Distributed Admission Control in Wireless Mesh Networks: Models, Algorithms, and Evaluation

Jihene Rezgui, Abdelhakim Hafid, and Michel Gendreau

***Abstract*—Wireless mesh networks (WMNs) have attracted increasing attention from the research community as a high-performance and low-cost solution to last-mile broadband Internet access. In WMNs, admission control is deployed to efficiently control different traffic loads and prevent the network from being overloaded. This paper introduces a distributed admission control scheme for WMNs, namely, routing on cliques admission control (RCAC). In particular, we propose an analytical model to compute the appropriate acceptance ratio and guarantee that the packet loss probability (PLP) in the network does not exceed a threshold value. The model also allows computing end-to-end delay to process flow requests with delay constraints. RCAC achieves scalability since it partitions the network into cliques, and only clique heads (CHs) are involved in the admission-control procedure. Using extensive simulations, we demonstrate that our RCAC achieves high resource utilization by providing lower blocking probabilities in a dynamic traffic-load environment while satisfying quality-of-service (QoS) constraints in terms of PLP and end-to-end delay. Moreover, we show that a contention access (CA) enforced with our RCAC outperforms the mesh deterministic access (MDA).**

***Index Terms*—Admission control, multichannel, quality of service (QoS), wireless mesh networks (WMNs).**

I. Introduction

WIRELESS mesh networks (WMNs) have recently emerged as a promising technology for next-generation wireless networks. A WMN consists of two types of nodes: 1) mesh clients (MCs) and 2) mesh routers (MRs). The MRs form a wireless mesh backbone infrastructure that forwards most of the traffic between MCs and Internet gateways. In general, MRs have very rare mobility and operate just like stationary routers, except that they are connected by wireless links using the very popular IEEE 802.11 wireless local area network (WLAN) standard or other wireless technologies such as WiMax. Using more than one radio interface in each MR allows for more channel diversity, resulting in less interference and, therefore, more throughput and capacity. However, this only improves the best effort traffic since supporting QoS for real-time traffic in WMNs remains an open challenge.

Admission control is one of the key traffic-management mechanisms that must be deployed to provide QoS support. A new traffic flow is accepted into the network only if there are sufficient resources. Admission control is the premise to the implementation of QoS routing, channel assignment, and multiple radio/channel scheduling, which have been proposed to improve capacity and maximize throughput in WMNs. Indeed, when the network is overloaded, none of these schemes can prevent QoS degradation.

The accuracy of admission control depends upon how well the network capacity is estimated. The estimation is difficult to obtain because, compared with wired networks, the links in WMNs are inherently shared, because of interferences, and difficult to isolate; this fact makes the performance of WMNs difficult to control. It is crucial that admission control considers both local resources and resources at neighboring nodes when analyzing the network performance [4], [5]; the reason is that interferences among links cause performance degradation, e.g., two interfering links that are simultaneously active often provide less throughput than two separated links.

In this paper, we propose a distributed admission control mechanism for WMNs, namely, routing on cliques admission control (RCAC); a preliminary version of RCAC has appeared in [6]. RCAC accepts a new flow request only when there are enough available resources to carry the flow while satisfying predefined thresholds of packet loss probability (PLP) and end-to-end delay. This will avoid situations in which uncontrolled resource usage leads to network breakdown (i.e., severe congestion). RCAC partitions the WMN into cliques, in which all vertices are adjacent to each other. A maximal clique is a clique that belongs to no other larger cliques. Only clique heads (CHs) are involved in the admission-control procedure; this makes RCAC scalable for large-sized networks. Inside a clique, RCAC computes the available bandwidth while making use of local bandwidth information and neighboring bandwidth information; this is necessary to take care of interferences among one- and two-hop-away nodes. To the best of our knowledge, RCAC is the first admission-control mechanism for WMNs to consider two QoS parameters: 1) packet loss and 2) end-to-end delay in multichannel and multiradio WMNs. RCAC attempts to answer the following question: For a given WMN, can new flows be accepted into the network while keeping the PLP under the PLP threshold and the end-to-end delay under the threshold value?

Manuscript received January 20, 2009; revised October 16, 2009 and June 19, 2009. First published December 28, 2009; current version published March 19, 2010. The review of this paper was coordinated by Prof. S. Ci.

J. Rezgui with the Network Research Laboratory, University of Montréal, Montréal, QC H3C 1J4 Canada, and also with the Interuniversity Research Center on Enterprise Networks, Logistics, and Transportation, University of Montréal, Montréal, QC H3C 3J7, Canada (e-mail: rezguiji@iro.umontreal.ca).

A. Hafid is with the Network Research Laboratory, University of Montréal, Montréal, QC H3C 1J4 Canada (e-mail: ahafid@iro.umontreal.ca).

M. Gendreau is with the Département de Mathématiques et Génie Industriel, École Polytechnique de Montréal, Montréal, QC H3T 1J4 Canada (e-mail: michel.gendreau@cirrelt.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2009.2039360

In our proposed approach, we take into consideration the knowledge of both local and neighboring resources in a distributed stochastic analytical model with two QoS parameters: 1) delay and 2) packet loss. We model interconnected CHs as a queuing network, and we approximate the PLP with the overflow probability in each clique; to this end, we estimate the total packet arrival at time t in each clique. The objective of our proposal is to compute an acceptance ratio for a given PLP. Indeed, for a threshold value of PLP, we are able to determine the number of flows that can be accepted into the network while satisfying the threshold. Our proposal also takes end-to-end delay into account when processing new flow requests.

The remainder of this paper is organized as follows. Section II presents related work. Section III presents our notations, assumptions, and network model followed by an analytical model in Section IV. Section V elaborates our admission-control algorithm and enhanced routing protocol in detail. Section VI evaluates the proposed admission control via simulations. Finally, Section VII concludes this paper.

II. Related Work

The new challenges in WMNs require more research effort from different perspectives to provide QoS management. The authors in [7] report that it is necessary to have a mechanism for admission control. However, they do not present any specific solution. Studies specifically focusing on admission control over IEEE 802.11 [4], [5], [8], [9] have mainly considered the available bandwidth to decide whether the request is accepted or not. In the stateless wireless ad hoc network (SWAN) [8], the admission controller listens to all packet transmissions to collect information about bandwidth and congestion. It proceeds by sending probe messages. However, probing causes a lot of overhead and packet loss. In addition, SWAN does not consider the fact that two nodes could contend for a channel even without directly communicating with each other. The authors in [4] proposed the contention-aware admission control protocol (CACP) mechanism. The CACP provides admission control for flows in a single-channel ad hoc network based on the knowledge of both local resources at a node and the effect of admitting new flows on neighboring nodes. A scheme closely related to the CACP was proposed in [5], which integrates admission control with ad hoc routing and channel reuse due to parallel transmissions for a more accurate estimation of channel utilization. In [9], the authors developed a measurement-based capacity utilization model for an IEEE 802.11-based mesh network; they also introduced routing metrics such as the maximum residual feasible path and new strategies like routing using call statistics. The proposed model provides the available capacity of each node for the admission of new voice-over-IP (VoIP) calls. The proposed approach [9] has been evaluated for a single-channel mesh network using only VoIP traffic. The applicability/adaptation of this approach in the case of multiple channels and traffic types has not been investigated.

Several schemes have been proposed to evaluate the expected bandwidth of a wireless network over IEEE 802.11e to provide a criterion for admission decision [10]–[13]. In [10], the authors proposed a method where each node measures both the occupied bandwidth and the average collision ratio; the measured value is compared with a given threshold; then, a decision to accept or not a new flow is made using a simple rule. The authors in [11] introduced an analytical model to evaluate the expected bandwidth and packet delay of each class of traffic based on the enhanced distribution coordination function (EDCF). This model provides a criterion for admission decision, as well as a theoretical analysis for EDCF. The proposed admission-control strategy satisfies the required bandwidth and limits the packet delay of each traffic class to a predefined level. In [12], the quality access point measures the medium utilization and confirms the transmission opportunity budget (TOXP) through beacon signals for each access category (AC). If the TOXP is consumed for one AC, then the new flow could not gain transmission time, and current flows could not increase their transmission time. In [13], the authors proposed a centralized admission control mechanism; their model is based on the concept of conflict graph. The major problem with this model is that the utilization of the conflict graph is highly complex; even for a moderate-sized network, the number of interference constraints can be hundreds of thousands. The model works well in a multihop single channel for a small-sized network. However, the approach is centralized, which is not convenient for large networks.

The mesh deterministic access (MDA) studied in [14]–[16] extends the typical IEEE 802.11 medium instantaneous reservation procedure [which is known as the virtual carrier sensing (V-CS)] to a more advanced reservation procedure using scheduled MDA opportunities (MDAOPs) within a two-hop neighborhood. MDA aims to provide stringent medium access control (MAC) delay guarantees for real-time services such as VoIP. IEEE 802.11 V-CS [14] is performed by a four-way handshake procedure in which request-to-send, clear-to-send, data, and acknowledgment packets are exchanged between two communicating nodes, while a network-allocation vector is set by the other nodes in the physical sensing range. V-CS correctly works in a single-hop wireless network but can cause severe interference in wireless links that are multiple hops away and are sharing the same channel within an overlapped transmission or interference range. Therefore, in multihop wireless networks, such as WMNs, a different access mechanism, namely, MDA, with less multihop interference, was adopted in the IEEE 802.11s draft amendment to provide stringent delay bounds. MDAOPs are first negotiated between neighboring mesh nodes by exchanging broadcast setup messages; then, MDAOP reservations in multiple time-slot units, during the delivery traffic-indication message (DTIM) periodic interval, are performed. To limit the message broadcast signaling overhead, MDA-related messages are sent only within the two-hop neighborhood.

