Abstract—The desired properties of a medium access control (MAC) protocol in mobile ad hoc network (MANET) include (1) meet quality of service (QoS) requirements for real-time nodes, (2) be decentralized, (3) achieve fairness from viewpoint of throughput or energy consumption, and (4) be immune to the hidden node problem. Though there have been numerous proposed MAC protocols for the IEEE 802.11 WLAN, few of them possess all of the above four properties. In this paper, we propose a new MAC scheme satisfying all of above four properties. Our protocol can support real-time traffic and satisfy QoS requirements, and can achieve fairness among non-real-time nodes. Also, without using any centralized control, it can be easily deployed in MANET. An analytic model of the protocol’s throughput has also been developed. We compare the protocol’s throughput obtained from its analytic model and simulation to validate each other.

Keywords-IEEE 802.11, WLAN, MAC, QoS, Ad Hoc network

I. INTRODUCTION

IEEE 802.11-based wireless LANs have been increasingly in demand and will soon be prevalent. The desired properties of an IEEE 802.11-based MAC protocol include: (1) meet QoS requirements for real-time nodes, (2) be decentralized, (3) achieve fairness from viewpoint of throughput or energy consumption, and (4) be immune to the hidden node problem. Though there have been several proposed MAC protocols for WLAN, to our knowledge, few protocols satisfy all of the above four properties.

In the literature, several QoS-guaranteed WLAN MAC protocols were proposed [1, 2, 3, 4, 5, 6, 7]. However, most protocols [1, 2, 3] designate one node as a central controller to coordinate the process of real-time data flows of the nodes within the radio broadcast range. The computational complexity of the bandwidth reservation process is very high and the battery power in the central controller may be consumed very soon. Besides, most protocols ignore fairness among the non-real-time nodes while meeting QoS requirements. There are also several protocols proposed with emphasis on high channel throughput and guaranteed fairness (see e.g. [8, 9, 10]). Though they are decentralized and some of them also consider the hidden node problem, they don’t provide QoS guarantees, and neither was designed as an IEEE 802.11-compliant one. Also, the hidden node problem has been one of the main factors causing collision, thereby degrading channel throughput. The standard IEEE 802.11 MAC protocol uses control packets, namely RTS and CTS, to avoid the hidden node problem. However, the MAC protocol to enhance the QoS requirement of 802.11 WLAN is still under development.

In this paper, we propose a new MAC scheme, namely neighborhood information classification and estimation (NICE) protocol, with all of the four desired properties. Based on a combination of the reservation and contention-based schemes, the proposed MAC scheme not only guarantees QoS requirements, but also provides the flexibility on bandwidth allocation between different traffic types. By overhearing the information exchanges, all the nodes within the same radio broadcast range are able to know the status of real-time traffic reservation and the number of neighboring nodes, thereby saving a lot of work in prediction and fully utilizing the channel capacity. The proposed NICE protocol also monitors the length of non-real-time traffic queue to give higher priority when the packet queue length of non-real-time traffic is larger than a predetermined threshold. During the operation of the proposed MAC scheme, our protocol does not rely on any central control mechanism. Thus, it can be employed in an ad hoc environment. The NICE protocol does not require each node to broadcast the beacon, while providing the fairness from the viewpoint of energy consumption.

In Section II, we detail our proposed NICE protocol. An analytic model of the protocol’s throughput is developed in Section III. In Section IV, we simulate the protocol using the ns2 simulator to study its performance in terms of channel throughput and access delay. Conclusions are given in Section V.

II. PROPOSED PROTOCOL

In this section, we detail our proposed NICE protocol. This protocol employs the IEEE 802.11 standard [11] as a subroutine for channel contention, but with a small modification in order to reduce the collision rate. We group data packets into two types, namely non-real time packets (nrt-packets), and real-time flows (rt-packets). For simplicity, a node that sends nrt-packets is called a nrt-node. Also, a node that sends rt-packets is called a rt-node. The MAC header field is the same as the 802.11 standard, except for rt-packets and the corresponding ACKs (rt-ACKs). The modified MAC header field for rt-packets and rt-ACKs are shown in Fig. 1.

