

Local Data Control and Admission Control for QoS Support in Wireless *Ad Hoc* Networks

Yang Xiao, *Senior Member, IEEE*, and Haizhon Li, *Student Member, IEEE*

Abstract—Wireless *ad hoc* networks consist of nodes having a self-centrally broadcasting nature of communication. To provide quality of service (QoS) for *ad hoc* networks, many issues are involved, including routing, medium-access control (MAC), resource reservation, mobility management, etc. Carefully designed distributed medium-access techniques must be used for channel resources, so that mechanisms are needed to efficiently recover from inevitable frame collisions. For *ad hoc* wireless networks with a contention-based distributed MAC layer, QoS support and guarantee become extremely challenging. In this paper, we address this challenging issue. We first consider MAC and resource-reservation aspects for QoS support in one-hop *ad hoc* wireless networks. We propose two local data-control schemes and an admission-control scheme for *ad hoc* networks with the IEEE 802.11e MAC standard. In the proposed fully distributed local data control schemes, each node maps the measured traffic-load condition into backoff parameters locally and dynamically. In the proposed distributed admission-control scheme, based on measurements, each node makes decisions on the acceptances/rejections of flows by themselves, without the presence of access points. The proposed mechanisms are evaluated via extensive simulations. Studies show that, with the proposed schemes, QoS can be guaranteed under a clear channel condition while maintaining a good utilization. Discussions on applying the proposed schemes into multihop *ad hoc* networks are also included.

Index Terms—Ad hoc networks, admission control, data control, IEEE 802.11, medium-access control (MAC), quality of service (QoS).

I. INTRODUCTION

QUALITY OF service (QoS) are particularly challenging for *ad hoc* wireless networks due to many reasons and much remains to be done [1]. *Ad hoc* wireless networks consist of a collection of mobile stations without a fixed infrastructure. In *ad hoc* wireless networks, peer-to-peer nodes conduct the initialization, organization, and administration of networks. Many challenges must be overcome to obtain the practical benefits of *ad hoc* networks, including routing, medium-access control (MAC), mobility management, power management, security, and QoS issues [1]. The nodes of an *ad hoc* network communicate directly with another in a peer-to-peer fashion and each node must function as a router. Power capacity and transmission range are further limited by the mobility of nodes. Due to the mobility, the network topology is dynamically

changed. Furthermore, the limited bandwidth of wireless channels and hostile transmission characteristics impose additional constraints. For *ad hoc* networks with a contention-based MAC layer, the nature of contentions further imposes challenges for QoS support.

There have been many reports on QoS efforts in *ad hoc* networks [1]–[8]. However, most research [1]–[8] has focused on designing QoS routing protocols. To provide QoS, many issues are involved, including routing, MAC, resource reservation, mobility management, etc. Carefully designed distributed medium-access techniques must be used for channel resources so that mechanisms are needed to recover efficiently from inevitable frame collisions [1]. For *ad hoc* wireless networks with a contention-based distributed MAC, QoS support and guarantee become extremely challenging. Without the MAC layer's support, the QoS guarantee solely in higher layers is not possible. In this paper, we address this challenging issue and focus on QoS support from MAC mechanisms for *ad hoc* networks with a contention-based MAC. Specifically, we propose two local data-control mechanisms and a fully distributed admission-control scheme for *ad hoc* networks. The distributed local data-control schemes and admission-control scheme are implemented in the MAC layer of each node. How the proposed schemes are applied to multihop *ad hoc* networks is also included. Note that the term of data means best-effort data to differentiate with voice and video in this paper.

We are particularly interested in *ad hoc* networks with the underneath IEEE 802.11 distributed MAC since it is available. The IEEE 802.11 MAC employs mandatory contention-based channel-access function called distributed coordination function (DCF) and an optional centrally controlled channel-access function called point coordination function (PCF) [9]. The DCF adopts a carrier sense multiple access with collision avoidance (CSMA/CA) with binary exponential backoff. Functions of the DCF and the PCF determine when a station/node, operating within a basic service set (BSS) or independent BSS (IBSS), is permitted to transmit. There are two types of 802.11 networks: infrastructure network (BSS), in which an access point (AP) is present, and *ad hoc* network (IBSS), in which an AP is not present. In this paper, we are particularly interested in *ad hoc* networks formed by multiple IBSSs, in which no AP is present.

The IEEE 802.11 DCF enables fast installation with minimal management and maintenance costs and is a very robust protocols for the best-effort service. However, the current DCF is unsuitable for multimedia applications with QoS requirements. Under the DCF, a station might have to wait an arbitrarily long time to send a frame, so real-time applications such as voice and video may suffer [10]. One possible solution is to provide a good

Manuscript received February 7, 2004; revised April 12, 2004 and May 28, 2004.

The authors are with the Computer Science Department, The University of Memphis, Memphis, TN 38152 USA (e-mail: yangxiao@ieee.org; hli1@memphis.edu).

Digital Object Identifier 10.1109/TVT.2004.833627

priority scheme for the DCF. Simple DCF priority schemes can be easily designed with minor changes in the DCF and are quite effective.

To support the MAC-level QoS, the IEEE 802.11 Working Group is currently working on the standardization of IEEE 802.11e [11], which is in the final stage. The emerging IEEE 802.11e standard provides QoS features and multimedia support to the existing 802.11b [12] and 802.11a [13] wireless local area network (WLAN) standards, while maintaining fully backward compatibility with these standards. The IEEE 802.11e MAC employs a channel-access function, called hybrid coordination function (HCF), which includes a contention-based channel access and a contention-free centrally controlled channel-access mechanism. The contention-based channel is also referred to as enhanced distributed channel access (EDCA),¹ which provides a priority scheme by differentiating the interframe space and the initial and maximum window sizes for backoff procedures. Therefore, voice, video, and data traffic are differentiated with different QoS parameters, i.e., different interframe spaces, different initial window sizes, and different maximum window sizes. A higher priority traffic class has smaller QoS parameter values. However, without a good admission-control scheme, QoS requirements cannot be guaranteed. Furthermore, both the original IEEE 802.11 DCF and the emerging IEEE 802.11e EDCA have a focus on infrastructure networks instead of *ad hoc* networks.

In this paper, we consider the EDCA for *ad hoc* networks. We propose a fully distributed and measurement-based admission-control scheme for differentiation services of the EDCA in *ad hoc* networks. Without the presence of APs, admission control is especially challenging due to the contention-based nature of the MAC layer. For voice and video traffic classes, in our proposed scheme, each node conducts history measurements at beacon intervals and available budgets are calculated/estimated. When one priority class's budget is near zero, new traffic streams (or flows) belonging to this class are denied by the node (itself) and existing streams also are not allowed to increase the transmission time that they are already using. For best-effort data-traffic class, local data-control mechanisms are proposed. Each node conducts history measurements at beacon intervals and dynamically adjusted its backoff parameters (the interframe space, initial window size, and maximum window size) based on traffic-condition indicators such as collision ratios, etc. The data-control mechanism is needed since too many data transmissions degrade the system performance, including the existing voice and video streams. On the other hand, if there are very few data transmissions, the throughput might decrease even though the voice and video become less vulnerable. In this paper, we propose two distributed local data-control schemes: *direct function mapping (DFM)* and *derivative tendency (DT)*. In the DFM scheme, a mapping function is defined for mapping traffic-load indications to parameters. In the DT scheme, parameters are changed dynamically according to the same tendency of traffic-load indications. With the proposed schemes, QoS performance can be greatly improved while maintaining a good utilization. Finally, we discuss how to apply the proposed

schemes to multihop *ad hoc* networks in Section VI in terms of MAC issues, QoS routing, and robustness.

The rest of this paper is organized as follows. We briefly introduce the IEEE 802.11 DCF and the EDCA in Section II. The local data-control and admission-control schemes for *ad hoc* networks are presented in Sections III and IV, respectively. Performance studies are carried out in Section V with extensive simulation results. Discussions on applying the proposed schemes into multihop *ad hoc* networks are included in Section VI and we conclude this paper in Section VII.

II. IEEE 802.11 DCF AND THE EDCA

A. IEEE 802.11 DCF

The IEEE 802.11 MAC employs a mandatory DCF and an optional PCF. In the long run, time is divided into repetition intervals called *superframes*. Each superframe starts with a beacon frame and the remaining time is further divided into a contention-free period (CFP) and a contention period (CP). The DCF works during the CP and the PCF works during the CFP. If the PCF is not active, a superframe will not include the CFP. However, a beacon frame always is periodically transmitted, regardless of whether the PCF is active or not. The beacon frame is a management frame for synchronization, power management, and delivering parameters. Beacon frames are generated in regular intervals called target beacon transmission time.

The DCF defines a basic access mechanism and an optional request-to-send/clear-to-send (RTS/CTS) mechanism. Under the basic access mechanism, a station with a frame to transmit monitors the channel activities until an idle period equal to a distributed interframe space (DIFS) is detected. After sensing an idle DIFS, the station waits for a random backoff interval before transmitting. The backoff time counter is decremented in terms of slot time as long as the channel is sensed to be idle. The counter is suspended when a transmission is detected on the channel and is resumed with the remaining backoff counter when the channel is sensed to be idle again for more than a DIFS interval. The station transmits its frame when the backoff timer reaches zero. For each new transmission attempt, the backoff counter is uniformly chosen from the range $[0, CW - 1]$ in terms of timeslots, where CW is the current backoff window size. At the very first transmission attempt, CW equals the initial backoff window size CW_{min} . After each unsuccessful transmission, CW is doubled until a maximum backoff window size value CW_{max} is reached or a retry limit is reached. After the destination station successfully receives the frame, it transmits an acknowledgment frame (ACK) following a short interframe space (SIFS) time. If the transmitter station does not receive an ACK within a specified ACK timeout or if it detects the transmission of a different frame on the channel, it reschedules the frame transmission according to the previous backoff rules.

The above mechanism is called the basic access mechanism. In such a mechanism, a *hidden node problem* may happen: transmissions of a station cannot be detected by a second station using carrier sense, but interfere with transmissions from the second station to a third station. To reduce the hidden station problem, an optional four-way data-transmission mechanism

¹EDCA used to be called enhanced distributed coordination function (EDCF).