However, interference outside the two-hop neighborhood can still occur with MDA; this may degrade the WMN performance. The authors in [16] tackle this problem using dynamic relocation (DR) of conflicting MDA time slots that are two hops away from each requesting node. In their proposed solution, if enough interference is detected outside the two-hop neighborhood, then flows are dynamically relocated, depending on their elapsed duration in the network. In fact, "older" flows will be

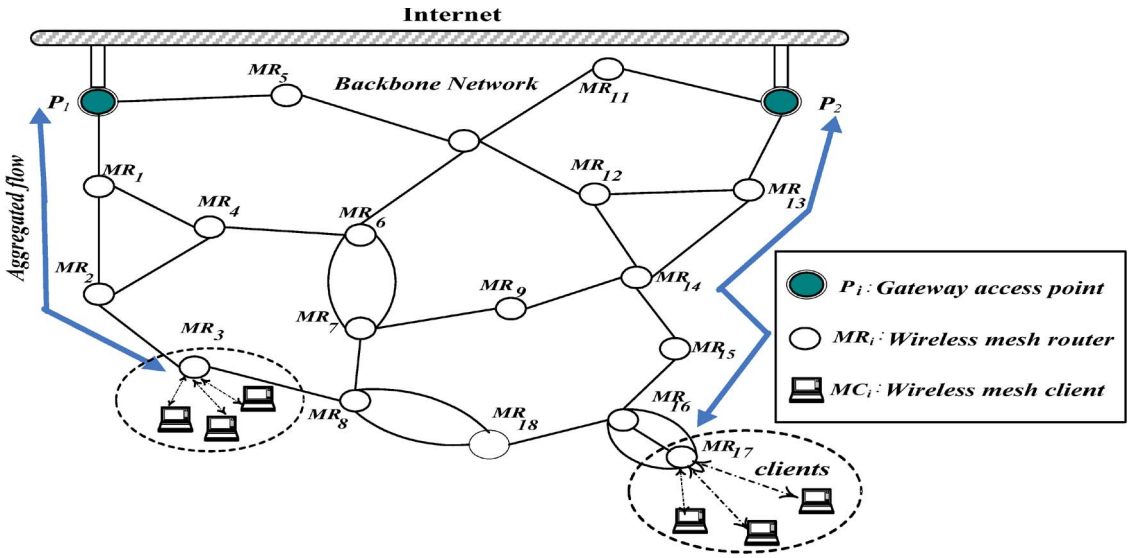

Fig. 1. Wireless mesh network.

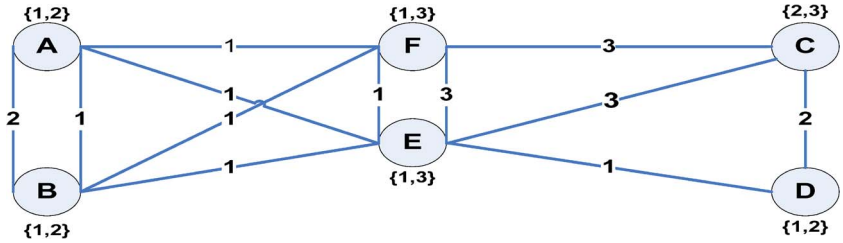

Fig. 2. Connectivity graph: G_A.

less likely to be relocated, i.e., its MDAOPs will be moved to another available time slot.

III. Network Model

In this section, we propose a network model for WMNs and illustrate how an equivalent queuing and stochastic network model can be constructed. In particular, we first define key concepts, namely, connectivity graph and cliques, and then present the assumptions/notations used in the rest of this paper. The network model will be used in Section V to develop the proposed distributed admission control mechanism for WMNs.

A. Connectivity Graph

We consider a multihop WMN, as illustrated in Fig. 1. In this network, MRs (e.g., MR_3 and MR_{17}) aggregate and forward the traffic from the MCs that are associated with them. They communicate with each other to form a multihop wireless backbone network. This backbone network forwards the user traffic to the gateway access points (e.g., P_1 and P_2), which are connected to the Internet.

We formally model the backbone of a WMN as an undirected graph called connectivity graph $G = (V, E)$, where V represents the set of mesh nodes, and E represents the set of edges between these nodes. Among these nodes, $P \subset V$ are the gateway access points that connect to the Internet. In the rest of this paper, MRs and gateway access points are collectively called mesh nodes. $\forall (u, v) \in V$, an edge $e = (u, v) \in E$ if the distance between u and v, which is denoted as $d(u, v)$, is smaller than the minimum range, which is denoted $\min(r_u, r_v)$, of u and v (i.e., $d(u, v) \leq \min(r_u, r_v)$), where r_u and r_v represent the radio transmission ranges of nodes u and v, respectively. Since we consider a multiradio and multichannel WMN, channel assignment is needed.

The connectivity graph with channel assignment is denoted as $G_A = (V, E, A_G)$, where $A_G = \{A_G(u), \forall\, u \in V\}$, and $A_G(u)$ is the set of channels assigned to $u\}$. We denote NC as the number of channels per radio and NR as the number of radios per node; typically, we have NR $\leq$ NC.

Fig. 2 shows an example of a connectivity graph $G_A = (V = \{A, B, C, D, E, F\}, E = \{(A, B), \ldots, (C, D)\}$, $A_G = \{A_G(A), \ldots, A_G(F)\})$ in which we connect two nodes u and v if they share the same channel and the distance between them is smaller than or equal to $\min(r_u, r_v)$. In this example, NR $= 2$ radios, and each node is labeled with its channel assignment; for example, D, with two radios, is assigned channels 1 and 2 $(A_G(D) = \{1, 2\})$. The radio transmission range of C and D is 200 and 250 m, respectively, the distance between C and D is smaller than $\min(200, 250)$, and they share channel 2; thus, the edge $(C, D) \in E$.

B. Cliques

A clique is represented by an undirected graph where, for each two vertices/nodes in the graph, there exists an edge connecting them; all the edges in the graph use the same

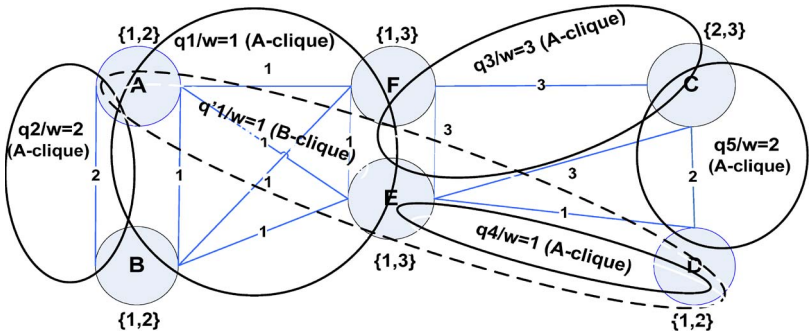

Fig. 3. *A-cliques* and *B-cliques*.

channel. A maximal clique is a clique to which no more vertices can be added.

The formal definitions of clique and maximal clique are given in the following.

Definition "Clique": Let $G = (V, E)$ be an undirected graph, where V is the set of vertices, and $E \subset V \times V$ is the set of edges. A subset $S \subset V$ of vertices is called a clique if, for every pair of vertices in S, there is an edge in E, i.e., the subgraph introduced by S is complete.

Definition "Maximal Clique": A maximal clique S is a clique of which the proper extensions are not cliques i.e., for any S', if $S \subset S'$ and $S \neq S'$, then S' is not a clique.

We use maximal cliques to determine the nodes that compete to access the same channel. Therefore, two nodes i and j that belong to the same clique must not be simultaneously active. Our proposed admission control scheme takes into account both local and neighboring resources. Therefore, we define two types of maximal cliques (see Fig. 3): 1) A-clique is defined as a set of nodes A sharing the same channel and having a pairwise distance smaller than or equal to the minimum radio transmission range of the pair nodes (i.e., $d(u, v) \leq \min(r_u, r_v), \forall\, u, v \in A$), and 2) B-clique is defined as a set of nodes B that use the same channel and have a pairwise distance in the interval $]\min(r_u, r_v), R]\ \forall\, u, v \in B$, where R is the interference range; B-cliques are used to identify nodes in the carrier sense range (CSR).

Fig. 3 shows the maximal cliques of types A and B computed using the connectivity graph shown in Fig. 2. For example, $q1/w = 1$ is an *A-clique* that consists of four nodes $\{A, B, F, E\}$. All of these nodes share the same channel 1. $\{A, D\}/w = 1$ is a *B-clique*, where the distance between nodes A and D is in $]\min(r_A, r_D), R]$, and they share the same channel 1.

Each *A-clique* is represented by its CH. In this paper, we use a simple algorithm that selects a node with the smallest degree as the CH; other algorithms can be used with no changes to our proposed admission-control scheme. CHs are the only nodes involved in the admission-control procedure; more specifically, they are responsible for computing the acceptance ratio, the available bandwidth, the maximum occupancy, and the average service time of *A-cliques* they represent (see details in the next sections). Algorithm I details the election of CHs. In this algorithm, we have three inputs: 1) *A-cliques*; 2) node degree; and 3) MAC address. Here, the degree of node X is denoted as X.degree, which is equal to the number of *A-cliques* to which X belongs. The MAC address is used as a tie breaker when two or more nodes have the same smallest degree.

Algorithm I. CH election algorithm
Input: A-cliques, Node degree, MAC address
Output: Head-of-A-cliques
1 X.state := CH; /* Each node X declares itself as a CH*/
2 Every node X broadcasts a CH_state_notify (X.degree);
/*All nodes in A-clique receive X-degree */
3 Upon receipt of a notification from X, a node Y performs the following:
4 **If** Y.degree == X.degree {
5 Y broadcasts a CH_state_notify (Y.degree, Y.MAC_addr);
6 **If** (Y.MAC_addr > X.MAC_addr) Y.state = ordinary; /* so that not all nodes will be CH */
7 Y.state = CH;
8 } **else if** (Y.degree > X.degree) Y.state = ordinary;

In the rest of this paper, A-clique and CH will be used interchangeably.

C. Assumptions/Notations

The assumptions considered throughout this paper are given as follows: 1) Each A-clique may experience a different load and may have a different capacity than other A-cliques. 2) New flows arriving in each A-clique are uniform, independent, and Poisson distributed. 3) All the information exchanges and the acceptance ratio computation happen only once at the beginning of each control period of length T. 4) The assignment of channels is static. 5) Network failures (link/node) are not considered. To improve the readability of this paper, we gather all the notations in Table I.

