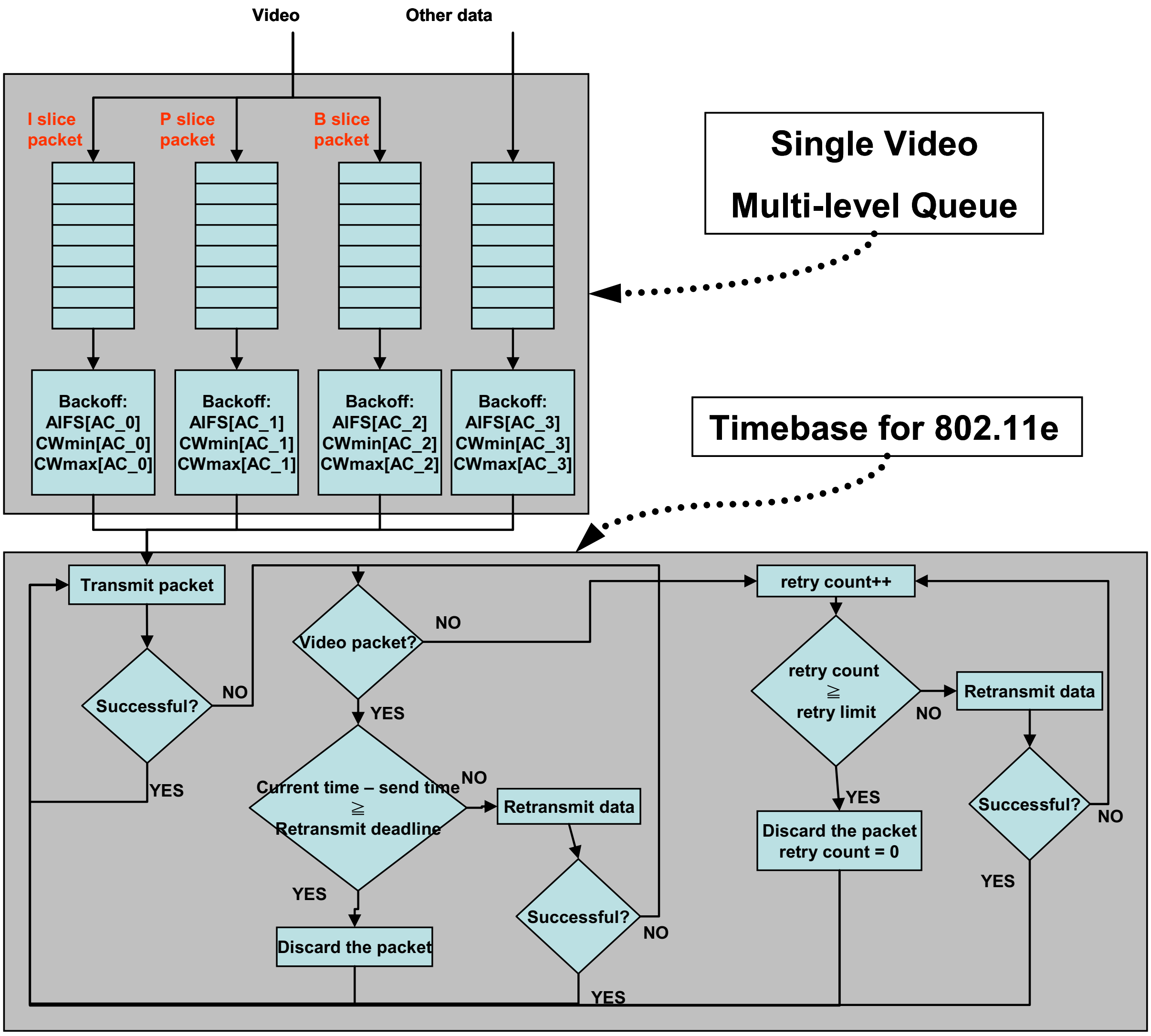

Figure 3.11: QoS framework of video streaming for IEEE 802.11e.

3.2.5 Calculate the retransmission time

We use the following example to show how our proposed retransmission deadline can save waiting time. In IEEE 802.11 the successful packet transmission time, $T_R^m(L)$, can be calculated using the following formula (3.1) and (3.2):

$$T_R^m(L) = \sum_{i=0}^{R} (T_{DeferAccess}^m(L) + \frac{\min(2^i(CW_{\min}+1)-1, CW_{\max})}{2} \cdot (p \cdot T_{DeferAccess}^m(L) + aSlotTime)) \quad (3.1)$$

$$T^m_{DeferAccess}(L) = T^m_{Packet}(L) + aSIFSTime + T^m_{Ack} + aDIFSTime \tag{3.2}$$

Where R is retry count, m is the mode of IEEE 802.11 transmission rates and L is packet length. $CW_{\min}$ and $CW_{\max}$ are default values of IEEE 802.11 contention window size, $T^m_{DeferAccess}(L)$ is the time spent in transmitting a packet in defer access interval, $\frac{\min(2^i(CW_{\min}+1)-1, CW_{\max})}{2}$ is the time of backoff interval, p is conditional collision probability.

We assume the network is IEEE 802.11a and we have the following settings: PHY transmission mode is 5 (24Mbps), packet length is 1024byte, retry count is 4, and p is 0.8. Then the successful packet transmission time of $T^5_4(1024)$ is 83msec using Table 3.3: IEEE 802.11b PHY Characteristics. and we assume the network is IEEE 802.11b and we have the following settings: PHY transmission mode is 4 (11Mbps), packet length is 1024byte, retry count is 4, and p is 0.8. Then the successful packet transmission time of $T^4_4(1024)$ is 342.8msec using Table 3.3: IEEE 802.11b PHY Characteristics.

As to retransmission deadline, we have the following calculation. If a video is of 30 FPS then the time interval between video frames is 33.3msec. The retransmission intervals for I-packet, P-packet and B-packet are 233.3msec, 100msec, and 33.3msec, respectively. This means that we can control a successful packet transmission within 233.3msec. This proves that our transmission deadline can save much waiting time.

Characteristics	*Value*	*Comments*
aSlotTime	9 μs	Slot time
aSIFSTime	16 μs	SIFS time
aDIFSTime	34 μs	DIFS time
CW_{min}	15	Minimal contention window size
CW_{max}	1023	Maximal contention window size
MAC header	272 bit	MAC header
PHY header	128 bit	PHY header
ACK size	240 bit	112 bit + PHY header

Table 3.2: IEEE 802.11a PHY Characteristics.

Characteristics	*Value*	*Comments*
aSlotTime	20 μs	Slot time
aSIFSTime	10 μs	SIFS time
aDIFSTime	50 μs	DIFS time
CW_{min}	31	Minimal contention window size
CW_{max}	1023	Maximal contention window size
MAC header	224 bit	MAC header
PHY header	192 bit	PHY header
ACK size	304 bit	112 bit + PHY header

Table 3.3: IEEE 802.11b PHY Characteristics.

mode	*bitrate*	*modulation*
1	6 Mbps	BPSK
2	9 Mbps	BPSK
3	12 Mbps	QPSK
4	18 Mbps	QPSK
5	24 Mbps	16-QAM
6	36 Mbps	16-QAM
7	48 Mbps	64-QAM
8	54 Mbps	64-QAM

Table 3.4: IEEE 802.11a mode.

mode	*bitrate*	*modulation*
1	1 Mbps	BPSK
2	2 Mbps	QPSK
3	5.5 Mbps	CCK
4	11 Mbps	CCK

Table 3.5: IEEE 802.11b mode.