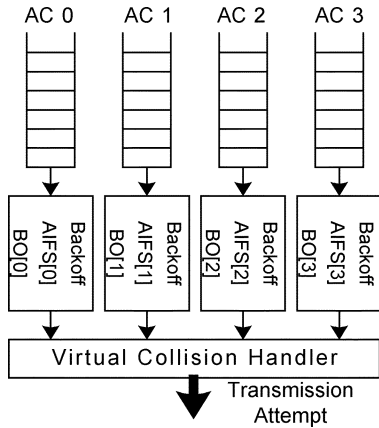


Fig. 1. Virtual transmission queues, where $BO[i]$ stands for the backoff counter for AC i .

called RTS/CTS is also defined in the DCF. In the RTS/CTS mechanism, before transmitting a frame, a short RTS frame is transmitted. The RTS frame also follows the backoff rules introduced above. If the RTS frame succeeds, the receiver station responds with a short CTS frame. Then, a frame and an ACK frame will follow. All four frames (RTS, CTS, data, and ACK) are separated by an SIFS time. In other words, the short RTS and CTS frames reserve the channel for that frame transmission, which follows.

Since a radio transmission has a range, the range (denoted as RA) of the source's RTS transmission and the range (denoted as RB) of the destination's CTS transmission are overlapped but not equal. Therefore, after the transmissions of the source's RTS and the destination's CTS, the channel is reserved for the data transmission followed and any station in either RA or RB will not transmit. A hidden station of the source, which is in RB but not in RA, will not interfere with the source data transmission since it hears the destination's CTS transmission.

In this paper, we consider the RTS/CTS mechanism in *ad hoc* networks when no AP is present.

B. IEEE 802.11e EDCA

We only consider the EDCA, but not the centrally controlled HCF since, in *ad hoc* networks, the centrally controlled channel access is difficult to use.

A new concept, transmission opportunity (TXOP), is introduced in IEEE 802.11e. A TXOP is a time period when a station has the right to initiate transmissions onto the wireless medium and is defined by a starting time and a maximum duration. A station cannot transmit a frame that extends beyond a TXOP. If a frame is too large to be transmitted in a TXOP, it should be fragmented into smaller frames.

The EDCA works with four access categories (ACs), which are virtual DCFs, as shown in Fig. 1, where each AC achieves a differentiated channel access. This differentiation is achieved through varying the amount of time that a station would sense the channel to be idle and the length of the contention window during a backoff. The EDCA supports eight different priorities, which are further mapped into four ACs, shown in Table I. ACs are achieved by differentiating the arbitration interframe space (AIFS), the initial window size, and the maximum window size.

TABLE I
PRIORITY TO ACCESS CATEGORY MAPPING

PRIORITY	AC	DESIGNATION
1	0	BEST EFFORT
2	0	BEST EFFORT
0	0	BEST EFFORT
3	1	VIDEO PROBE
4	2	VIDEO
5	2	VIDEO
6	3	VOICE
7	3	VOICE

For the AC i ($i = 0, \dots, 3$), the initial backoff window size is $CW_{\min}[i]$, the maximum backoff window size is $CW_{\max}[i]$, and the arbitration interframe space is $AIFS[i]$. For $0 \leq i < j \leq 3$, we have $CW_{\min}[i] \geq CW_{\min}[j]$, $CW_{\max}[i] \geq CW_{\max}[j]$, and $AIFS[i] \geq AIFS[j]$, and at least one of above inequalities must be strict. In other words, the EDCA employs $AIFS[i]$, $CW_{\min}[i]$, and $CW_{\max}[i]$ (all for $i = 0, \dots, 3$) instead of DIFS, CW_{\min} , and CW_{\max} , respectively. If one AC has a smaller AIFS or CW_{\min} or CW_{\max} , the AC's traffic has a better chance to access the wireless medium earlier.

Fig. 2 shows the EDCA timing diagram, where three ACs are shown: i , j , and k . Fig. 1 shows four transmission queues implemented in a station and each queue supports one AC, behaving roughly as a single DCF entity in the original IEEE 802.11 MAC. It is assumed that a payload from a higher layer is labeled with a priority value and it is enqueued into the corresponding queue according to the mapping in Table I. Each queue acts as an independent MAC entity and performs the same DCF function, with a different interframe space ($AIFS[i]$), a different initial window size ($CW_{\min}[i]$), and a different maximum window size ($CW_{\max}[i]$). Each queue has its own backoff counter ($BO[i]$), which acts independently in the same way as the original DCF backoff counter. If there is more than one queue finishing the backoff at the same time, the highest AC frame is chosen to transmit by the virtual collision handler. Other lower AC frames whose backoff counters also reach zero will increase their backoff counters with $CW_{\min}[i]$ ($i = 0, \dots, 3$), accordingly. Furthermore, we have $AIFS[i] \geq PIFS$, where PIFS is point (coordination function) interframe space. The values of $AIFS[i]$ ($i = 0, \dots, 3$), $CW_{\min}[i]$ ($i = 0, \dots, 3$), and $CW_{\max}[i]$ ($i = 0, \dots, 3$) are referred to as the EDCA parameters.

III. LOCAL DATA CONTROL

In this section, we propose a measurement-based local data-control mechanisms for best-effort data transmissions, i.e., AC ($i = 0$). The proposed schemes are fully distributed data-control mechanisms, since stations dynamically control parameters themselves locally based on traffic load condition. Stations make decisions based on the local observed measurements.

Each station measures network traffic condition and estimates appropriate parameters dynamically, i.e., $AIFS[0]$, $CW_{\min}[0]$, and $CW_{\max}[0]$. The reasons for this local data control are as follows. First, too many data transmissions degrade the system performance, including the existing voice and video streams,

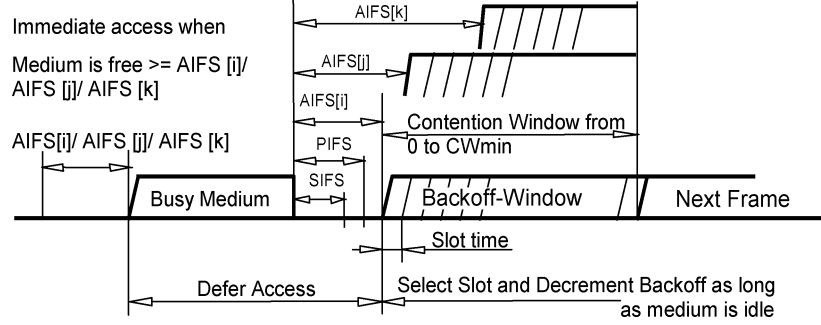


Fig. 2. EDCA timing diagram.

since many data transmissions cause a lot of collisions. Therefore, the existing voice and video streams become vulnerable to data transmissions. In such a case, a station should notice that the network condition becomes worse by indications such as an increased collision ratio, etc. Therefore, the station should increase its parameters, i.e., $AIFS[0]$, $CW_{min}[0]$, and $CW_{max}[0]$. On the other hand, if there are very few data transmissions, the throughput might decrease even though the voice and video become less vulnerable. In such a case, each station should notice that the network condition becomes better by indications such as a decreased collision ratio, etc. Therefore, the station should decrease its parameters, i.e., $AIFS[0]$, $CW_{min}[0]$, and $CW_{max}[0]$. The challenging issue is how to adaptively adjust these parameters based on those indications. Second, the reason for handling real-time transmissions and data transmissions differently is that real-time transmissions need guaranteed QoS, whereas data transmissions do not.

For data transmissions, where ($i = 0$), each station dynamically changes the three variables/parameters ($CW_{min}[0]$, $CW_{max}[0]$, and $AIFS[0]$) with time, based on the measurements of traffic load condition. We propose two schemes for the local data control: DFM and DT. In the DFM scheme, a mapping function is defined for mapping traffic load indications to parameters, i.e., $CW_{min}[0]$, $CW_{max}[0]$, and $AIFS[0]$. The challenging issue is how to obtain two or more pairs of accurate mapping values, with which a function can be built by either a linear function or a more complex function. In the DT scheme, parameters are changed dynamically according to the same tendency of traffic load indications, i.e., the increased/decreased tendency.

Assume that we have n traffic load indication parameters: S_1, S_2, \dots, S_n . They may or may not be directly related to data transmissions ($i = 0$). For the DFM scheme, we have

$$CW_{min}[0](t+1) = F_1(S_1(t), S_2(t), \dots, S_n(t)) \quad (1)$$

$$CW_{max}[0](t+1) = F_2(S_1(t), S_2(t), \dots, S_n(t)) \quad (2)$$

$$AIFS[0](t+1) = F_3(S_1(t), S_2(t), \dots, S_n(t)). \quad (3)$$

$F_1(\cdot)$, $F_2(\cdot)$, and $F_3(\cdot)$ are three direct mapping functions and t stands for the t th beacon interval. For the DT scheme, we have

$$\Delta CW_{min}[0](t+1) = G_1(\Delta S_1(t), \Delta S_2(t), \dots, \Delta S_n(t)) \quad (4)$$

$$\Delta CW_{max}[0](t+1) = G_2(\Delta S_1(t), \Delta S_2(t), \dots, \Delta S_n(t)) \quad (5)$$

$$\Delta AIFS[0](t+1) = G_3(\Delta S_1(t), \Delta S_2(t), \dots, \Delta S_n(t)). \quad (6)$$

$G_1(\cdot)$, $G_2(\cdot)$, and $G_3(\cdot)$ are three easier mapping functions and Δ is the difference function, e.g., $\Delta S(t) = S(t) - S(t-1)$.