D. Queuing Model

The network consists of V routers, Q_A *A-cliques*, and Q_B *B-cliques*. All two-hop neighbors transmitting on the same channel are interfering neighbors. Each node may be either a source or a destination. Assume that the packet size is L (see Section IV-A for the details of our traffic model) and that a client may transfer a packet to its MR as soon as it is generated. Then, the delay between the generation of a packet and its transfer to the MR is negligible. In practice, each MR has a

TABLE I
LIST OF SYMBOLS/PARAMETERS

V	The set of nodes (routers) in the network.
Clique	Every vertex is adjacent to each other.
Maximal clique q_i	The clique which does not belong to any other larger clique: (*A-clique* / *B-clique*).
Q_A	The total number of *A-cliques*.
CH_i	Clique Head of *A-clique* q_i.
C_i	Available bandwidth of *A-clique* q_i.
$N_i(t)$	Number of active flows at time t in *A-clique* q_i.
N_{ct}	Set of *A-cliques* that can contend for channel k utilization, $k \in \{1..NC\}$.
θ_k	Bandwidth of k^{th} channel.
NC	Number of channels per radio; channels are identified/numbered from 1 to NC.
NR	Number of radios per node.
MO_i	Maximum occupancy, i.e., maximum number of flows that can be supported by *A-clique* q_i.
T	Length of the control period.
$PA_i(t)$	Packet arrival rate at time t in *A-clique* q_i.
$Loss_i(t)$	Packet loss probability (PLP) at time t in *A-clique* q_i.
a_i	Flow acceptance ratio in *A-clique* q_i.
R_i	Flow blocking ratio in *A-clique* q_i.
b_i	Average service time of CH_i
D	End-to-end delay.
m_i	New flow request arrival rate in *A-clique* q_i.
$E[X]$	The mean of variable X.
$A_i(t)$	Number of flows that are active in *A-clique* q_i at $X*T$ minus the number of flows that terminate during $]X*T,t]$, where $X*T \le t \prec (X+1)*T$ and X is an Integer.
$New_i(t)$	The sum of the number of flows that have been generated in *A-clique* q_i and the number of flows that started transiting *A-clique* q_i during $]X*T,t]$ where $X*T \le t \prec (X+1)*T$ and X is an Integer.
$Left_i(t)$	Number of flows in *A-clique* q_i that terminate during $]X*T,t]$ where $X*T \le t \prec (X+1)*T$ and X is an Integer.
P_{loss}	PLP threshold.
Δ_{delay}	End-to-end delay threshold.
r_v	Radio transmission range for node v. $\forall u \in V$, if u is in the r_v of node v, u and v are neighbors.
R	Interference range.
CSR	Carrier Sensing Range $[\min(r_u, r_v), R]$,$\forall u, v \in V$ if u is in the CSR of node v, u and v may interfere.
C_neighbors	$\forall u, v \in V$, if u is in the CSR of node v, u and v are C_neighbors.
$D^{(u)}_{\text{degree}}$	Node degree (number of *A-cliques* to which node u belongs).
S_k	The packet generating process of an individual flow k.
$E[S]$	Average flow generation (packets/s)
$P_{q_i q_j}$	The proportion of the traffic generated in *A-clique* q_i that is routed through *A-clique* q_j
$\beta_{q'q}$	The fraction of traffic generated in A-clique q' and going to A-clique q.
$INTER_i$	The waiting time incurred by interferences between *A-clique* q_i and its C_neighbors *A-cliques*.
L	The packet size.
B_{req}	Bandwidth requirement.
Γ_S	The moment generating function of S_k.
Γ_P	The moment generating function of $PA_i(t)$.

finite physical buffer. In our approach, we aggregate all the physical buffers in the *A-clique* into a logical buffer at the CH. The packets are served by the CHs in a first-come–first-serve manner. We propose to model WMNs as a queuing network. The stations/nodes of the queuing network represent the CHs.

Contention Matrix: The contention matrix is equivalent to the connectivity graph (see Section III-A). It shows nodes that are grouped in the same *A-clique*, and only one of these nodes can be active at any given time. Let us define the contention matrix C for all channels as

$$C_q^u = \begin{cases} w, & \text{if node } u \in A{-}clique\ q \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where the dimension of matrix C is $Q_A \times V$.

Let us now define the contention matrix for a number of channels NC as

$$C_q^u = \sum_{w=1}^{NC} wC^w, \qquad w \in \{1, \ldots, NC\} \tag{2}$$

where C^w defines the unit matrix related to a given channel w.

Node Degree: The degree of node u is defined as the number of *A-cliques* that the node u belongs to using the same or different channels. To compute the degree, we have to sum the lines in the matrix represented in (2) for each channel w as

$$for\ u \in V, \qquad D^{(u)}_{\text{degree}} = \sum_{w=1}^{NC} \sum_{i=1}^{Q_A} C^w_{i,u}. \tag{3}$$

Fig. 3 shows that the degree of node E is 3 because E belongs to three *A-cliques*: $q_1/w = 1$, $q_3/w = 3$, and $q_4/w = 1$.

Maximum Occupancy: We assume that the bandwidth requirements of the incoming flows are multiples of F. For example, if F is equal to 100 kb, and the bandwidth requirement B_{req} of flow j is 1 Mb, then B_{req} is $10F$.

The maximum occupancy MO_i of *A-clique* q_i is equal to the number of (unit) flows that can be accepted/supported by q_i, i.e.,

$$MO_i = \frac{C_i}{F} = X \tag{4}$$

where C_i represents the minimum available bandwidth of all nodes belonging to A-clique q_i.

If more than MO_i flows exist in *A-clique* q_i, then q_i is overloaded. In this case, the probability of packet loss is very high. Our proposed scheme, i.e., RCAC, rejects flows when *A-cliques* are overloaded or the PLP is higher than P_{loss}. Thus, the first step in RCAC is to evaluate the maximum occupancy in each *A-clique* so that the bandwidth requirement B_{req} does not exceed the available resources within the *A-clique*. Since each node has different channel views, the maximum occupancy is not simply a local concept. To demonstrate this relationship, we illustrate a scenario with six stations (Y, D, X, D_1, Z, and D_2), as shown in Fig. 4. The MAC layer protocol is IEEE 802.11 with radio transmission ranges of 150 m for Y, 250 m for X, 200 m

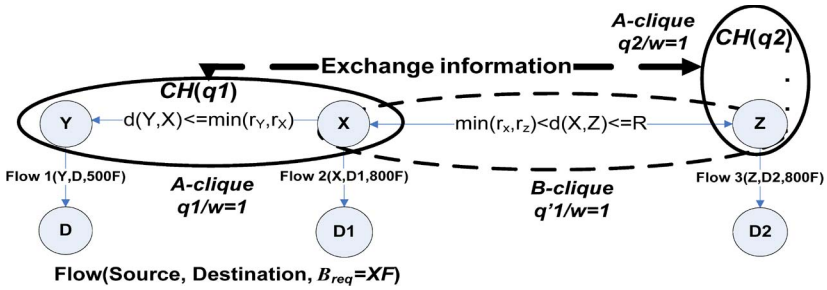

Fig. 4. Traffic scenario. An example.

TABLE II
CHANGES OF LOCAL MAXIMUM OCCUPANCY

At time $t=t_0$, MO in q1,q2	Y (MO_{q1}=2000)	X (MO_{q1}=2000)	Z (MO_{q2}=2000)
Flow1 starts	(MO_{q1}=1500)	(MO_{q1}=1500)	(MO_{q2}=2000)
Flow2 starts	(MO_{q1}=700)	(MO_{q1}=700)	(MO_{q2}=1200)
Flow3 starts	(MO_{q1}=700)	congested	(MO_{q2}=400)

for Z, and $R = 550$ m. The bandwidth of the wireless channel θ_1 is 2 Mb/s. X and Z are C_neighbors. Y is X's neighbor and is out of Z's CSR. Thus, we can conclude that *A-cliques* q_1 and q_2 are C_neighbors.

Table II shows the values of MO of the different cliques, as computed by Y, X, and Z. When flow 1 starts transmitting, only local information (the MO of q_1) is used; however, when flow 2 starts transmitting, not only the local MO is used but the neighboring MO of q_2 as well since 1) X belongs to a *B-clique* and 2) X and Z are C_neighbors; thus, the MOs of both *A-cliques* are reduced. If flow 3 starts transmitting, the congestion will occur (Table II); to avoid this congestion, one has to check the availability of resources in both q_1 and q_2 before accepting flow 3. In our proposed scheme, to compute the maximum occupancy of node i, we use

$$MO_i = \min\left(MO_i(\text{local}), MO_j(\text{local}), \ldots, MO_k(\text{local})\right) \tag{5}$$

where $j, \ldots, k$ are C_neighbors of i.

For the example shown in Fig. 4, the maximum occupancy of Z (after accepting flow 1 and flow 2 in the network) is $MO_i(Z) = \min(1200, 700) = 700$; since flow 3 requires 800, Z will simply reject it when using RCAC.

IV. Proposed Analytical Model: Computing the Clique-Acceptance Ratio and End-to-End Delay

In this section, we first characterize the traffic model based on which the acceptance ratio could easily be computed. With the acceptance ratio, we derive the PLP. Finally, we determine the end-to-end delay for each path. The computed acceptance ratio and the end-to-end delay will be used in the admission-control algorithm (RCAC) to accept or reject new flows into the network; a new flow will be accepted if its requirements in terms of loss rate and delay are satisfied.