The proposed protocol can be divided into four main procedures: observation, contention, reservation procedures and frame synchronization mechanism.
A. Observation Procedure

A node that just turned on its power or just moved in a new radio broadcast range has to observe the channel for a period longer than $R_{\text{Pmax}}$ in order to know the system status of the radio broadcast range, where $R_{\text{Pmax}}$ is defined as the maximum repetition period of a super frame in the PCF mode of the IEEE 802.11 standard. During this observation period, the node has to record what is overheard into a reservation table (RST) or a neighborhood information table (NIT). The RST records the reservation information carried in the MAC header field of rt-packets or rt-ACKs of different flows transmitting in current super-frame. The NIT records the time instance ($D_{\text{TS}}$) and source address of all the nrt-packets appearing in the current super-frame. This table provides the information about the number of active nrt-nodes within the region, which will be used to reduce the collision rate and improve the channel throughput in our design.

B. Contention Procedure

Our contention procedure consists of two phases, which is an extended version of 802.11 and 802.11e [12] standards. The first phase is identical to the DCF mode of 802.11 standard except that in our design nodes adopt different size of contention window (CW) for different types of packet. The nrt-nodes and rt-nodes use the values of $(nrt-CW_{\text{min}}, nrt-CW_{\text{max}})$ and $(rt-CW_{\text{min}}, rt-CW_{\text{max}})$ to control and limit the CW, respectively. In general, the priority of rt-packets is assigned to be higher than that of nrt-packet. Thus, we have the following relations for different CW boundaries:

\[
\begin{align*}
    nrt-CW_{\text{min}} & > rt-CW_{\text{min}} & (1a) \\
    nrt-CW_{\text{max}} & \geq rt-CW_{\text{max}} & (1b)
\end{align*}
\]

When a packet collides before, the corresponding rt-CW for rt-packet or nrt-CW for nrt-packet has the following relations:

\[
\begin{align*}
    rt-CW &= \min(rt-CW_{\text{max}}, rt-CW_{\text{min}} \times (\text{Num\_att}-1)) & \text{(2a)} \\
    nrt-CW &= \min(nrt-CW_{\text{max}}, nrt-CW_{\text{min}} \times 2^{(\text{Num\_att}-1)}) & \text{(2b)}
\end{align*}
\]

where $\text{Num\_att}$ is the number of attempts that the packet has tried. Once the transmission is successful, both rt-CW and nrt-CW are reset to their minimum value.

In the second phase, this packet gating mechanism decides whether the packet should be transmitted right away or deferred. First, the node needs to calculate the probability of transmission ($P_T$), based on the information obtained from the NIT table and gathered during the observation procedure. If the packet is permitted after this mechanism, it will get a green light for transmission. If the packet is deferred, the packet needs to restart the contention procedure again, and the contention procedure takes this packet as a collided one. The purpose of the gating mechanism is to reduce the collision rate based on the status of channel usage.

C. Reservation Procedure

Only nodes with rt-packets have to execute the reservation procedure. The node waits for a PCF inter-frame spacing (PIFS) period until the reserved time stamp ($R_{\text{TS}}$) in RST arrives. The $R_{\text{TS}}$ indicates that the node has to transmit the rt-packet at the $R_{\text{TS}}$th time slot in the current super-frame. The value of $R_{\text{TS}}$ is assigned in the previous rt-packet. Two rt-packets in the same super-frame are spaced with a PIFS period. The other information in RST is $R_{\text{LEN}}$, which is also in the unit of slot time. The value of $R_{\text{LEN}}$ depends on the required bandwidth ($ABR_i$, average bit rate) of flow $i$.

D. Frame Synchronization Mechanism

Frame synchronization is another important issue in our proposed protocol. In the IEEE 802.11 or 802.11e standards, a point coordinator broadcasts a “beacon” signal as the start of a super-frame. However, from the viewpoint of energy consumption, broadcasting beacon signal wastes battery energy, especially for energy-sensitive environment such as MANET. Therefore, we propose a decentralized frame synchronization mechanism (FSM).

The proposed FSM filters the timing of the outgoing of nrt-packets in the nrt-nodes. Figure 2 shows the flow-chart representation of the proposed FSM. An nrt-node has to ensure that it can complete the RTS and CTS transmission before the end of a super-frame. Otherwise, this nrt-node needs to stop transmission in current super-frame and defer it to the next super-frame. Once the RTS and CTS packets are transmitted and received successfully, all the nodes in the region extend their super-frame boundaries to the time tick. After all the nodes extend their boundaries, they will not change the super-frame boundaries any more. Even though the data packet has an error or there is no ACK, all the nodes still retain the extended super-frame boundary.
III. ANALYSIS

In this section, we analyze the throughput of our proposed protocol with mixed real-time and non-real-time traffic. We assume that (1) the channel is ideal, i.e., there is no packet error, (2) a fixed number of nrt-nodes are present in the region of radio broadcast scope, and each active node is always transmitting a packet, and (3) the “on/off” model is adopted for real-time traffic, where the periods of “on” and “off” state are exponentially distributed.