Equations (1)–(3) and (4)–(6) provide very general definitions for the DFM scheme and the DT scheme. Let us make an example as follows. Define a traffic load indication, the transmission collision ratio (CR) for data frames transmitted by a station. Let $N_FTA(t)$ denote the number of failed data transmission attempts/retries from the station during the t th beacon interval. Let $SFC(t)$ denote the number of successful data transmissions from the station during the t th beacon interval. We also adopt the following Leaky-Bucket integration technique to take history measurements into consideration. Let $N_FTA_{measurement}$ and $SFC_{measurement}$ denote the current measurements of $N_FTA(t)$ and $SFC(t)$, respectively. Note that, at the first measurement, the history values are equal to the corresponding measurements. Later, we define

$$N_FTA(t) = \alpha \times N_FTA(t-1) + (1-\alpha) \times N_FTA_{measurement} \quad (7)$$

$$SFC(t) = \alpha \times SFC(t-1) + (1-\alpha) \times SFC_{measurement}. \quad (8)$$

α is a smoothing/aging factor. Then, we have

$$CR(t) = \frac{N_FTA(t)}{N_FTA(t) + SFC(t)}. \quad (9)$$

$CR(t)$ is the transmission collision ratio during the t th beacon interval. In this example, CR is the only indication defined and we can define a linear function for the DFM scheme

$$CW_{min}[0](t+1) = CW_{Low} + \frac{CW_{High} - CW_{Low}}{CR_{High} - CR_{Low}} \times (CR(t) - CR_{Low}) \quad (10)$$

$$CW_{max}[0](t+1) = CW_{High} \quad (11)$$

$$AIFS[0](t+1) = AIFS[0]. \quad (12)$$

In (10), the initial window size for data transmissions is dynamically adjusted by a linear function, whereas the maximum window size and the AIFS are constant. CW_{Low} and CW_{High} are the lower bound and the upper bound of the initial window size, respectively, and we have $\min(CW_{Low}, CW_{High}) = CW_{Low} \geq CW_{min}[1]$, where $CW_{min}[1]$ is the minimum window size for the AC ($i = 1$). CR_{Low} is the highest value of CR so that the QoS of voice and video is still pretty good when $CW_{min}[0] = CW_{Low}$. We want

to maintain $CR \leq CR_{Low}$. In the case that CR is temporarily larger than CR_{Low} , $CW_{min}[0]$ will be increased by (10) so that CR is changed back to CR_{Low} or smaller. CR_{High} is the upper bound of the transmission collision ratio. CW_{Low} is the initial window size that CR_{Low} can be obtained. CW_{High} is the initial window size of data transmissions so that CR will change from CR_{High} back to CR_{Low} or a little smaller. In other words, the assumption under this example for the DFM scheme is that two pairs of values (CR_{Low}, CW_{Low}) and (CR_{High}, CW_{High}) are known or can be obtained by means such as simulation measurements.

To loose this assumption, we can define the following DT scheme to extend this example. We observe that if we make a derivative of (10), we obtain

$$\frac{d[CW_{min}[0](t+1)]}{dt} = \psi \frac{d[CR(t)]}{dt}$$

where

$$\psi = \frac{CW_{High} - CW_{Low}}{CR_{High} - CR_{Low}}. \quad (13)$$

However, (13) is difficult to use. To become practical, we make a difference of (10) instead of making derivative. We have

$$\Delta[CW_{min}[0](t+1)] = \psi \{\Delta[CR(t)]\}. \quad (14)$$

If the upper bound of the initial window size is considered, we rewrite (14) as (15), where $\theta = \psi/CW_{max}[0]$. Equations (15)–(17) form the example of the DT scheme, shown at the bottom of page.

We observe that in (15), we have only one parameter θ that needs to be defined. Compared with (10), (15) is much easier to control and less complex and avoids obtaining two pairs of values (CR_{Low}, CW_{Low}) and (CR_{High}, CW_{High}). The value θ can be either predefined or dynamically changed based on some other measurements at the run time. The predefined value θ is embedded in the beacon frames.

In summary, (10)–(12) is an example of the DFM scheme and (15)–(17) is an example of the DT scheme. Both examples are simple and will be evaluated in the next section. More complex examples of the DFM and DT schemes can be designed easily. However, the two examples given here are good enough for our local data-control purpose according to our simulation results in Section V.

IV. ADMISSION CONTROL FOR *AD HOC* NETWORKS

In this section, we propose a measurement-based admission control for real-time transmissions, i.e., ACs ($i > 0$). It is a fully distributed admission control, since individual stations make decisions on acceptance or rejection for a newly arrived voice/video stream based on the local observed measurements.

Each station makes measurements during beacon intervals and calculates/estimates *transmission budget*. The transmission budget indicates the allowable transmission time per AC in addition to what is being utilized. Each station determines an internal transmission limit per AC for each beacon interval, based on the transmission count during the previous beacon period and transmission budget calculated. The local voice/video transmission time per beacon interval will not exceed the internal transmission limit per AC. When the transmission budget for an AC is nearly depleted, new streams are not allowed be able to gain transmission time, while existing streams are not allowed to increase the transmission time per beacon interval, which they are already using. This mechanism protects existing voice and video streams.

The QoS parameter set element (QPSE) provides information needed by stations for a proper operation of the QoS facility during the contention period. The QPSE includes $CW_{min}[i]$, $CW_{max}[i]$, AIFS[i] for ($i = 0, 1, 2, 3$), and SurplusFactor[i] for ($i = 1, 2, 3$). SurplusFactor[i] (>1) represents the ratio of over-the-air bandwidth reserved for AC i to the bandwidth of the transported frames required for successful transmission. The QPSE is embedded in beacon frames. Since no AP is present, beacon frames are sent by one station, either the one who starts up the network or the chosen one based on a distributed selection algorithm later on [9]. In this paper, for ACs ($i > 0$), the QPSE provides fixed QoS parameter value. In other words, for voice and video transmissions ($i = 1, 2, 3$), the first three variables/parameters are constants. For AC ($i = 0$), the QPSE provides initial QoS variables/parameter values and later on these values are adjusted by the local data-control mechanisms introduced in the section above.

Each station needs to calculate TXOPBudget[i] ($i > 0$) during each beacon interval. TXOPBudget[i] ($i > 0$) specifies the additional amount of time available for AC i during the next beacon interval. Each station will measure the amount of time occupied by transmissions from each AC, denoted as TxTime[i], during the beacon period, including associated SIFS and ACK times if applicable. Each station will maintain a set of counters TxTime[i], which will be set to zero immediately following the transmission/reception of a beacon frame. For each frame transmission (either uplink or downlink), each station will add the time, equal to the frame transmission time and

$$CW_{min}[0](t+1) = \begin{cases} CW_{min}[0](t), & \text{if } CR(t) < \delta \text{ (a predefined value)} \\ \min \{ \theta \times CW_{max}[0] \times [CR(t) - CR(t-1)] \\ + CW_{min}[0](t), CW_{max}[0] \}, & \text{otherwise} \end{cases} \quad (15)$$

$$CW_{max}[0](t+1) = CW_{max}[0] \quad (16)$$

$$AIFS[0](t+1) = AIFS[0] \quad (17)$$

all overhead involved such as SIFS and ACK, to the TxTime counter corresponding to the AC of that frame. Each station determines TXOPBudget[i] by

$$\begin{aligned} & \text{TXOPBudget}[i] \\ & = \max(\text{ATL}[i] - \text{TxTime}[i] \times \text{SurplusFactor}[i], 0) \quad (18) \end{aligned}$$

where ATL[i] is for the maximum amount of time that may be used for transmissions of AC i per beacon interval.

For voice and video communications at stations, when the transmission budget for an AC is nearly depleted, new flows cannot gain transmission time, while existing flows cannot increase the transmission time per beacon interval, which they are already utilizing. Accordingly, this mechanism protects existing flows.

Each station has to maintain the following local variables for each AC: TxUsed[i], TxCounter[i], TxLimit[i], TxRemainder[i], and TxMemory[i]. These are local variables in the sense that each station locally updates these variables by counting only those related to itself. TxUsed[i] counts the amount of time occupied on-air by transmissions, irrespective of success or not, from AC i of this station, including associated SIFS and ACK times if applicable. TxCounter[i] counts for the transmission time for successful transmissions. A station will not transmit a frame if doing so would result in the value in TxUsed[i] exceeding the value in TxLimit[i], where how to determine this value is presented later. If the station is prevented from sending a frame for this reason, it may carry over the partial frame time remainder to the next beacon interval, by storing the remainder in TxRemainder[i], where TxRemainder[i] = TxLimit[i] - TxUsed[i]. Otherwise, TxRemainder[i] = 0. TxMemory[i] “memorizes” the amount of resource that AC i of this station utilized during a beacon interval. Let f denote the damping factor whose function will be explained in the later part. At each target beacon transmission time, the TxMemory, TxLimit, and TxCounter variables are updated according to the following procedure.

- If TXOPBudget[i] $\leq \varepsilon$, where ε is a small value
 - Both TxMemory[i] and TxRemainder[i] will be set to zero for new stations that start transmission with this AC in the next beacon interval. All other stations’ TxMemory[i] remains unchanged;
- Otherwise, if the TXOPBudget[i] $> \varepsilon$
 - For new stations, which start transmission with this AC in the next beacon interval, an initial value for TxMemory[i] could be between 0 and TXOPBudget[i]/SurplusFactor[i]. All other stations’ TxMemory[i] are updated according to the following procedure:
 - TxMemory[i] = $f \times \text{TxMemory}[i] + (1 - f) \times (\text{TxCounter}[i] \times \text{SurplusFactor}[i] + \text{TXOPBudget}[i])$;
 - TxCounter[i] = 0;
 - TxLimit[i] = TxMemory[i] + TxRemainder[i]

From the above procedure, when the transmission budget for an AC becomes near zero

- its TxLimit[i] will becomes near zero and, hence, AC i of any new station will not be able to gain a transmission time in the next beacon interval;
- existing stations’ TxMemory[i] remains unchanged and, hence, the existing stations’ TxLimit[i] remains unchanged. In other words, existing stations will not be able to increase the transmission time above what they are already using. Note that this mechanism protects existing flows.

From the above procedure, as long as when the transmission budget is larger than ε , both TxMemory[i] and TxLimit[i] need be adjusted periodically. The new TxMemory[i] value is a weighted average of the old TxMemory[i] value and the sum of the successful transmission time and the budget. The value TxCounter[i] \times SurplusFactor[i] + TXOPBudget is the target to which TxMemory converges. The TxLimit is equal to TxMemory plus a possible capped remainder, where TxMemory “memorizes” the amount of time that a specific AC of the station has been able to utilize per beacon interval. Once the budget is nearly depleted (i.e., TXOPBudget hovers around 0), TxMemory converges to TxCounter, which is the lower limit. This ensures that a station can continue consuming the same amount of time in subsequent beacon intervals. The damping allows for some amount of fluctuation to occur. However, TxMemory cannot grow any further in the saturated state. This prevents new flows from entering a specific AC when it is saturated.

The damping factor does not affect the entrance of a new flow into the system when an enough budget is available, because the decreased TXOPBudget is offset by an increased TxCounter instantaneously, so TxMemory does not change a lot. The damping factor does affect TxMemory when a new stream starts up in another station. In that case, the decreased TXOPBudget is not offset by an increased TxCounter and the TxMemory consequently converges to the lower target value. Stations will not increase their TxLimit[i] if they did not transmit traffic of AC i during the previous beacon interval.

For each video/voice stream, a Leaky-Bucket algorithm and a Token-Bucket algorithm can be also implemented at the stations to control the flow rate.

V. PERFORMANCE EVALUATION

In this section, we study the performance of the proposed mechanisms, i.e., the fully distributed admission control and local data control for one-hop *ad hoc* wireless networks. In the simulations, all the stations are within the transmission range of each other with a clear channel. The simulation programs of IEEE 802.11a and the IEEE 802.11e EDCA are implemented using Java with discrete event simulation.