A. Traffic Characterization

We denote the packet-generating process of an individual flow k as S_k, and we assume that individual packet-generating processes are independent identically distributed random variables with mean $E[S]$. Thus, the total packet-arrival rate $PA_i(t)$ in an *A-clique* q_i at time t is expressed as

$$PA_i(t) = \sum_{k=1}^{N_i(t)} S_k \tag{6}$$

where $N_i(t)$ denotes the number of active flows at time t in *A-clique* q_i. To characterize $PA_i(t)$ as a Poisson process, we need to specify the mean of $PA_i(t)$. Using the moment-generating functions of random processes $PA_i(t)$ and S_k, we obtain [17, eq. (12)] (see [17] for details). Let Γ_S denote the moment-generating function of S_k, i.e., $\Gamma_S(\theta) = E[e^{\theta S_k}]$, and let Γ_P denote the moment-generating function of $PA_i(t)$. $\Gamma_P(\theta)$ can be computed as

$$\begin{aligned}\Gamma_P(\theta)_{|N_i(t)=N} &= E\left[e^{\theta PA_i(t)} | N_i(t) = N\right] \\ &= E\left[e^{\theta \sum_{k=1}^{N_i(t)} S_k} | N_i(t) = N\right] \\ &= E\left[e^{\theta \sum_{k=1}^{N} S_k}\right] = \{\Gamma_S(\theta)\}^N . \end{aligned} \tag{7}$$

Therefore, we obtain

$$E\left[e^{\theta PA_i(t)} \middle| N_i(t)\right] = \{\Gamma_S(\theta)\}^{N_i(t)} . \tag{8}$$

Applying the mean to (8), we obtain

$$E\left[E\left[e^{\theta PA_i(t)} \middle| N_i(t)\right]\right] = E\left[\{\Gamma_S(\theta)\}^{N_i(t)}\right] . \tag{9}$$

Using

$$\begin{aligned} E[X] &= \sum_x xp(x) = \sum_{xy} xp(x,y) \\ &= \sum_{xy} xp(x/y)p(y) = \sum_y \left(\sum_x xp(x/y)\right) p(y) \\ &= E_y[E_x[X/Y]] \end{aligned} \tag{10}$$

we obtain

$$E\left[e^{\theta PA_i(t)}\right] = E\left[E\left[e^{\theta PA_i(t)}\middle| N_i(t)\right]\right] = E\left[\{\Gamma_S(\theta)\}^{N_i(t)}\right]. \quad (11)$$

Using (8) and (11), we have

$$E\left[PA_i(t)\right] = E\left[N_i(t)\right]E[S]. \quad (12)$$

The number of active flows in *A-clique* q_i at time t can be expressed by the summation of the number of active flows $A_i(t)$ and new flows $new_i(t)$ (see Section III-C) as

$$N_i(t) = A_i(t) + New_i(t). \quad (13)$$

Therefore, to compute $E[PA_i(t)]$, we only need to compute $E[N_i(t)]$ since $E[S]$ is known *a priori* from the source traffic parameters (see the next section for details).

B. PLP

Assuming that the threshold loss probability is sufficiently small, we approximate the PLP by the overflow probability in each *A-clique*. We assume that the WMN is operating in moderate offered-load conditions so that we develop our analysis in the nonsaturated case. This assumption is reasonable since the RCAC objective is to prevent a high offered load from overloading the WMN and degrading the QoS of already-admitted flows. Moreover, most real traffic, either data or VoIP, exhibits ON/OFF behavior and therefore experience idle periods that help the WMN to operate in an unsaturated case but allow longer overflow periods. However, admission control such as our RCAC limits the overflow probability to very small values. On the other hand, it is known that when the overflow probability decays to zero, both measures, i.e., PLP and overflow probability, converge to the same value, and the difference becomes negligible. Therefore, we approximate the PLP, which is denoted by $Loss_i(t)$, by the overflow probability in each *A-clique* as

$$Loss_i(t) = Probability\left\{PA_i(t) \rangle C_i\right\} \quad (14)$$

where $PA_i(t)$ denotes the total (new and cross) packet-arrival rate into *A-clique* q_i at time t. In this paper, we formulate $PA_i(t)$ as a Poisson process. Such type of traffic behavior is expected when the network is accessed by a large number of users.

As mentioned earlier, we approximate the PLP by the overflow in each *A-clique* using (14). With the assumption of Poisson process, $Loss_i(t)$ is expressed as

$$Loss_i(t) = \sum_{k=C_i+1}^{\infty} e^{-\lambda t}\frac{\lambda t^k}{k!} = 1 - \sum_{k=0}^{C_i} e^{-\lambda t}\frac{\lambda t^k}{k!} \quad (15)$$

$$\lambda = E\left[PA_i(t)\right] = E\left[N_i(t)\right]E[S] \quad (16)$$

where the mean number of active flows in *A-clique* q_i at time t is given by $E[N_i(t)] = E[A_i(t)] + E[new_i(t)]]$.

It is worth noting that both incoming (i.e., starting) and outgoing (i.e., terminating) flows are modeled as Poisson processes [see (17)].

In general, the packet service time in carrier sense multiple access/collision avoidance (CSMA/CA)-based IEEE 802.11 is not usually exponentially distributed due to the binary exponential backoff that is performed in random multiples of discrete time slots. However, in this paper, we assume that the packet service time is exponentially distributed for the sake of mathematical tractability. In fact, the authors in [18] demonstrate that the exponential distribution is a good approximation model for the IEEE 802.11 MAC layer service time.

Furthermore, recall that the packet arrival process of each admitted flow by RCAC to a given node is modeled as a Poisson, and RCAC rejects from the entire network all flows that, if accepted, cause ongoing admitted flows to experience significant packet drops. Therefore, since, in the worst case, only very few packets are dropped in a given node, the packet-departure process of each admitted flow by RCAC from each node can be approximated as a Poisson process too, i.e.,

$$E\left[A_i(t)\right] = N_i(x * T) - E\left[left_i(t)\right] \quad (17)$$

where $X * T \leq t \prec (X+1) * T$, and X is an integer.

In the following, we express the traffic generated by local *A-clique* q_i and the transient traffic from adjacent *A-cliques*:

$$E\left[new_i(t)\right] = M_i \times \begin{bmatrix} a_1 m_1 \\ \vdots \\ a_i m_i \\ \vdots \\ a_{Q_A} m_{Q_A} \end{bmatrix} \quad (18)$$

where M_i is the ith row of the following matrix M:

$$M = \begin{bmatrix} 1 & P_{q_1 q_2} & \cdots & P_{q_1 q_{Q_A}} \\ P_{q_2 q_1} & 1 & \cdots & P_{q_2 q_{Q_A}} \\ \vdots & \vdots & \ddots & \vdots \\ P_{q_{Q_A} q_1} & \cdots & \cdots & 1 \end{bmatrix}. \quad (19)$$

The dimension of the matrix M is $Q_A \times Q_A$, and it represents the proportions of the traffic generated in one *A-clique* that is routed through another, where Q_A is the total number of *A-cliques*. The value of a cell M_{ij} is equal to 1 if $i = j$ and equal to $P_{q_i q_j}$ if $i \neq j$, where $P_{q_i q_j}$ consists of the fraction of the traffic generated in *A-clique* q_i that is routed through *A-clique* q_j. To compute $P_{q_i q_j}$, we propose to use a heuristic called Ford–Fulkerson-based matrix computation (FFMC).

FFMC takes as input the amount of traffic generated by each *A-clique*, the capacities of the links connecting the *A-cliques*, source *A-cliques* (i.e., *A-cliques* that generate traffic), and one destination; if there is only one gateway in the network, then the destination is that gateway; otherwise, the destination is a virtual node to which all gateways are connected. First, FFMC executes the (multisource) Ford–Fulkerson algorithm (which computes the maximum flow in a network [19]) on the input to compute the amount of traffic that passes through

each of the links connecting the *A-cliques*. Second, it selects an *A-clique* from the set of *A-cliques*, which receives no traffic from direct neighbors; we make the assumption that there is at least one *A-clique* that will be selected. Then, it computes the row for the selected *A-clique* in the matrix M using the output of the Ford–Fulkerson algorithm; FFMC fills the matrix using normalized values of the output of the algorithm. At the ith step, we subtract the traffic originating from the *A-clique* corresponding to rows that have already been filled, and we apply the same procedure for the rest of the *A-cliques*. Algorithm II presents the pseudocode of FFMC, which has two inputs $G = (S, E)$ and S_{q_i} and one output M, where 1) S represents the set of *A-cliques*, represented by their CHs, and E represents the set of edges between *A-cliques*; 1) the value/cost $e_{q_j q_k}$ associated with an edge connecting two cliques (q_j, q_k) is equal to the bandwidth between them; 2) S_{q_i} is the amount of traffic generated in *A-clique* q_i; and 3) M is the traffic proportion matrix between *A-cliques*.

The variable β in Algorithm II is the fraction of traffic generated in *A-clique* q' and going to *A-clique* q and is expressed as follows:

$$\beta_{q'q} := \frac{e'_{q'q}}{\sum All\ outgoing\ traffic\ from}\ q'. \tag{20}$$

Algorithm II. FFMC: Pseudocode

Input: $G = (S, E)$, S_{q_i}
Output: M
Variables: q_i: *A-clique* from S, C: is a set of *A-cliques*, $G' := (S, E')$, t: real, *Result*: *A-Clique*;

1 $C \coloneqq \varnothing$, $Result \coloneqq Null$, $Mq_iq_j = \begin{cases} 0 & \text{if } ij,\ i \text{ and } j \in \{1..Q_A\} \\ 1, & \text{otherwise} \end{cases}$;
2 G' := Execute_Ford_Fulkerson_algorithm(G);
/* *the value/cost* $e'_{q_jq_k}$ *associated with an edge* (q_j, q_k) (*belonging to* E') *connecting 2 cliques* q_j and q_k *is equal to the fraction of traffic generated in* q_j and *transiting going to* q_k.*/
3 **Repeat**
4 Randomly select q_i from S such that q_i has no traffic coming from its neighbors;
5 $S := S - \{q_i\}$;
6 $C := C \cup \{q_i\}$;
7 **Repeat**
8 Result := Choose an *A-clique* q in $\overline{C}$ such that all of its incoming traffic comes from *A-cliques* in C and $q' \in C$ and there is a flow from q' to q;
/* *If an A-clique is not found*, $Result := Null$ */
9 **if** (Result! = Null) {
10 $P_{q_iq} \coloneqq P_{q_iq} + \beta_{q'q} \times P_{q_iq'}$; (Equation (19))
11 $C := C \cup \{q\}$; }
12 **until** all *A-cliques* are in C or Result = Null
13 **For** every q_j and q_k in G' **do**
14 $e'_{q_jq_k} := e'_{q_jq_k} - S_{q_i} \times P_{q_iq_j} tq_jq_k$;
/* *subtract all the traffic originating from* q_i *from the traffic of the graph* G' */
16 $C := \varnothing$;

C. Computing the Acceptance Ratio

With our traffic model, the acceptance ratio a_i can be computed as follows.