A. Previous case: Non-real-time traffic

Throughput analysis of non-real-time traffic is summarized as follows. First, define the normalized throughput, from \([13]\), expressed by

\[
S = \frac{E[\text{payload transmitted in a slot time}]}{E[\text{length of a slot time}]} = \frac{\sigma}{(1 - P_s)\sigma + \frac{p_{s}}{P_c}T_s + \frac{p_s}{P_c}(1 - P_s)\sigma},
\]

where \(P_s\) is the probability of at least one packet being transmitted in one slot, \(P_c\) is the probability of exactly one packet being transmitted in one slot, \(E[P]\) is the average packet payload size, \(\sigma\) is the duration of an idle slot, \(T_s\) is the average successful transmission time, and \(P_c\) is the average collision time. Assume that there are \(n\) nodes contending for transmission, and each node transmits a packet in a slot with the probability \(\tau\). Then,

\[
P_n = 1 - (1 - \tau)^n
\]

and

\[
P_s = \frac{n \tau (1 - \tau)^{n-1}}{P_c} = \frac{n \tau (1 - \tau)^{n-1}}{1 - (1 - \tau)^{n}}.
\]

Now, to find the probability that a node transmits a packet in a slot, denoted as \(\tau\), we consider a two-dimension discrete-time Markov chain shown in Figure 3. Denote \(b(t)\) and \(s(t)\) be the backoff time counter and the backoff stage at time \(t\), respectively. Each state in the model represents the CW size in different backoff stage. Let the stationary probability of each state

\[
\lim_{t \to \infty} P_s(t) = i, b(t) = k \text{ where } i \in \{0, m\}, k \in \{0, W_s - 1\}
\]

where \(m\) is the maximum backoff stage and \(W_s = 2^d\) is the maximum super-frame size.

B. Mixed real-time and non-real-time traffic

The analytic model presented in section III.A is for non-real-time traffic alone. In this part, we consider a mixed traffic model with both real-time traffic and data traffic. Rt-nodes and nrt-nodes generate real-time traffic and non-real-time traffic, respectively. The maximum number of rt-nodes and nrt-nodes is \(M\) and \(K\). Let \(p(t)\) represent the number of rt-nodes that request to transmit packets at time \(t\). We assume that during a super-frame only one node joins or leaves the network, and real-time traffic and non-real-time traffic are generated independently.

We model the process \(\{p(t)\}\) by a Markov chain as shown in Figure 4. Each state in the figure stands for the number of acting rt-nodes, i.e. which have packets waiting to be transmitted. The state probability can be expressed as

\[
P_i = P[p(t) = i] = \frac{M!}{i!(M-i)!} \rho^i (1-\rho)^{M-i}
\]

where \(\rho\) is the ratio of incoming probability and leaving probability, i.e. \(\rho = \lambda/\mu\). Here, \(\lambda\) and \(\mu\) are the arrival rate and departure rate, respectively.

Denote \(T(M, K)\) as the throughput of \(M\) rt-nodes and \(K\) nrt-nodes. Then, the sum of the throughput of real-time traffic and that of non-real-time traffic is given as

\[
T(M, K) = \sum_{i=0}^{M} \left[ \sum_{j=0}^{K} \frac{M!}{i!(M-i)!} \rho^i (1-\rho)^{M-i} \right] P_s(t) = \sum_{i=0}^{M} \left[ \sum_{j=0}^{K} \frac{M!}{i!(M-i)!} \rho^i (1-\rho)^{M-i} \right] \frac{n \tau (1 - \tau)^{n-1}}{1 - (1 - \tau)^{n}}.
\]
where $L$ is the time span of a super-frame, and $L_0$ is the total time spent for transmitting an rt-packet in a super-frame. Note that

$$\rho_1 = \frac{1}{1 + \rho_0}$$

and

$$\rho_2 = \frac{1}{1 + \rho_0}$$

Note that $\rho_1$ and $\rho_2$ are the ratio of the probability of incoming rt-packets and leaving probability for type 1 real-time traffic, respectively. The total throughput, $T(M, N, K)$, can be rewritten as

$$T(M, N, K) = T_{\text{rt}}(K) + T_{\text{nrt}}(K)\frac{L}{L} + \rho_1 + \frac{L}{L} + \rho_2$$

where all the parameters have been defined in (9).

IV. PERFORMANCE SIMULATION

In this section, we discuss the performance of our protocol in terms of channel throughput and access delay via simulations using the ns-2 simulator. The simulation parameters are listed in Table 1.