In our simulations, we have two classes: video (AC 1) and data (AC 0). We have the following parameters unless stated otherwise: AIFS[1] = 25 μ s; AIFS[0] = 34 μ s; CW_{min}[1] = 16; CW_{min}[0] = 32; CW_{max}[1] = 1024; CW_{max}[0] = 8192; beacon interval is 100 ms; damping factor is 0.9; SurplusFactor is 1.1; the initial value of TxMemory[1] is 0.8 \times TXOPBudget[1]/SurplusFactor[1]. Each video flow is 1.46 Mb/s, which is generated by a constant interarrival

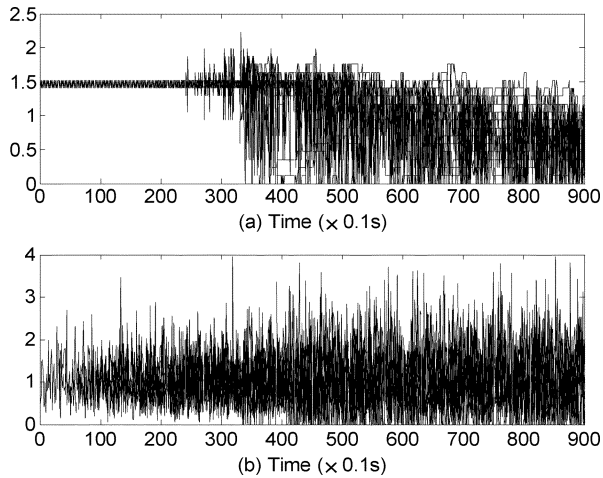


Fig. 3. Throughput (without admission control) in megabits per second: (a) per video flow and (b) per data station.

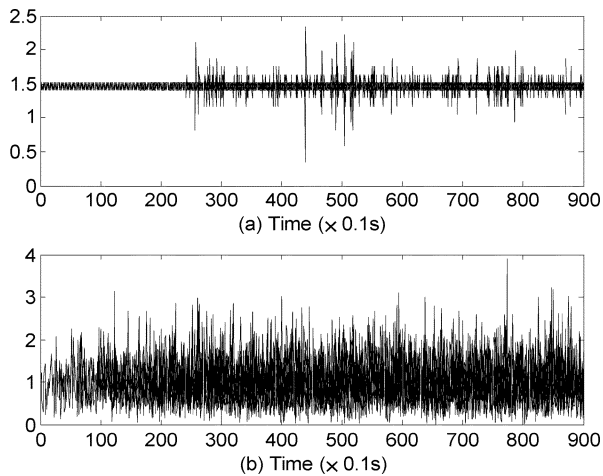


Fig. 4. Throughput (with admission control) in megabits per second: (a) per video flow and (b) per data station.

time 8 ms with a mean payload size 1464 B. Each data station generates data frames with a mean interarrival time 12 ms and a fixed payload size 1500 B. We adopt IEEE 802.11a, and the data rate is 54 Mb/s and the control rate are 24 Mb/s. Simulation time is 90 s. The RTS/CTS mechanism is used.

We conduct extensive simulations to study different parameters such as throughput, throughput per flow per station, delay per flow per station, $TxLimit[i]$, $TXOPBudget[i]$, total throughput, total throughput per AC, collision ratio per station, $CW_{min}[0]$ per station, fairness factors for local data control, and effects of traffic load on admission control, as well as effects of θ on the DT scheme.

A. With and Without Admission Control

We have three simulations in this section: low, medium, and high video traffic load simulation. In all three simulations, data traffic load is medium. The next section presents simulations for high data traffic load. The simulation scenario in this section is stated as follows. Initially, there is a video stream and one data station in the system.

- For every 3 s, a video stream arrives to the system until there are a total of 5, 11, and 30 video stream arrivals for

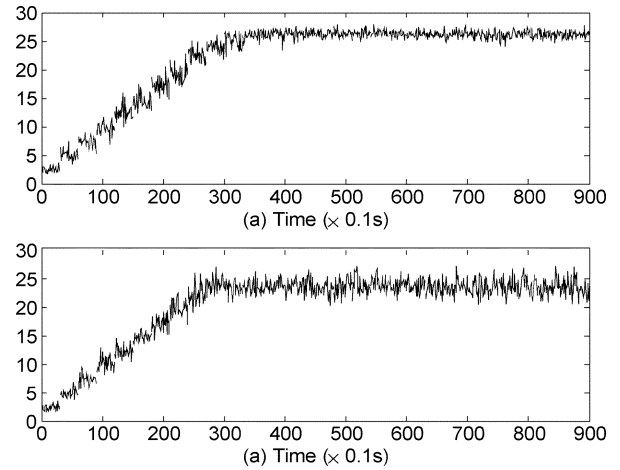


Fig. 5. Total throughput in megabits per second: (a) without admission control and (b) with admission control.

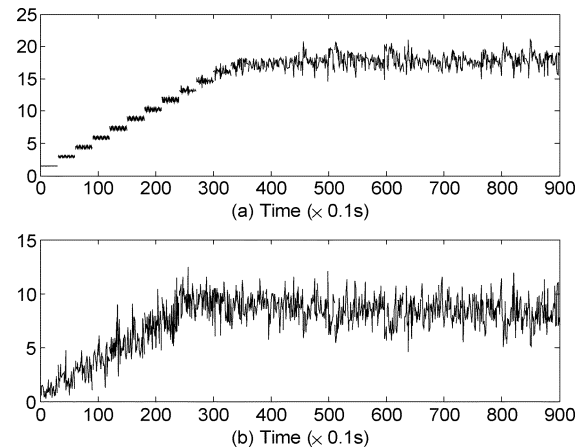


Fig. 6. Total throughput in megabits per second per AC (without admission control): (a) video and (b) data.

the low, medium, and high video traffic load simulation, respectively. Without admission control, all video streams will join the system. With admission control, some video streams may be rejected.

- For every 3 s, one data station arrives to the system until there are total nine data stations. All data stations will join the system.

In this section, distributed local data control is not implemented, but will be discussed in the next section. We will compare performance with and without admission control. Figs. 3–9 are for the high video traffic load simulation.

Fig. 3(a) and (b) shows throughput per video flow and throughput per data station, respectively, without admission control. Fig. 4(a) and (b) shows throughput per video flow and throughput per data station, respectively, with admission control. Without admission control, Fig. 3(a) shows that when video traffic load is large enough, throughputs of all flows are degraded far below 1.46 Mb/s and they fluctuate in a large range. With admission control, Fig. 4(a) shows that throughputs of video flows are improved and guaranteed at about 1.46 Mb/s, since some later video flows are rejected by admission control. Some minor fluctuations in Fig. 4(a) are caused by data traffic

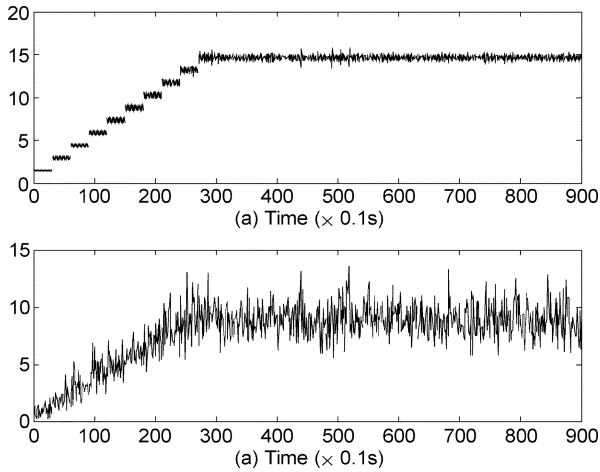


Fig. 7. Total throughput in megabits per second per AC (with admission control): (a) video and (b) data.

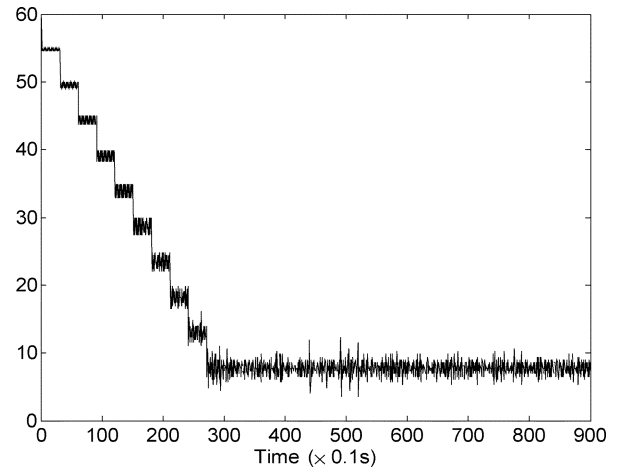


Fig. 9. TXOPBudget in milliseconds for video (with admission control).

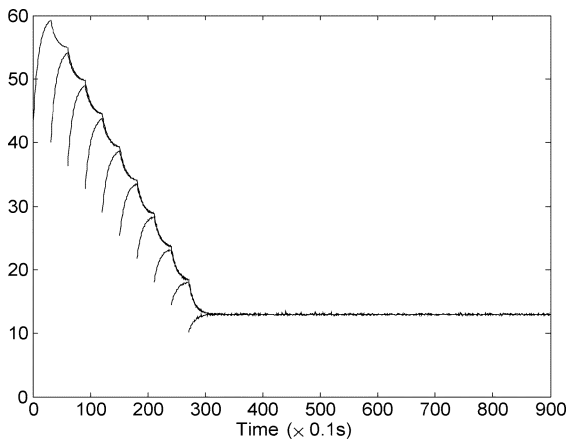


Fig. 8. TxLimit in milliseconds (with admission control).

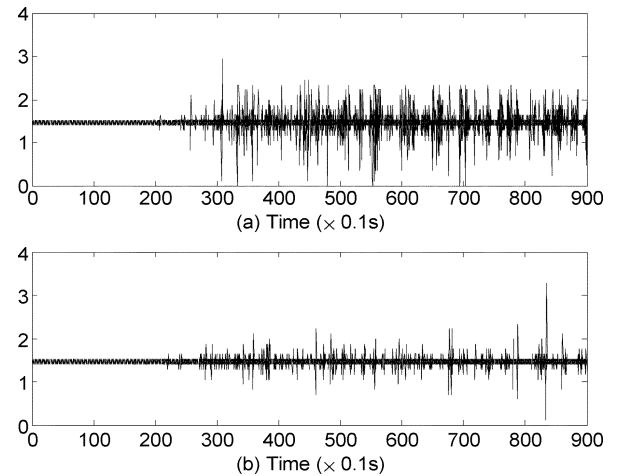


Fig. 10. Video throughput in megabits per second per flow, the medium video traffic load simulation, (a) without admission control and (b) with admission control.

and we will see in the next subsection that, with local data control, performance can be further improved. Both Figs. 3(b) and 4(b) show that data throughput per station does not have much difference.