1) Compute the number of active flows in the *A-clique* using (13).
2) Compute $E[PA_i(t)]$ [see (12)] the mean of the packet arrival process in the *A-clique* using (12) with the mean $E[S]$ as *a priori* knowledge (see Section IV-B to compute $E[N_i(t)]$).
3) Compute the PLP $Loss_i(t)$ in each *A-clique* q_i.
4) Use the PLP and the predefined QoS constraint $Loss_i(t) \leq P_{\text{Loss}}$ to compute the acceptance ratio a_i according to (21).

Thus, the acceptance ratio a_i for each *A-clique* q_i can be computed by solving the following equation, where the only unknown parameter is a_i, and P_{Loss} is the PLP threshold:

$$Loss_i(t) = 1 - \sum_{k=0}^{C_i} e^{-(YE[S])t} \frac{(YE[S])\, t^k}{k!} \leq P_{\text{Loss}} \tag{21}$$

where $Y = (N_i(X * T) - E[left_i(t)]) + (M_i \times [a_i m_i]_{1 \leq i \leq Q_A})T$.

D. Computational Complexity of Acceptance Ratio Computing

The added performance and benefits realized by RCAC come at the cost of slightly higher complexity of the FFMC heuristic when computing the proportions of traffic routed between cliques. The complexity of the FFMC algorithm is on the order of $o(n^2)$, where $n = card(V)$ is the number of nodes. In fact, the complexity of computing the acceptance ratios mainly comes from filling the matrix M using Algorithm II (FFMC heuristic).

E. Computing the Delay in Each A-Clique

In this section, we present the delay analysis of the WMN model described in Section III.D. Let N_{ct} denote the number of C_neighbors A-cliques of q_i identified by B-cliques. Before transmitting a packet, each node counts a random timer that is exponentially distributed with mean backoff duration $1/\xi$ [20].

The average service time of q_i is expressed as

$$b_i = \frac{1}{\xi} + \frac{L}{\theta_k} + INTER_i \tag{22}$$

where $INTER_i$ is the waiting time incurred by interferences between *A-clique* q_i and its C_neighbor A-cliques.

In the case of no interferences

$$INTER_i = 0. \tag{23}$$

If interferences exist, then we consider that all interfering *A-cliques* have the same probability to access the medium. In this case

$$INTER_i = \frac{L}{\theta_k} \times \left(\frac{\sum_{j \in N_{ct}} PA_j C_j}{\sum_{j \in (N_{ct} \cup i)} PA_j C_j} \right) \tag{24}$$

where PA_j is the packet arrival rate in A-clique q_i.

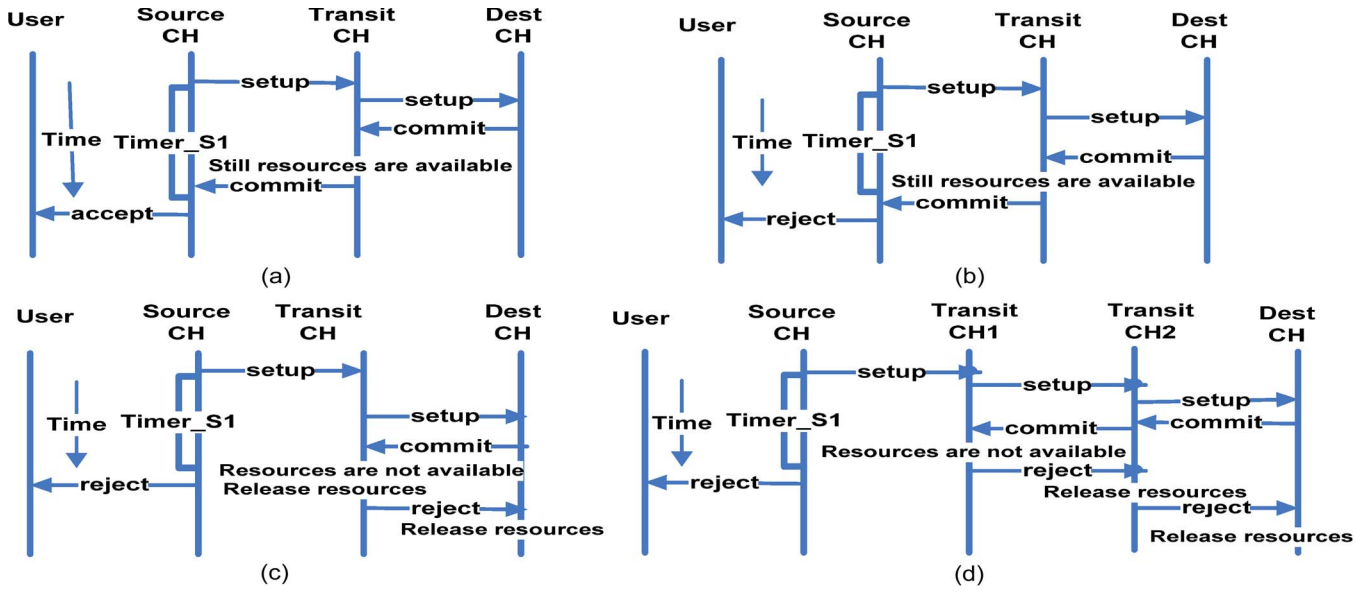

Fig. 5. Inter-CH interactions.

The end-to-end delay for each path is determined by computing the delay at each intermediate CH as

$$D = \sum_{i \in PATH} b_i \tag{25}$$

where $PATH$ is the set of nodes in the end-to-end path from the user to the destination.

The delay parameter is the second QoS metric considered in our model. Before accepting a flow, RCAC checks whether the delay of the selected route is lower than the threshold value (Δ_{delay}), which is predefined according to the requirement of different traffic types. This constraint is expressed as follows:

$$D = \sum_{i \in PATH} b_i < \Delta_{\text{delay}}. \tag{26}$$

V. Routing on Cliques Admission Control Algorithm

In this section, we present the details of the operation of RCAC; more specifically, we describe the operations and interactions among CHs in processing flow requests.

When a CH receives a new flow request from a node (called user) in the *A-clique* (in this case, it is called source CH), it determines 1) whether there is sufficient bandwidth to accommodate the flow (by computing the maximum occupancy [see (5)]) and to satisfy the loss probability threshold (by computing the acceptance ratio [see (21)]) and 2) whether the CH's delay [computed using (25)] is smaller than the delay threshold. If the response is yes, then the source CH sends the request, including the CH's delay, to the next-hop CH toward the destination CH; otherwise, a rejection is sent back to the user. Upon receipt of the request, a transit CH computes the sum of its delay and the delay included in the request; if the accumulated delay is smaller than the delay threshold, then the CH determines whether there is sufficient bandwidth to accommodate the flow and to satisfy the loss threshold. If the response is yes, then it propagates the request, including the accumulated delay to the next-hop CH. Each transit CH repeats this step. When the destination CH receives the request, it carries out a similar process; if the response is yes, then it sends "commit" toward the source CH [see Fig. 5(a)]. A timer is used by the source CH; if the timer expires before receiving the corresponding "commit," then the source CH assumes that the flow request cannot be accommodated and notifies the user [see Fig. 5(b)].

The CHs involved in the flow request setup "conditionally" commit the network resources for the request only after receiving "commit." Indeed, this occurs in the second phase of the setup process, where each CH that receives "commit" forwards it to its previous-hop CH, toward the source CH, if resources are still available to accommodate the request; otherwise, it sends "reject" toward the destination [see Fig. 5(c)]. When the source CH receives "commit," it sends "accept" to the user who can start sending his/her traffic toward the destination.

Upon receipt of "reject," a (transit or destination) CH releases the resources conditionally committed and forwards "reject" to the next hop toward the destination [see Fig. 5(d)].

RCAC, via CHs, supports concurrent processing of setup requests. It realizes concurrence control using an *optimistic* scheme. Indeed, it assumes that it is very unlikely that concurrent flow requests contend for the same resources at the same time, i.e., the success of one request causes failure of the other. This is the reason why, during the first phase, no resources are committed. The optimistic scheme allows CHs to accept more requests than other schemes (e.g., pessimistic scheme) and commit resources only when a request can be accommodated by all CHs involved in the flow request processing. Indeed, the optimistic scheme makes "optimal" the use of the network resources. Let us consider the following example: A transit CH receives a request req_1, decides that it can accommodate the request, commits the corresponding resources, and forward the request to the next-hop CH. Then, it receives a request req_2 (with the same characteristics as in req_1), decides that it cannot accommodate the request, and sends "reject" toward the source CH. Later on, it receives "reject" that corresponds to

req_1 because one of the CHs toward the destination determined that it cannot accommodate req_1. In this case, the two requests are rejected. If the optimistic approach has been used, then req_2 would have been accommodated (assuming that all CHs involved in processing req_2 were able to accommodate the request).

Algorithm III presents the pseudocode of the operation of RCAC in processing a flow request. More specifically, it presents the operation of the source CH, the transit CH, and the destination CH.