### Table 1:

<table>
<thead>
<tr>
<th>Parameter Names</th>
<th>Parameter Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cover Range</td>
<td>10m x 10m</td>
</tr>
<tr>
<td>Data Rate</td>
<td>2 Mbps</td>
</tr>
<tr>
<td>Simulation time</td>
<td>100 sec</td>
</tr>
<tr>
<td>Unit slot time</td>
<td>20 usec</td>
</tr>
<tr>
<td>SIFS/PIFS/DIFS</td>
<td>10/30/50 usec</td>
</tr>
<tr>
<td>Min./Max. CW for nrt-packets</td>
<td>31/1023</td>
</tr>
<tr>
<td>CW for out of queue threshold</td>
<td>15</td>
</tr>
<tr>
<td>Min./Max. CW for rt-packets</td>
<td>7/31</td>
</tr>
<tr>
<td>Queue length threshold</td>
<td>20</td>
</tr>
<tr>
<td>Node expire threshold</td>
<td>100 super-frames</td>
</tr>
</tbody>
</table>

A. Traffic Model

We consider three types of traffic models in the simulation:

- **Voice traffic model**: The voice traffic is modeled as a Poisson process with “on” and “off” states. In the “on” state, the simulator continuously generates 164-byte packets every 20.48 ms, i.e. 64 Kbps. In the “off” state, the simulator stops creating packets. The duration of the “on” state and “off” state follows the exponential distribution. The average duration of the “on” and “off” state are 1 and 1.3 seconds respectively.

- **FTP data traffic model**: The FTP traffic model continuously generates constant sized packets. That is, after one node finishes transmitting a packet, it immediately attempts to transmit another packet.

- **Telnet data traffic**: In the simulation, the generation of a Telnet packet follows Poisson process. For simplicity, the packet size is assumed to be constant for all scenarios, but the arrival rates for different scenarios are different.

B. Performance Measurements

Our protocol is evaluated in terms of throughput, fairness, and packet dropping rate.

- **Normalized throughput**: The normalized throughput is defined as the ratio of the number of successfully received useful data bits to the total number of transmitted bits.

- **Fairness**: In evaluating the fairness, we collect the delay of all the packets in all the connections, and calculate the standard deviation of all the connections. We then illustrate the results of the connection that has the maximum value of standard deviation among all the connections.

- **Dropping rate**: The dropping rate is defined as the ratio of the number of dropped rt-packets to the total number of transmitted rt-packets. If an rt-packet is received longer than 2 $R_{\text{max}}$, i.e. 40.96 msec, the packet is considered to be a dropped packet.
C. Numerical Results

Figure 6 shows the normalized throughput with mixed data and voice traffic. We consider the case when 10 nodes send voice traffic to 10 other corresponding nodes, and the other 40 nodes send FTP data packets. From the figure, one can see that our analysis, (7), and simulation results are very close especially for small packets. Even for a large packet size, the discrepancy between analysis and simulation is still less than 3%.

Figure 7 compares the fairness of the NICE protocol with the IEEE 802.11 standard with mixed Telnet data and voice traffic. Figure 8 shows the performance comparison of NICE and the IEEE 802.11 standard in terms of dropping rate with voice traffic. As shown in the figure, the dropping rate of NICE is almost negligible, while the dropping rate of the IEEE 802.11 standard is higher than 40% at heavy traffic load. For the IEEE 802.11 standard, every voice packet needs to contend for a time slot with other nodes. However, with the help of the reservation procedure in our protocol, each rt-node can reserve time slots for the data flow of voice packets.

V. CONCLUSIONS

We have proposed a new MAC protocol, called Neighborhood Information Classification and Estimation (NICE), for multimedia applications in MANET. Without using a central control, the NICE MAC protocol outperforms the standard IEEE 802.11 DCF mode in terms of throughput, mean access delay, fairness, and the QoS guarantee for the mixed data and voice traffic. In the NICE protocol, we have proposed four new mechanisms: (1) dynamic gating mechanism for throughput enhancement; (2) stall avoidance mechanism for achieving fairness; (3) linear backoff scheme for reducing the dropping rate of real-time traffic; and (4) reservation procedure for QoS guarantee. Via analysis and simulation by ns-2 simulator, we demonstrate that the throughput performance of NICE is at least 50% better than the IEEE 802.11 standard. The mean value and maximum standard deviation of access delay are 8 times less. The dropping rate of voice traffic of NICE is almost negligible, but the dropping rate of the IEEE 802.11 standard is at least 40%.

REFERENCES