Fig. 5(a) and (b) shows total throughputs with and without admission control, respectively. As illustrated in these figures, total throughput with admission control [Fig. 5(b)] is a little lower than total throughput without admission control [Fig. 5(a)]. However, the difference is minor. Comparing Figs. 3-5, we observe that our goal of admission control has been achieved, i.e., guaranteeing QoS with minor degraded total throughput. In other words, an approach that does not push too hard is normally better.

Fig. 6 shows total throughput per AC without admission control and Fig. 7 shows total throughput per AC with admission control. Both Figs. 6(a) and 7(a) show some throughput jumps since a new video flow joins in each 3 s. Fig. 6(a) shows that total throughput almost cannot increase after 35 s of simulation. However, the video traffic flows keep coming. Therefore, more flows compete limited resource and throughput per flow [Fig. 3(a)] is severely degraded. On the other hand, with admission control, Fig. 7(a) shows that, after 30 s, there are no jumps. In other words, all video flows after 30 s are rejected. Comparing total video throughputs in Figs. 6(a) and 7(a), total

video throughput with admission control is a little lower than that without admission control. However, throughput is guaranteed for existing flows. Comparing Figs. 6(b) and 7(b), total data throughputs do not have a big difference.

Fig. 8 shows TxLimit with admission control. We observe that ten total video flows are accepted into the system before 30 s; after 30 s, all video flows are rejected. Fig. 8 also shows the effects of TxLimit convergence: after a couple of seconds, TxLimits of different video flows converge to almost the same value fairly.

Fig. 9 shows TXOPBudget with admission control. We observe that the budget decreases ten times and reaches a fairly stable level. After 30 s, there are still some minor fluctuations due to data transmissions, since we did not implemented local data control. We can see that, in the next section, with local data control, the budget will become more stable.

Now, we will study effects of video traffic load on admission control in Figs. 3, 4, 10, and 11.

Fig. 11(a) and (b) shows video throughputs per flow with and without admission control, respectively, for the low video traffic load simulation. We observe that they are the same since, under low video traffic load, even without admission control, video

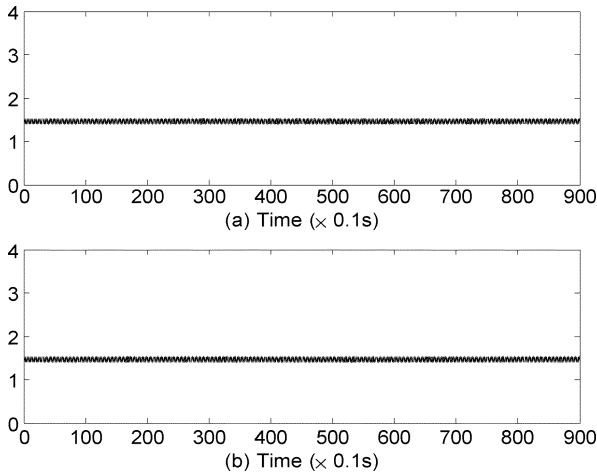


Fig. 11. Video throughput in megabits per second per flow, the low video traffic load simulation, (a) without admission control and (b) with admission control.

traffic flows are pretty much guaranteed. Fig. 10(a) and 10(b) shows video throughputs per flow with and without admission control, respectively, for the medium video traffic load simulation. We observe that throughput per flow without admission control is degraded more [Fig. 10(a)] and, with admission control, throughput per flow [Fig. 10(b)] is improved. Figs. 3(a) and 4(a) show video throughputs per flow with and without admission control, respectively, for the high video traffic load simulation. We observe that throughput per flow without admission control is severely degraded after 30 s [Fig. 3(a)]. With admission control, throughput per flow [Fig. 4(a)] is improved and guaranteed. Comparing Figs. 3, 4, 10, and 11, we can conclude that admission control is not useful under low video traffic, is much more useful under medium video traffic, and is extremely useful under high video traffic. In other words, admission control improved QoS, especially in high video traffic load condition.

B. With Admission Control, but With and Without Local Data Control

In the previous section, we studied admission control under medium data traffic. This section adopts large video traffic and large data traffic. We show that local data control under large data traffic is also needed, since too many data transmissions degrade the system performance, including the existing video flows.

In this section, admission control is always used and we compare the scheme without local data control, the DFM local data-control scheme, and the DT local data-control scheme. The simulation scenario in this section is stated as follows. Initially, there is a video stream and one data station in the system.

- For every 3 s, a video stream arrives to the system until there are total 30 video stream arrivals. Note that later video flows may be rejected.
- For every 3 s, one data station arrives to the system until there are total 18 data stations. All data stations will join the system.

For the DFM scheme, we explain how we obtain initial mappings of (CR_{Low}, CW_{Low}) and (CR_{High}, CW_{High}) . Note that,

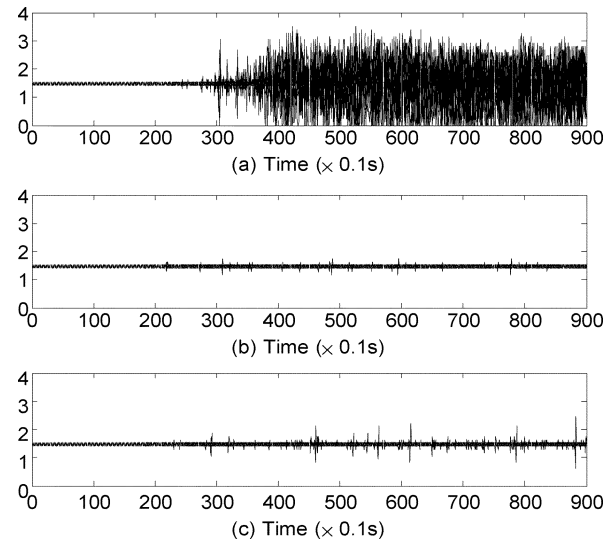


Fig. 12. Video throughput in megabits per second per flow versus simulation time ($\times 0.1$ s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

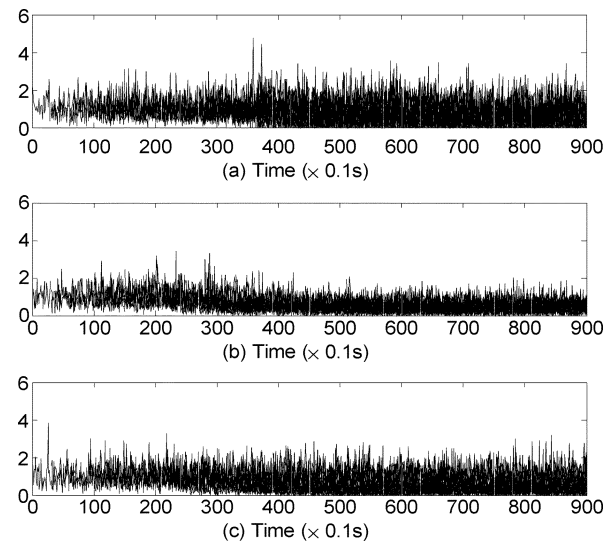


Fig. 13. Data throughput in megabits per second per station versus simulation time ($\times 0.1$ s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

for the DFM scheme, it is difficult/challenging to find these two mapping pairs. Also, this is the drawback of this approach. Here, we obtain them via simulation measurements. We first let $CW_{Low} = CW_{min}[0] = 32$ and conduct measurements to obtain CR_{Low} that are the highest value of CR so that QoS of voice and video is still pretty good. In our experiment, we have $CR_{Low} = 7\%$. Then, we arbitrarily increase both video and data traffic until extremely high traffic load. We observe that CR under this extremely high traffic load is about $CR = 67\%$ and we let $CR_{High} = 67\%$. We then try to find CW_{High} , with which CR will change from CR_{High} back to CR_{Low} or a little smaller. In our case, we have $CW_{High} = 8192$.

For the DT scheme, no mapping pair is needed. We first predefine $\theta = 0.9$ and, later in this section, we also study effects of different θ values. The θ value could be dynamically changed, but in this paper we fixed the θ value.

Fig. 12 shows video throughput per flow for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). As

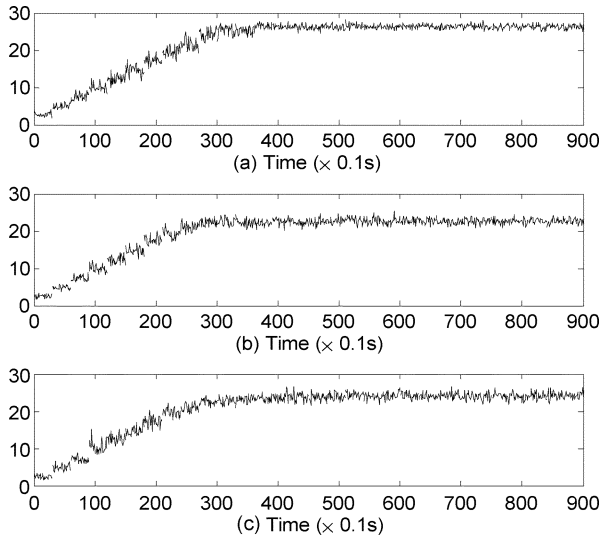


Fig. 14. Total throughput in megabits per second versus simulation time (*0.1 s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

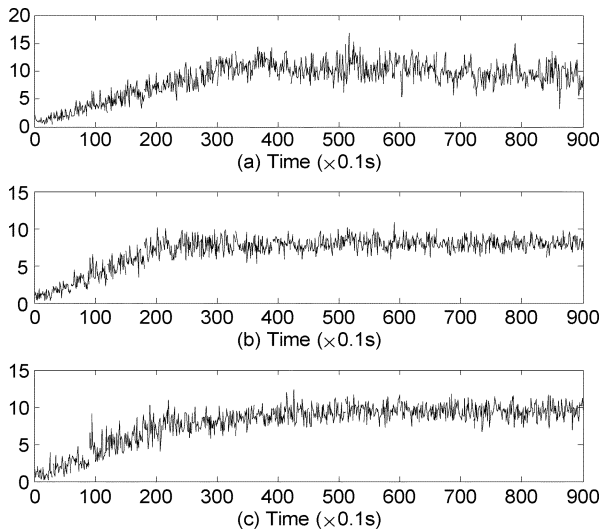


Fig. 15. Total data throughput in megabits per second versus simulation time (*0.1 s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

illustrated in this figure, without local data control, throughput is severely degraded when data traffic is very large after 40 s in Fig. 12(a). With either the DFM scheme [Fig. 12(b)] or the DT scheme [Fig. 12(c)], throughput per flow is improved greatly and guaranteed. Also, we observe that the DFM scheme performs a little better than the DT scheme.