Algorithm III. Admission-control algorithm on A-CH

Input: Flow request(*user*, D, B_{req}) in q_j where user is the source node, D is the destination node, B_{req} is the required bandwidth

Output: Admission decision: accept or reject

Variables: timer_S1;

At Source CH:

1 Receive the flow request from user with QoS requirement
2 **if** (Local_Resource_Admission_Test() == accept) {
3 Forward setup request to the next hop CH toward D;
4 Start up timer_S1;
5 **while** (timer_S1 is not expired && commit message not received) **do** {Wait;}
6 **if** (timer_S1 expired) send "reject" message to the user;
7 **else if** (commit received) send "accept" message to the user;
8 } **else** Send "reject" message to the user;

At Transit CH:

9 **if** (Local_Resource_Admission_Test() == $accept$) {
10 Propagate the request with the pending admission control parameters to the next hop CH;
11 **if** (commit received && resources are still available) {
12 Commit "conditionally" resources;
13 Forward "commit" message toward the source CH;
14 } **if** (reject message received) {
15 Release resources;
16 Send "reject" message toward destination CH; }
17 } **else** Send "reject" message toward the source CH;

At Destination CH

18 **if** (Local_Resource_Admission_Test() == accept) {
19 commit "conditionally" resources;
20 send "commit" message toward the source CH;
21 **if** (reject message received) Release resources;
22 } **else** Send "reject" message toward the source CH;

Function: Local_Resource_Admission_Test ()

23 Compute local MO_j (Equation (4));
24 Exchange information with C_neighboring cliques about MO to compute new MO_j (Equation (5));
25 Compute the acceptance ratio a_j by solving (Equation (21));
26 **if** ($N_j(t) < MO_j$ or RANDOM_UNIFORM$(0, 1) > a_j$) return (**reject**);
27 Compute the average service time b_j (Equation (22));
28 **if** (the delay (D) from the source to this q_j is bigger than the threshold value Δ_{delay} (Equation (26))) return (**reject**);
29 return (**accept**);

A CH needs a routing protocol to determine the next-hop CH toward the destination. In our simulations, we used a modified version of destination-sequenced distance-vector routing (DSDV) [21] called MyDSDV as the routing protocol to realize RCAC. MyDSDV is based on delay instead of the distance metric used in the original DSDV. It is used by a source or transit CH that receives a flow request to determine the next-hop CH toward the destination; a CH selects the next-hop CH with the shortest delay. The rationale behind our choice is that DSDV uses the Bellman–Ford algorithm, which is often preferred for mesh networks or sensor static wireless networks. A previous study in [22] shows that the Bellman–Ford algorithm has a lower control message overhead compared with the flooding-based route discovery used in ad hoc on-demand distance vector routing (AODV) [23]. This being said, RCAC can be used with other routing protocols for WMNs.

VI. Simulation Results

In this section, we conduct a simulation study using ns-2 [24] to evaluate and compare the performance of our proposed scheme, i.e., RCAC, with other existing schemes. We evaluate several performance metrics: 1) the end-to-end delay; 2) the throughput; 3) the packet loss and outage probabilities; 4) the flow blocking probability; and 5) the overhead. Note that the outage probability is defined as the ratio of the number of flows experiencing packet losses higher than the given threshold to the total number of accepted flows.

A. Simulation Configurations

In the first scenario (1), the WMN topology used in simulations is arranged as a regular grid of 5×5 IEEE 802.11 stations acting as MRs. More specifically, node spacing is 100 m, and the position of each node from the regular grid is perturbed by choosing a random angle uniformly in $[0, 2\pi]$ and a radius uniformly in [0 m, 25 m]. This perturbation is used in several existing WMN performance studies (e.g., [16]).

In the second scenario (2), the positions of MRs are uniformly distributed in a 1000 m $\times$ 1000 m coverage area. The radio transmission range r takes the value of 150, 200, or 250 m, and the transmission interference R of each wireless station is 550 m.

Real-time traffic flows arrive at each wireless station according to a Poisson process; each flow generates on the average $E[S] = 20$ packets/s. We set a constraint on PLP not exceeding a given threshold P_{loss} and a constraint on delay not exceeding Δ_{delay}; thus, RCAC accepts new flows only if these constraints are satisfied. The other parameters are presented in Table III. It is worth noting that, in the simulation results, we are using WLAN IEEE 802.11 CSMA/CA for RCAC and without RCAC and WLAN IEEE 802.11 MDA for existing schemes in the related work to which we compare.

B. Results Analysis

MAC Access Method Analysis—CA versus MDA Using Scenario (1): As presented in Section II, the IEEE 802.11 draft 2.0 allows an optional contention-free MDA method besides the

TABLE III
SIMULATION PARAMETERS

P_{loss}	5%
Δ_{delay} : scenario (1)/scenario (2)	50 ms/5ms
r	250 meters
R	550 meters
T	5 sec
L: Packet size	1000 bytes
Traffic type	CBR
Rate : scenario (1)/scenario (2)	1 Mbps/10Kbps
NR : scenario (1)/scenario (2)	1/2 radios
NC : scenario (1)/scenario (2)	1/12 channels
WithOut RCAC : W.O.RCAC	a_i =1 , no delay and loss constraints
Topology : scenario (1)/scenario (2)	Grid 5X5 / Uniform-distributed topology
Node spacing : scenario (1)	100 m
Threshold (outage)	5%
Link Tx rate: scenario (1)/scenario (2)	18 Mbps /1Mbps
PHY radio model	SINR-based [16]
Routing protocol for W.O.RCAC	DSDV
DTIM/ DTIM utilization threshold	32 ms/70%
Channel assignment	Static channel assignment [25, 26]
$E[S]$	20 packets/sec
Routing protocol for MDA/Dynamic relocation	AODV
Routing protocol for RCAC	MyDSDV
m_i	20 to 200 flows/ sec
C_i	200 F, F=10 kb/sec
SIFS	10 µs
Backoff Slot time	20 µs
DIFS	50 µs
Data rate	11 Mbps
MAC Header	192 bits
Ack Frame Size	112 bits
PLCP Preamble and Header Length	192 bits
Time slot	20 µs

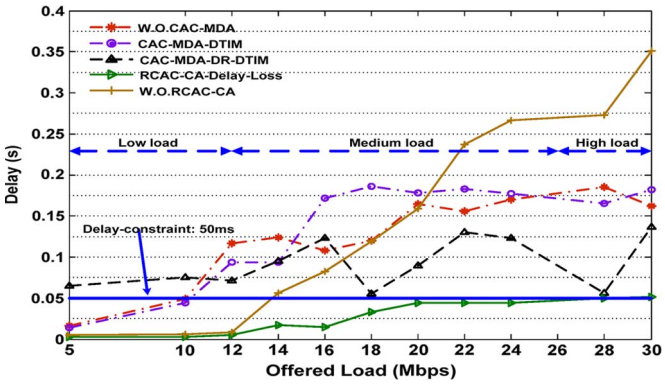

Fig. 6. Delay variation with MDA versus CA.

well-known CSMA/CA contention-based access (CA) method. We studied the performance of both of these access methods when transmitting data on the channel of each clique. In Fig. 6, we present the average end-to-end delay experienced by network flows when using the various simulated schemes; in this section, we analyze the performance of the two MAC access methods without admission control, i.e., WithOut RCAC (W.O.RCAC)-CA and W.O.CAC-MDA schemes in Fig. 6. The average end-to-end delay is the sum of the access delays (both queuing delay, which is negligible in low load, and contention

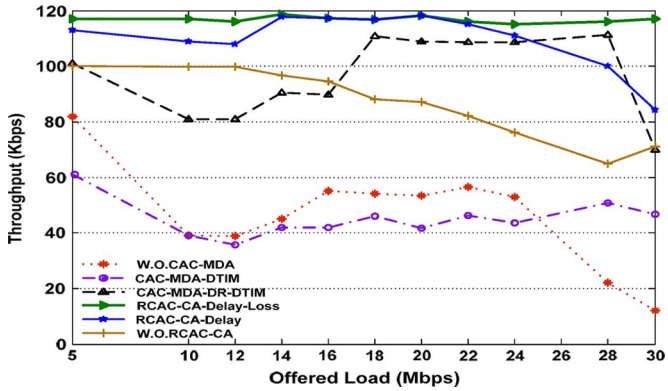

Fig. 7. Throughput variation with MDA versus CA.

delay) experienced in hops along the path's flow from the MC to the gateway.

In a low-offered-load condition, where collisions are very rare, the CA method provides lower delays since it transmits almost instantaneously in a random time slot no later than 0.64 ms, which corresponds to a minimal contention window of CW = 32 time slots of 20 μs. The CW size is initially set to 8, 16, or 32 but is doubled each time a node experiences a collision, until it reaches a maximum value of 1024. However, in this same light-offered-load condition, the MDA method waits longer periods before transmitting in specific reserved contiguous time slots, i.e., particularly MDAOPs, that can be scheduled to start as late as the end of a DTIM period of 32 ms. This scheduling is performed regardless of the absence of interferences and although earlier time slots are available, since it needs contiguous available time slots to transmit packets. Therefore, the average access delay is higher with MDA compared with CA when the offered load is low. On the other hand, in a high-offered-load condition, the end-to-end delay with MDA does not exceed 224 ms; it is bounded by the DTIM interval, which is equal to 32 ms, multiplied by the maximum number of hops in a path that is equal to 7 in our topology. Whereas the delay provided by CA increases without any bounds with the increase of the offered load, it results in many more nodes contending for the same channel, causing many more collisions and resulting in both longer binary exponential backoffs and more frequent MAC retransmissions. Note that binary exponential backoffs and retransmissions are not used by the MDA; this results in much higher packet losses (see Fig. 8) for MDA compared with CA.

In Fig. 7, we compare the average throughput provided by the two access methods. We measure the throughput as the total number of bits that are correctly received for each flow in the WMN in the unit of time. Fig. 7 shows a much lower throughput provided by the MDA compared with CA. MDA throughput degrades much more in a high-offered-load condition. This is explained by the MDAOP reservation mechanism that automatically rejects the newly arrived MDAOP setup requests. This is caused by the contiguous time slots in the DTIM slotted interval that become less available when the offered load increases. This can be verified in Figs. 8 and 9 from the increase in the packet losses and the outage probabilities that are much higher for MDA compared with CA.

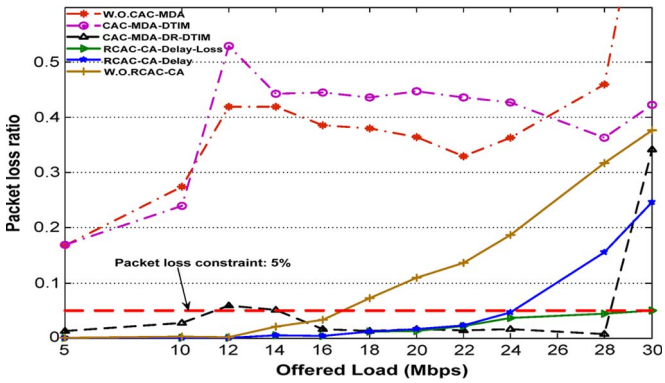

Fig. 8. Loss variation with MDA versus CA.

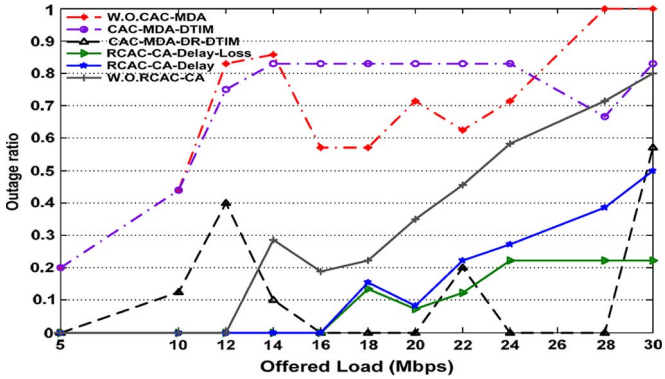

Fig. 9. Outage variation with MDA versus CA.

As opposed to MDA that behaves like an admission control since it bounds the access delay within a fixed DTIM interval, the CA method needs to be enforced with an admission-control mechanism to limit its access delay. This admission control is done by efficiently restricting the number of flows contending for the same channel in a given clique, depending on several parameters. We present the performance of our proposed admission-control scheme RCAC and compare it with other existing schemes in the next section.

Admission-Control Analysis Using Scenario (1): We study the performance of two variants of our proposed RCAC: one based on a delay constraint only, namely, RCAC-CA-Delay, and the other based on both delay and loss constraints, namely, RCAC-CA-Delay-Loss. For the related work, since regular MDA, namely, W.O.CAC-MDA, does not provide a flexible constraint on access delay beyond the DTIM interval, we consider an MDA-based scheme called CAC-MDA-DTIM. This scheme acceptance parameter is based on a constraint on DTIM utilization that does not exceed a given threshold. The DTIM utilization threshold is the maximum fraction of the period of time that can be used by MDAOPs. We present the performance of CAC-MDA-DTIM combined with another MDA improvement that considers the DR [16] of the reserved MDAOPs two hops away; we call this scheme CAC-MDA-DR-DTIM.

In Fig. 6, we notice that the average end-to-end delay experienced by the CA enforced by our proposed admission control RCAC-CA-Delay-Loss is bounded by a much lower delay, i.e., at least three times lower starting from an offered load of 16 Mb/s, compared with that of the MDA with admission

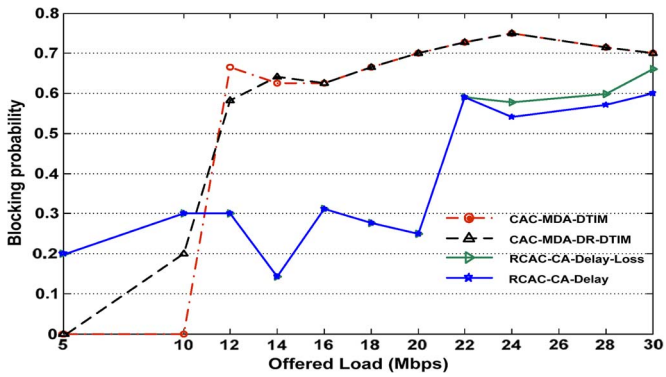

Fig. 10. Blocking variation with MDA versus CA.

control [27]. This is explained by the lower access delay of CA when we limit the number of contending nodes with our RCAC-CA-Delay-Loss compared with that of MDA; the latter generally waits to transmit in specific reserved time slots, even though it is bounded with a DTIM utilization threshold.

In Fig. 8, we observe that MDA-based schemes, either with or without a DTIM utilization constraint, experience much higher packet losses compared with all the other schemes, even in low-offered-load conditions. This seems counterintuitive since MDA is supposed to, by means of MDAOP reservations, provide far fewer collisions/interferences and, therefore, fewer packet losses. However, as presented in [16], the MDAOP reservation process usually suffers from the DTIM fragmentation problem, which can occur even in light-offered-load conditions. In fact, it is usually difficult to find contiguous time slots for reserving long MDAOPs for long packets. Therefore, MDA does not accept MDAOP reservations, and long packets are rejected. In fact, we tested the performance of MDA using small packet sizes of less than 160 B, and we found near-zero packet losses comparable with that provided by the CA. Moreover, interferences outside the two-hop neighborhood are not prevented by MDAOP reservations. These issues are tackled using the DR of interfering MDAOPs; indeed, CAC-MDA-DR-DTIM keeps the packet loss to a near-zero value that becomes significant only after the offered load exceeds 28 Mb/s.

In Fig. 8, we also notice that our proposed schemes RCAC-CA-Delay and RCAC-CA-Delay-Loss provide quite comparable packet losses under low- and medium-offered-load conditions. However, only RCAC-CA-Delay-Loss outperforms all the other schemes by providing a low and bounded packet loss for all offered-load conditions, even in high-offered-load conditions. In Fig. 7, the throughput variation with the offered load is tightly related to the packet loss variation in Fig. 8. Fig. 7 shows that our RCAC-CA-Delay-Loss provides the highest throughput compared with all the other schemes since it limits the number of contending flows by adjusting the acceptance ratio a_i, depending on the offered load, so that the packet loss does not exceed the threshold P_{loss}.

Fig. 10 shows the blocking probability when varying the offered load. Note that the blocking probability is expressed as

$$PR_i = \sum_{i=1}^{Q_A} m_i R_i \Big/ \sum_{i=1}^{Q_A} m_i \tag{27}$$

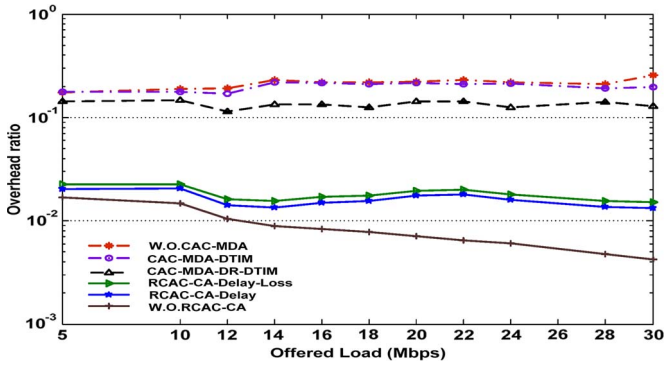

Fig. 11. Overhead variation with MDA versus CA.

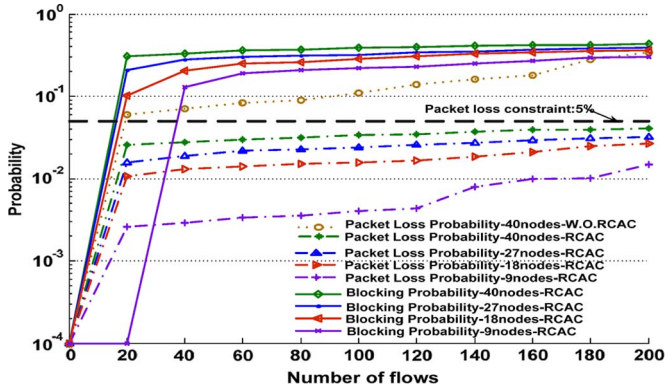

Fig. 12. Blocking PLP of RCAC versus W.O.RCAC for different network sizes.

where $R_i = 1 - a_i$, and m_i is the new flow arrival rate in an *A-clique* q_i.

From Fig. 8, we notice that the packet loss threshold P_{loss} is reached starting from an offered load of 24 Mb/s since the blocking probability in Fig. 10 experienced by RCAC-CA-Delay-Loss becomes greater than that experienced by RCAC-CA-Delay. In fact, to guarantee a packet loss not exceeding the given threshold, our proposed RCAC-CA-Delay-Loss has to block only a little more flows compared with RCAC-CA-Delay.

One can argue that we do not need explicit admission control in WMNs if we use a deterministic access method such as MDA, since the latter guarantees delay bounds required by real-time services. However, our performance study shows that this access method, although it bounds delays, suffers a lot from packet losses and blocking. Some of the MDA improvements presented in the literature may significantly reduce the packet loss, but the blocking probability is still higher than that of our proposed RCAC. In fact, simulation results show that the regular CSMA/CA enforced with our delay-constrained and packet-loss-constrained admission control, i.e., RCAC-CA-Delay-Loss, is able to provide a more efficient utilization of the clique's radio channel since it provides a lower flow blocking probability compared with MDA-based schemes while guaranteeing both delay and packet losses, even in high-offered-load conditions.

Fig. 11 shows the overhead generated by schemes based, respectively, on RCAC-CA and on MDA when varying the offered load. The overhead is computed as

$$Overhead = \frac{\text{Number of control packets}}{\text{Number of control packets} + \text{Number of Data packets}} \tag{28}$$

where the number of control packets for MDA-based schemes includes AODV overhead packets, MDA advertisements, MDA setup requests, etc. For the RCAC schemes, the overhead of MyDSDV includes the DSDV overhead and inter-CH signaling (see Fig. 5). Without RCAC, only the DSDV overhead is considered.

In our case, we observe that RCAC-CA generates a bigger overhead compared with W.O.RCAC-CA. Indeed, the gains of RCAC-CA in terms of delay, throughput, PLP, and blocking come at the cost of a bigger overhead compared with the case without RCAC-CA. However, RCAC-CA still generates a smaller overhead compared with MDA-based schemes.

Admission-Control Scalability Using Scenario (2): Specifically, Fig. 12 shows that, in the case of a nine-node network, RCAC rejects flows starting from the 40th flow, whereas in the case of 18-, 27-, and 40-node networks, RCAC rejects flows starting from the 20th flow. This can be explained by the fact that, in the case of 18, 27, and 40 nodes, interferences are more severe than in the case of nine nodes (we use the same geographic area size and random topologies); furthermore, we use manual (ad hoc) channel assignment. Thus, different results may be achieved using different channel-assignment schemes (e.g., optimal schemes [25], [26]).