As a tradeoff, the DFM scheme [Fig. 13(b)] and the DT scheme [Fig. 13(c)] have a little smaller data throughput per station than the case without local data control [Fig. 13(a)]. However, we will see in later that total data throughput only decreases a little.

Fig. 14 shows total throughput for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). As illustrated in this figure, total throughput for both the DFM scheme [Fig. 14(b)] and the DT scheme [Fig. 14(c)] are a little lower than total throughput without local data control [Fig. 14(a)]. However, the difference is minor. We observe that our goal has been achieved, i.e., guaranteeing QoS with minor degraded total

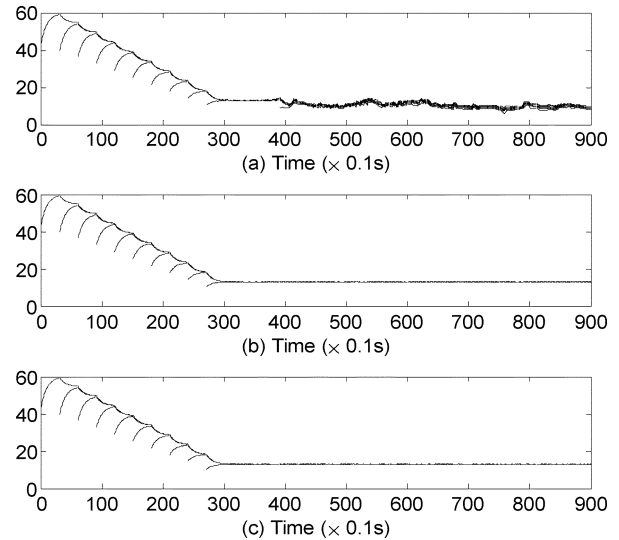


Fig. 16. TxLimit versus simulation time (*0.1 s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

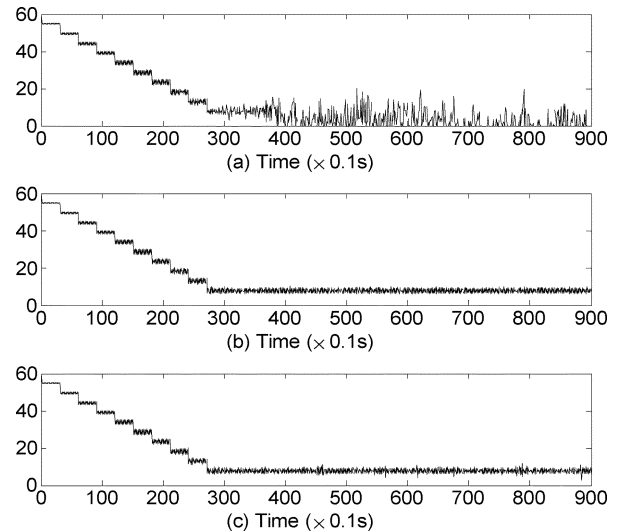


Fig. 17. TXOPBudget versus simulation time (*0.1 s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

throughput. In other words, an approach that does not push too hard normally is a better approach. In fact, this is our design philosophy for both admission control and data control.

Fig. 15 shows total data throughput for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). We observe that total data throughput decreases only a little for the DFM and DT schemes.

Fig. 16 shows TxLimit for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). As illustrated in this figure, without local data control TxLimit is somewhat unstable when data traffic is very large after 40 s in Fig. 16(a). With either the DFM [Fig. 16(b)] or DT schemes [Fig. 16(c)], TxLimit is improved greatly.

Fig. 17 shows TXOPBudget for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). As illustrated in this figure, without local data control, TXOPBudget is very unstable when data traffic is very large after 40 s in Fig. 17(a), and almost becomes zero. With either the DFM scheme

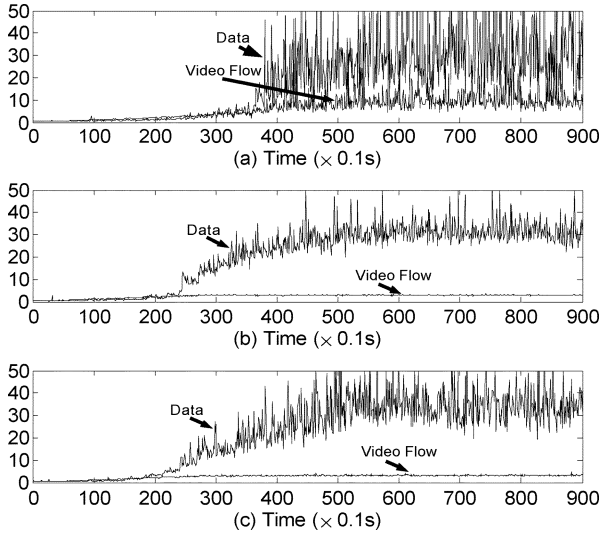


Fig. 18. Average delay in milliseconds versus simulation time ($\times 0.1$ s) (a) without local data control, (b) FDM, and (c) DT ($\theta = 0.9$).

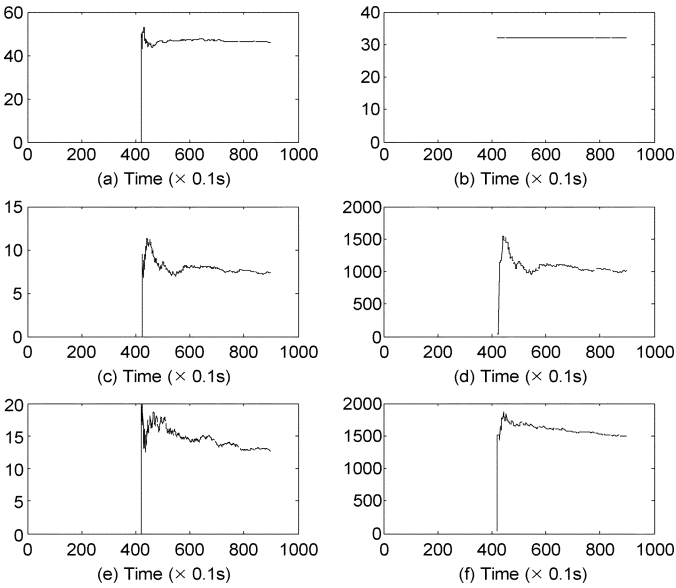


Fig. 19. Collision ratio and $CW_{\min}[0]$ for Station 15 versus simulation time ($CW_{\min} \times 0.1$ s). (a) Collision ratio (without local data control), (b) $CW_{\min}[0]$ (without local data control), (c) collision ratio (DFM), (d) $CW_{\min}[0]$ (DFM), (e) collision ratio (DT, $\theta = 0.9$), and (f) $CW_{\min}[0]$ (DT, $\theta = 0.9$).

[Fig. 17(b)] or the DT scheme [Fig. 17(c)], TXOPBudget is improved greatly.

Fig. 18 shows average delay per flow for the case without local data control, the DFM scheme, and the DT ($\theta = 0.9$). As illustrated in these figures, with local data control [Fig. 18(b) and (c)], delay for video flows has been greatly improved.

Fig. 19 show collision ratio and $CW_{\min}[0]$ of a random chosen data station, Station 15 for (a) and (b) the case without local data control, (c) and (d) the DFM scheme, and (e) and (f) the DT scheme ($\theta = 0.9$). Since Station 15 joins the system at the time around 40 s, before 40 s, both collision ratio and $CW_{\min}[0]$ are zeros. For all parts except (b), the value is fluctuated a little bit and then becomes stable. Without local data control, collision ratio [Fig. 19(a)] is about 40%. For the DFM scheme, collision ratio [Fig. 19(c)] is controlled around 7%. For

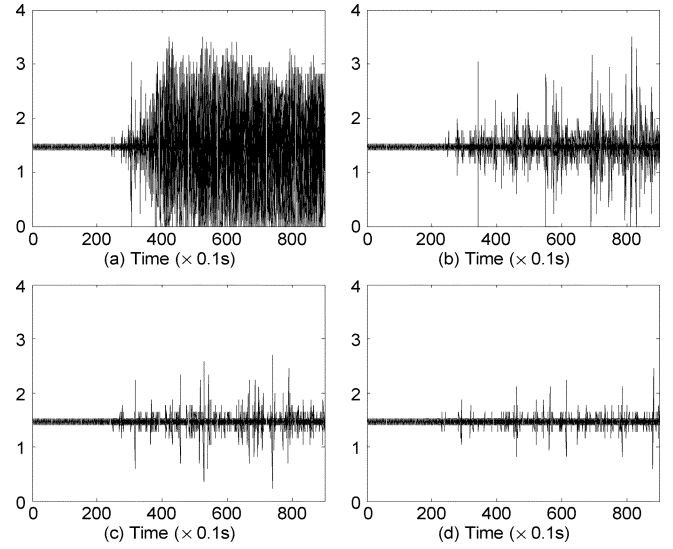


Fig. 20. Effects of θ on video throughput per flow (a) without local data control, (b) DT ($\theta = 0.3$), (c) DT ($\theta = 0.6$), and (d) DT ($\theta = 0.9$).

the DT scheme, collision ratio [Fig. 19(e)] is controlled around 12%. Without local data control, $CW_{\min}[0]$ [Fig. 19(b)] is fixed and equals 32. For the DFM scheme, $CW_{\min}[0]$ [Fig. 19(d)] is controlled around 1000. For the DT scheme, $CW_{\min}[0]$ [Fig. 19(f)] is controlled around 1500.

Fig. 20 shows the effects θ on video throughput per flow for case (a) without local data control, (b) DT ($\theta = 0.3$), (c) DT ($\theta = 0.6$), and (d) DT ($\theta = 0.9$). Without local data control, video throughput per flow is severely degraded when data traffic load is very high. With the DT local data control scheme, video throughput per flow has greatly improved, especially for a large θ value.

C. Fairness Analysis for Local Data Control

In this section, we study fairness issues for local data controls. Assume that there are total n active stations ($i = 1, \dots, n$). Let x_i denote a performance metric (such as throughput, delay, the initial window size, etc.) of the station i . Fairness factor is defined as [14]

$$f(x_1, x_2, \dots, x_n) = \frac{\left(\sum_{i=1}^n x_i \right)^2}{n \sum_{i=1}^n x_i^2}. \quad (19)$$

The fairness factor lies between 0 and 1. If the values equal to each other, the factor equals one and it is equally fair. We adopt (19) to study fairness with throughput and delay for different local control schemes.

Fig. 21 shows throughput fairness factor for different schemes. As illustrated in this figure, the DFM scheme has the best throughput fairness factor; the case without local data control has a throughput fairness factor 0.78, which is similar to the DT scheme. For the DT scheme, the case of $\theta = 0.9$ has a little less throughput fairness factor than those of $\theta = 0.6$ and $\theta = 0.3$, since a too large θ value may be a little progressive. However, different θ values do not influence a lot on the throughput fairness factor.

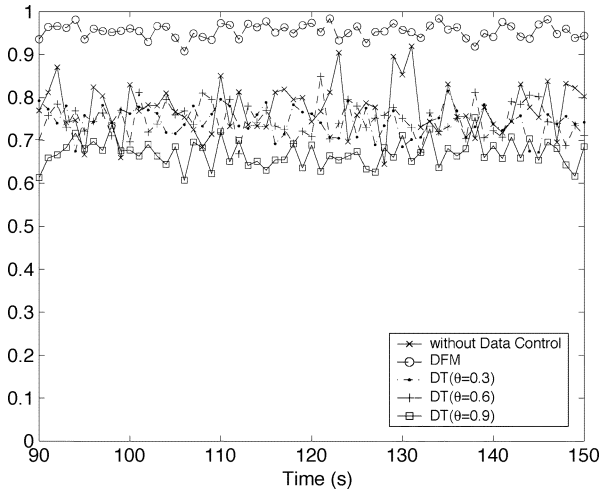


Fig. 21. Throughput fairness factor.

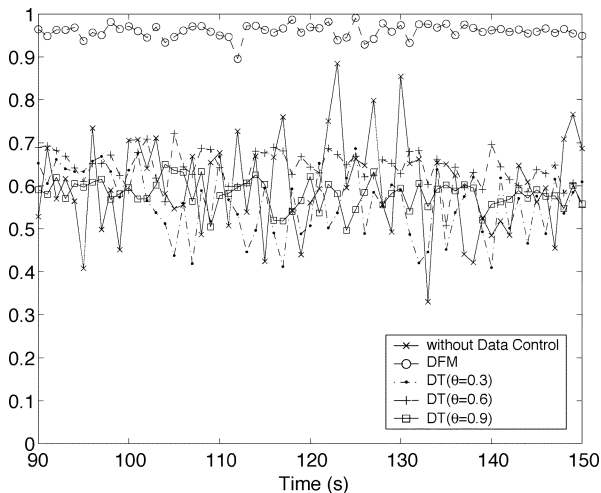


Fig. 22. Delay fairness factor.

Fig. 22 shows the delay fairness factor for different schemes. As illustrated in this figure, the DFM scheme has the best delay fairness factor; the case without local data control has a throughput fairness factor round 0.6, which is similar to the DT scheme. For the DT scheme, different θ values have similar delay fairness factors. We also observe that the case without local data control has a larger variance than local data-control schemes.

In summary, the DFM scheme has the best fairness factor overall and the DT scheme has a similar fairness factor to the case without local data control for delay and throughput.

VI. MULTIHOP *AD HOC* NETWORKS

For one-hop wireless *ad hoc* networks, our goal is well achieved, i.e., QoS performance can be greatly improved and guaranteed while the utilization of the system is very good too. We believe that our work is one novel step toward QoS support in multihop *ad hoc* wireless networks.

In this section, we first provide short surveys for MAC protocols and QoS routing for *ad hoc* networks in Sections VI-A and VI-B, respectively. Finally, we discuss how to apply proposed approaches into multihop *ad hoc* networks in Section VI-C.

A. Wireless MAC

Wireless MAC protocols are classified into two categories: distributed and centralized [15]. Distributed MAC protocols can be further classified into random-access and token-passing protocols. Centralized MAC protocols can be further classified into polling protocols and combined random-access and polling protocols.

In a random-access protocol, stations compete the channel access and collisions happen when multiple stations transmit in an overlap time. The first random-access protocol is ALOHA [16], [17], in which a station transmits a frame if available and retransmits it in a random time if collided. Slot ALOHA [18] divides time into slots and stations can only transmit at the beginning of each slot. Slot ALOHA improves performance of ALOHA by reducing the vulnerable period of a transmission into half length with increased device complexity to be capable of time synchronization among stations. The ALOHA protocol can be further improved if stations can sense the channel before transmissions, i.e., carrier sense multiple access (CSMA) [19], in which device complexity is further increased by being capable of sensing the channel. CSMA can further be improved if stations are capable of conducting collision detection (CD), i.e., CSMA/CD [20] (Ethernet, IEEE 802.3) so that, if a collision is detected, transmissions are stopped immediately. In CSMA/CD, if collisions happen, stations choose a random time before retransmitting and binary exponential backoff is adopted for further retransmissions. To perform CD, stations need to have the ability of performing transmitting and listening at the same time. Stations in the IEEE 802.11 wireless local area networks (LANs) are not capable of transmitting and listening at the same time; therefore, collision avoidance (CA) is adopted by both physical and virtual sensing. To reduce the hidden terminal problem and the possibility of collisions, the RTS/CTS mechanism is also adopted in the IEEE 802.11 DCF, as well as an ACK message for error control due to the unreliable nature of the wireless medium.

In token-passing protocols, such as the token ring or wireless token-passing protocol, a token is exchanged among stations in a distributed manner. A station can transmit a frame if it has the token and passes the token to the next station after finishing transmission.

In both polling and token-passing protocols, stations access the channel in an orderly manner, usually in a round-robin fashion [15]. In polling protocols such as Bluetooth and the PCF in the IEEE 802.11 WLANs, the master polls each station and the station sends data in response to the poll. Both polling and token-passing protocols may provide guaranteed QoS, but token-passing protocols have some disadvantages, since token loss is common and token recovery has a huge overhead.

Request-grant mechanisms are adopted in most combined random-access and polling protocols [15], in which each station sends a request to the master with required transmission time and bandwidth using a random-access protocol. The master then decides whether to grant the request and allocates bandwidth. Combined random-access and polling protocols can be further classified into random reservation access and demand assignment protocols [15]. In random reservation access protocols, implicit rules for reserving upstream bandwidth, are used

such as a rule: a successful request results in a periodic reservation of an upstream slot. In demand-assignment protocols, the master controls upstream data transmissions according to their QoS requirements by collecting all the requests and uses scheduling algorithms to make bandwidth allocations.

B. QoS Routing Protocols for Ad Hoc Networks

Routing protocols in *ad hoc* wireless networks can be classified into two categories: table-driven and on-demand routing protocols. In table-driven routing protocols, paths are found for connections based on the whole network topology information. On the other hand, for on-demand routing protocols, paths are found only when packets needed to be sent.

In table-driven routing protocols, each station maintains a routing table, including paths to all stations in the network. Table-driven protocols are further classified into flat and hierarchical protocols. A flat protocol, such as the destination sequenced distance vector (DSDV) protocol, adopts a flat network topology and each station maintains paths to all stations in the network. In hierarchical protocols, based on the hierarchical cluster technology and topology, stations are divided into several clusters and cluster headers help to find paths so that only partial information of the wireless network are needed.

In on-demand routing protocols, stations do not need to maintain routing tables for all nodes in the network as destinations and to periodically update them. They only need to learn a path and add route information into the route table when they have packets to send. On-demand routing protocols have less update information overhead as compared with table-driven routing protocols. Many on-demand protocols exist, such as the dynamic source routing (DSR) protocol [21], *ad hoc* on-demand distance vector (AODV) protocol [22], and temporally ordered routing protocol (TORA) [23], etc.

DSR adopts source routing with advantages such as loop free and efficiency for learning via forwarding and overhearing. DSR has two functions: route discovery and route maintenance. Stations first search route cache to see if it has a hit. If a cache misses, it initiates a route-discovery mechanism by sending a route request message. When an intermediate station receives this route request message, it searches its own cache to see if it has a cache hit. If it does not, it appends its identification (ID) to the packet and forwards the packet to the next node. This procedure continues until either station with a route to the destination is found or the destination station receives the packet. In either case, the station sends a route reply packet which has a list (path) of all of the intermediate stations.

In AODV, each node maintains a routing table containing entries to destinations. Each entry includes destination address, next hop neighbor address, distance to the destination, life time, and precursor nodes. If the source node cannot find a route to the destination in its routing table, it issues route discovery and broadcasts a route request packet including source node address, destination address, and packet broadcast ID, which is increased automatically. Once an intermediate node receives the route-request packet, it first checks whether it received the packet before; if the packet was received before, the packet is discarded; otherwise, the node checks whether it is the destination node and whether it has an entry to the destination. If one of the answers

is yes, the node sends back a route reply packet to the source node following the route; otherwise, the intermediate node creates a reverse entry to the source node in the route table for forwarding packets in the future. Then, it broadcasts the route request packet to its neighbors. Finally, the destination node receives the route-request packet and sends a reply message to the source node. When an intermediate node receives the reply packet, it creates reverse entry including destination node address, address of next hop to destination, and distance to the destination in order to forward packets to the destination in the future. It forwards the reply packet to the next hop toward the source node. The route is maintained as long as there are data packets periodically traveling from the source to the destination along the route. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node-routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error message to the source node to inform it of the now unreachable destination(s). After receiving the message, if the source node still desires the route, it can reinitiate route discovery.

Many QoS routing protocols have been proposed to improve QoS requirements, such as extensions of DSR over synchronous time division multiple access (TDMA) [24], QoS-AODV [25], [26], QoS-TORA [25], etc. Most of the QoS models assume TDMA, CDMA-over-TDMA, etc. However, TDMA systems have fundamental problems in terms of synchronization, robustness, and scalability for many stations, since there are no central control stations. Furthermore, TDMA systems are not practical for *ad hoc* networks mainly due to there being no central control stations and stations may move.

C. Discussions and Applying Proposed Schemes Into Multihop Ad Hoc Networks

Polling schemes and combined random-access and polling schemes are not appropriate *ad hoc* networks, since there are no central controlled stations in *ad hoc* networks. TDMA-based protocols are not scalable when mobility exists and the number of stations is unknown and keeps changing. Reservation-based protocols waste a lot of bandwidth, cause starvations of other stations, and are difficult to maintain synchronization. Token-passing protocols are not reliable and the recovery of tokens causes a huge overhead.