Fig. 12 shows the variation of PLP with RCAC and W.O.RCAC while varying the number of flows in a network of 40 nodes. W.O.RCAC corresponds to the basic scheme that always accepts and routes flows from sources to destinations using DSDV [21]. Fig. 12 illustrates that, even when the traffic increases, the PLP does not exceed 5% using RCAC (MyDSDV). However, without RCAC (W.O.RCAC), the PLP threshold is exceeded when the number of flows in the network approaches or exceeds 20. Up to this point, RCAC did not start blocking flows. When RCAC starts blocking flows, the PLP is more than three times higher without RCAC than with RCAC when the number of flows in the network approaches 60. The PLP keeps increasing with the number of flows if RCAC is not used.

Furthermore, Fig. 12 illustrates the PLP variation versus the number of flows in WMN for different network sizes of nine, 18, 27, and 40 nodes. We observe that when both the traffic load and the network size increase, RCAC using MyDSDV maintains a PLP under the 5% threshold. These results confirm our analytical findings: RCAC controls the PLP in the network.

In summary, the simulation results confirm that RCAC can efficiently satisfy the packet loss and the end-to-end delay requirements for real-time traffic in WMNs while providing better utilization of the network resources and not incurring extra overhead (smaller overhead compared with MDA-based schemes). Moreover, we compared the performance results of RCAC using CSMA/CA against an admission control using standard MDA and another admission control using an improved version of MDA [16]. We conclude that our scheme RCAC outperforms existing schemes based on MDA in terms of delay, throughput, packet loss, and blocking probabilities.

VII. Conclusion

In this paper, we have proposed a new distributed admission-control scheme based on cliques, namely, RCAC, to support real-time services in WMNs. Particularly, we have considered PLP and end-to-end delay as two major criteria in the design. Our admission control algorithm is carried out by each CH in parallel; thus, computation efficiency and scalability are achieved. Simulations show that RCAC can effectively prevent the PLP and packet delay in WMNs from exceeding predefined thresholds.

Furthermore, we have concluded from our performance study that the regular CSMA/CA enforced with our delay-constrained and packet-loss-constrained admission control, i.e., RCAC-CA-Delay-Loss, is able to provide a more efficient utilization of the clique's radio channel since it provides a lower flow blocking probability compared with MDA-based schemes while guaranteeing both delay and packet losses, even in high-offered-load conditions.

In a future work, we plan to add more sophisticated admission control for MDA to minimize the blocking probability while satisfying a constraint on packet loss not exceeding a given threshold. Recall that, in our study, we adopted the approach of static channel assignment, which is widely used in the research community. We will investigate the impact/adaptation of RCAC using dynamic channel assignation. Furthermore, we will extend RCAC to handle different types of traffic. Indeed, we will study the scenario of multiple traffic classes with different QoS requirements.

References

[1] P. Kyasanur and N. Vaidya, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE WCNC*, 2005, pp. 2051–2056.

[2] M. Alicherry, R. Bhatia, and L. E. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. ACM Mobicom*, 2005, pp. 58–72.

[3] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. ACM Mobicom*, 2005, pp. 73–87.

[4] Y. Yaling and R. Kravets, "Contention-aware admission control for ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 4, no. 4, pp. 363–377, Jul./Aug. 2005.

[5] L. Lou, M. Gruteser, H. Liu, D. Raychaudhuri, K. Huang, and S. Chen, "A QoS routing and admission control scheme for 802.11 ad hoc networks," in *Proc. ACM Workshop Dependability Issues Wireless Ad Hoc Netw. Sensor Netw.*, 2006, pp. 19–28.

[6] J. Rezgui, A. Hafid, and M. Gendreau, "A distributed admission control scheme for wireless mesh networks," in *Proc. BROADNETS*, 2008, pp. 594–601.

[7] Y. Dong, T. Yang, and D. Makrakis, "SRL-enabled QoS model for mobile ad hoc networks," in *Proc. ICCCAS WESINO EXPO*, 2002, pp. 414–418.

[8] G. Ahn, A. T. Campbell, A. Veres, and L. Sun, "SWAN: Service differentiation in stateless wireless ad hoc networks," in *Proc. IEEE INFOCOM*, 2002, pp. 457–466.

[9] A. Kashyap, S. Ganguly, S. R. Das, and S. Banerjee, "VoIP on wireless meshes: Models, algorithms and evaluation," in *Proc. IEEE INFOCOM*, 2007, pp. 2036–2044.

[10] D. Gu and J. Zhang, "A new measurement-based admission control method for IEEE 802.11 wireless local area networks," in *Proc. IEEE PIMRC*, 2003, pp. 2009–2013.

[11] Y. L. Kuo, C. Lu, E. Wu, and G. Chen, "An admission control strategy for differentiated service in IEEE 802.11," in *Proc. IEEE GLOBECOM*, 2003, pp. 707–712.

[12] Y. Xiao and H. Li, "Local data control and admission control for QoS support in wireless ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 52, no. 5, pp. 1558–1572, Sep. 2004.

[13] Y. Lin and V. W. S. Wong, "An admission control algorithm for multi-hop 802.11e based WLANs," *Comput. Commun.*, vol. 31, no. 14, pp. 3510–3520, Sep. 2008.

[14] G. R. Hiertz, S. Max, Y. Zang, T. Junge, and D. Denteneer, "IEEE 802.11s MAC fundamentals," in *Proc. IEEE MeshTech*, 2007, pp. 1–8.

[15] G. R. Hiertz, S. Max, T. Junge, D. Denteneert, and L. Berlemann, "IEEE 802.11s—Mesh deterministic access," in *Proc. 14th EW*, 2008, pp. 1–8.

[16] C. Cicconetti, L. Lenzini, and E. Mingozzi, "Scheduling and dynamic relocation for IEEE 802.11s mesh deterministic access," in *Proc. IEEE SECON*, 2008, pp. 19–27.

[17] M. Ghaderi, R. Boutaba, and G. W. Kenward, "Joint call and packet QoS in cellular packet networks," *Comput. Netw.*, vol. 51, no. 4, pp. 1060–1071, Mar. 2007.

[18] Z. Hongqiang, K. Younggoo, and F. Yuguang, "Performance analysis of IEEE 802.11 MAC protocols in wireless LANs," *Wirel. Commun. Mobile Comput.*, vol. 4, no. 8, pp. 917–931, Dec. 2004.

[19] [Online]. Available: http://fr.wikipedia.org/wiki/Algorithme_de_Ford-Fulkerson, visited Jan. 15, 2009.

[20] N. Bisnik and A. A. Abouzeid, "Delay and throughput in random access wireless mesh networks," in *Proc. IEEE ICC*, 2006, pp. 403–408.

[21] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *Proc. Conf. Commun. Archit. Protocols Appl.*, London, U.K., Aug. 31–Sep. 2, 1994, pp. 234–244.

[22] Y. Yaling and W. Jun, "Design guidelines for routing metrics in multihop wireless networks," in *Proc. IEEE INFOCOM*, 2008, pp. 1615–1623.

[23] C. E. Perkins, E. M. Belding-Royer, and S. Das, Ad Hoc on demand distance vector (AODV) routing, IETF RFC 3561.

[24] NS-2 simulator. [Online]. Available: http://www.isi.edu/nsnam/ns/

[25] D. Benyamina, A. Hafid, and M. Gendreau, "Wireless mesh network planning: A multi-objective optimization approach," in *Proc. BROADNETS*, 2008, pp. 602–609.

[26] A. Beljadid, A. Hafid, and M. Gendreau, "Optimal design of broadband wireless mesh networks," in *Proc. IEEE GLOBECOM*, 2007, pp. 4840–4845.

[27] M. Soleri, IEEE 802.11s mesh deterministic access: Design and analysis, Pisa, Italy, 2007. [Online]. Available: http://etd.adm.unipi.it/ETD-db/NDLTD-OAI/oai.pl

Jihene Rezgui received the Master's and Engineer's degrees in computer science from the Sciences University of Tunis, Tunis, Tunisia. She is currently working toward the Ph.D. degree with the University of Montréal, Montréal, QC, Canada. The primary focus of her Ph.D. thesis is to define a mechanism that supports quality-of-service management for wireless mesh networks.

She was a Support Engineer with Tunisia Telecom for three years. She is currently with the University of Montréal as a member of the Network Research Laboratory and the Interuniversity Research Center on Enterprise Networks, Logistics, and Transportation.

Abdelhakim Hafid received the Ph.D. degree in computer science from the University of Montréal, Montréal, QC, Canada, in 1996.

He was with Telcordia Technologies (formerly Bell Communication Research), Piscataway, NJ, and the University of Western Ontario, London, ON, Canada. He was a Research Director with the Advance Communication Engineering Center, Canada, a Researcher with the Computer Research Institute of Montréal, Montréal, and a Visiting Scientist with GMD-Fokus, Germany. He is currently a Professor with the Département d'Informatique et de Recherche Opérationnelle, University of Montréal, Montréal, where he founded the Network Research Laboratory in 2005. He has extensive academic and industrial research experience in the area of the management of next-generation networks, including wireless and optical networks, quality-of-service management, distributed multimedia systems, and communication protocols.

Michel Gendreau received the Ph.D. degree in computer science from the University of Montréal, Montréal, QC, Canada, in 1984.

From 1999 to 2007, he was the Director of the Centre for Research on Transportation, University of Montréal, Montréal, which is a center devoted to multidisciplinary research on transportation and telecommunications networks. He is currently a Professor of operations research with the Département de Mathématiques et Génie Industriel, École Polytechnique de Montréal, Montréal,. He has published more than 140 papers on topics related to transportation and telecommunications, as well as coedited six books. His main research interests deal with the development of exact and approximate optimization methods for transportation and telecommunication network planning problems, some of which have been included in commercially available software. He has worked for more than 20 years on telecommunication network design and dimensioning problems, with a focus on the development of specialized optimization algorithms for these classes of problems.

Dr. Gendreau has been the Editor-in-Chief of *Transportation Science* since January 2009.