On the other hand, the IEEE 802.11 DCF (CSMA/CA) is the only available MAC layer on the market for *ad hoc* network protocols. Our proposed local data-control schemes and admission control for the IEEE 802.11e EDCA will be a good choice for *ad hoc* networks in terms of practical value. Compared to other MAC with QoS constraints, our proposed schemes have the minimum number of assumptions and can be easily implemented in reality.

We discuss some issues on how to apply the proposed schemes into multihop *ad hoc* networks as follows.

- **QoS MAC:** we will use the IEEE 802.11e EDCA coupled with the proposed local data and admission control. Since the proposed local data and admission controls are fully distributed algorithms without the presence of any AP, they can be used in *ad hoc* networks without changes.

- **QoS Routing:** Any on-demand routing protocol will work for us, such as DSR, AODV, etc. We choose AODV since it is more consistent with our approaches. Our purpose is not to reserve bandwidth along the path, but just to find a path. When the first group of frames for a real-time (video/voice) flow is routed along the path, this flow is subjected to admission control in each new hop it goes through for the first time. If in any hop the flow is rejected, the whole flow will be rejected via the following back-deny mechanism. When the flow is rejected in one hop, the source station of the hop stops forwarding transmissions of this flow and sends a rejection ACK message to the source station of the previous hop, who also repeats the same procedure. Finally, the source station of this flow receives the a rejection ACK message and either stops this flow or tries another path. This procedure is exactly similar to those in AODV. In other words, path failure for real-time transmissions sometime may happen due to limited resource. The route is maintained as long as there are data packets periodically traveling from the source to the destination along the route.
- **Robustness for routing:** In order to improve routing robustness, multiple paths can be obtained before real transmissions happen, so that in the case that one path fails, another path can be used immediately. Since bandwidth reservations are not needed during route discovery, there is no bandwidth wasted for reservations.

Performance evaluation for the above proposed schemes for multihop *ad hoc* networks is our current and future work.

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose and study a fully disturbed measurement-based admission-control scheme and two local data-control schemes for one-hop *ad hoc* wireless networks when an access point is not present. Local data-control schemes include the DFM and DT schemes. We conduct extensive simulations and our simulation results indicate the following conclusions.

- Without admission control, when video traffic load is large enough, throughputs of all flows are severely degraded. With admission control, throughputs of video flows are improved and guaranteed. Total throughput with admission control is a little lower than total throughput without admission control. However, the difference is minor. Our goal of admission control has been achieved, i.e., guaranteeing QoS with minor degraded total throughput. In other words, an approach that does not push too hard normally is a better approach. In fact, this is our design philosophy for both admission and data control.
- Simulations also show the effects of TxLimit convergence: after some time, TxLimits of different video streams converge almost the same value fairly.
- Admission control improved QoS, especially in the high video traffic load condition.
- Without local data control, throughput is severely degraded when data traffic is very large and, with either the DFM or DT scheme, video throughput per flow is improved greatly and guaranteed. Total throughput for

both the DFM and DT schemes are a little lower than the total throughput without local data control. However, the difference is minor. We observe that our goal has been achieved, i.e., guaranteeing QoS with minor degraded total throughput. With local data control, delay for video flows has been greatly improved.

- The DFM scheme has the best fairness factor overall and the DT scheme has a similar fairness factor to the case without local data control for delay and throughput.
- For local data control, the DFM scheme has a drawback: it is difficult to obtain two or more pairs of accurate mapping values. The DT scheme is much simpler and does not require such information. The DFM scheme seems to perform a little better than the DT scheme in terms of data traffic control and fairness. However, for practical issues, we recommend the DT scheme.

For one-hop *ad hoc* wireless networks, our goal is well achieved, i.e., QoS performance can be greatly improved and guaranteed while the utilization of the system is very good too.

We further discussed how to apply our proposed schemes to multihop *ad hoc* wireless networks in the previous section. A comprehensive performance evaluation for multihop *ad hoc networks* is our current and future work.

REFERENCES

- [1] S. Chakrabarti and A. Mishra, "QoS issues in ad hoc wireless networks," *IEEE Commun. Mag.*, vol. 39, pp. 142–148, Feb. 2001.
- [2] J. L. Sobrinho and A. S. Krishnakumar, "Quality-of-service in ad hoc carrier sense multiple access wireless networks," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1353–1414, Aug. 1999.
- [3] A. Iwata *et al.*, "Scalable routing strategies for *ad hoc* wireless networks," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1369–1379, Aug. 1999.
- [4] C. R. Lin and J.-S. Liu, "QoS routing in *ad hoc* wireless networks," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1426–1438, Aug. 1999.
- [5] R. Sivakumar, P. Sinha, and V. Bhargavan, "CEDAR: A core extraction distributed *ad hoc* routing algorithm," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1454–1465, Aug. 1999.
- [6] S. Chen and K. Nahrstedt, "Distributed quality-of-service routing in *ad hoc* networks," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1488–1505, Aug. 1999.
- [7] S. Chen, "Routing support for providing guaranteed end-to-end quality-of-service," Ph.D. dissertation, Univ. Illinois, Urbana-Champaign, 1999.
- [8] R. Ramanathan and M. Steenstrup, "Hierarchically organized, multihop mobile wireless networks for quality of service support," *Mobile Network Apps.*, vol. 3, pp. 101–119, 1998.
- [9] *IEEE 802.11 WG Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, Aug. 1999.
- [10] D.-J. Deng and R.-S. Chang, "A priority scheme for IEEE 802.11 DCF access method," *IEICE Trans. Commun.*, vol. E82-B, no. 1, pp. 96–102, 1999.
- [11] *IEEE 802.11 WG, Draft Supplement to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)*, IEEE Std 802.11e/D3.3.2, Nov. 2002.
- [12] *IEEE 802.11b, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification: High-Speed Physical Layer Extension in the 2.4 GHz Band*, Sept. 1999.
- [13] *IEEE 802.11a WG, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification: High-Speed Physical Layer in the 5 GHz Band*, Sept. 1999.
- [14] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. New York: Wiley, 1991.
- [15] A. Chandra, V. Gummalla, and J. O. Limb, "Wireless medium access control protocols," *IEEE Commun. Surv. Tutorials*, pp. 2–15, 2000.

- [16] N. Abramson, "The ALOHA system: Another alternative for computer communications," in *Proc. AFIPS Conf.*, 1970, pp. 281–285.
- [17] —, *The ALOHA System, Computer Communication Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1973.
- [18] L. G. Robert, "Aloha packet system with and without slots and capture," *ACM SIGCOMM Comput. Commun. Rev.*, pp. 28–42, 1972.
- [19] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I: Carrier sense multiple-access models and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. COM-23, pp. 1400–1416, Dec. 1975.
- [20] *IEEE/ANSI Standard: Carrier Sense Multiple Access With Collision Detection*, 1985.
- [21] C. E. Perkins, *Ad Hoc Networking*. Reading, MA: Addison-Wesley, 2001.
- [22] (1998, Aug.). *Ad hoc on demand distance vector (AODV) routing*
- [23] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proc. INFOCOM'97*, 1997, pp. 1405–1413.
- [24] W.-H. Liao, Y.-C. Tseng, and K.-P. Shih, "A TDMA-based bandwidth reservation protocol for QoS routing in a wireless mobile *ad hoc* network. Communications," in *Proc. IEEE Int. Conf. Communications (ICC'02)*, pp. 3186–3190.
- [25] I. Gerasimov and R. Simon, "A bandwidth-reservation mechanism for on demand *ad hoc* path finding," in *IEEE/SCS 35th Annu. Simulation Symp.*, San Diego, CA, Apr. 2002, pp. 27–33.
- [26] —, "Performance analysis for *ad hoc* QoS routing protocols," in *Proc. Int. Mobility and Wireless Access Workshop (MobiWac'02)*, pp. 87–94.



Yang Xiao (S'98–M'01–SM'04) received the Ph.D. degree in computer science and engineering from Wright State University, Dayton, OH, in 2001.

He has been a Software Engineer, Senior Software Engineer, and Technical Lead in the computer industry from 1991 to 1996. From August 2001 to August 2002, he was with Micro Linear-Salt Lake City Design Center, UT, as an MAC Architect involving the IEEE 802.11 (wireless local area network) standard enhancement work. Since August 2002, he has been an Assistant Professor of computer science at The University of Memphis, Memphis, TN. He currently serves on the Editorial Boards of the *Journal of Wireless and Mobile Computing* and the *International Journal of Signal Processing*. He served as a Guest Editor for the *Journal of Wireless Communications and Mobile Computing* Special Issue on "Mobility, Paging and Quality of Service Management for Future Wireless Networks" in 2004, as a guest editor for the *International Journal of Wireless and Mobile Computing* Special Issue on "Medium Access Control for WLAN's, WPAN's, *Ad Hoc* Networks, and Sensor Networks" in 2004, and as an Associate Guest Editor for the *International Journal of High Performance Computing and Networking* special issue on "Parallel and Distributed Computing, Applications and Technologies" in 2003. His current research interests include wireless local area networks, wireless personal area networks (PANs), and mobile cellular networks.

Dr. Xiao was awarded the Dayton Area Graduate Studies Institute (DAGSI) Ph.D. Fellowship from 1996 to 2001. He was a Recipient of the 1999 Graduate Student Excellence Award in recognition of outstanding academic achievements in the computer science and engineering Ph.D. program at Wright State University. He is a Voting Member of the IEEE 802.11 Working Group and a Member of the Association for Computing Machinery (ACM) and of the IEEE Computer, Communications, and Vehicular Technology Societies. He serves as a Symposium Cochair for the Symposium on Data Base Management in Wireless Network Environments for the 2003 IEEE Vehicular Technology Conference (IEEE VTC'03). He serves as a Technical Program Committee Member for the IEEE ICC'05, IEEE WCNC'05, IEEE ICDCS'04, IEEE GLOBECOM'04, IEEE WCNC'04, IEEE ICCCN'04, IEEE PIMRC'04, ACM WMASH'04, ACM SAC'04, WLN'04, MWN'04, MDC'04, and CIT'04.



Haizhon Li (S'03) received the M.S. degree in computer science from The University of Memphis, Memphis, TN, in 2003 and is currently working toward the Ph.D. degree in computer science at the University of Memphis.

His research interests include quality of service and medium-access control enhancement for IEEE 802.11 wireless local area networks (WLANs), performance analysis, and integration of WLANs, wireless PANs, and third-generation cellular networks.