



Design and implementation of CLASS: A Cross-Layer ASSociation scheme for wireless mesh networks

Yan He^{*}, Dmitri Perkins, Sritej Velaga

Center for Advanced Computer Studies, University of Louisiana at Lafayette, Lafayette, LA 70504, United States

ARTICLE INFO

Article history:

Available online 12 March 2011

Keywords:

Wireless mesh networks
IEEE 802.11
Association
Airtime cost
Network performance

ABSTRACT

This paper focuses on the design and implementation of CLASS, a Cross-Layer Association scheme for IEEE 802.11-based multi-hop wireless mesh networks. The widely-used association strategy in traditional IEEE 802.11 wireless LANs allows a Mobile Station (MS) to scan wireless access links and then associate with the Access Point (AP) that has the best Received Signal Strength Indication (RSSI) value. Unlike traditional wireless LANs, IEEE 802.11-based wireless mesh networks consist of a multi-hop wireless backhaul. As such, the performance experienced by an MS after association with a specific Mesh Access Point (MAP) depends heavily on the conditions of both the access link (e.g., traffic load of associated stations, the frame error rate between an MS and an MAP) and the mesh backhaul (e.g., end-to-end latency and asymmetric uplink/downlink transportation costs). That is, selecting the MAP that yields the “best” performance depends on several factors and cannot be determined solely on the RSSI of the MS-MAP access link. CLASS uses an end-to-end airtime cost metric to determine the MAP to which an MS should associate. The airtime cost metric is based on the IEEE 802.11s, and comprises the access link airtime cost and the backhaul airtime cost. The proposed association scheme considers the frame error rate for various packet sizes, the available bandwidth on the access link after the association of the new MS, and the asymmetric uplink and downlink transportation costs on the backhaul. All experimental results are based on actual Linux-base testbed implementation. We also implement a general Cross-Layer Service Middleware (CLSM) module that is used to monitor network conditions and gather relevant metrics and factor values. Experimental results show that the proposed association scheme is able to identify the MAP which yields the highest end-to-end network performance for the mobile stations after their associations.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

IEEE 802.11-based multi-hop wireless mesh networks (WMNs) comprise of Mesh Routers (MRs) that connect together using wireless radio links to form large-scale wireless broadband networks. Fig. 1 shows the typical architecture of 802.11-based wireless mesh networks. Mesh routers on the mesh backhaul communicate with

each other and connect the client nodes (i.e., Mobile Stations (MSs)) to the mesh network, using 802.11a [1] and/or 802.11b/g [2] techniques. According to the functions of a mesh router, it can be a Mesh Access Point (MAP) providing Access Point (AP) services in addition to mesh services, a Mesh Point Portal (MPP) connecting a mesh network to external networks, or a Mesh Point (MP) only without AP and portal functions [3]. When an MPP connects the mesh network to the Internet, it is also called an Internet GateWay (IGW). Different from the traditional 802.11 WLANs, the backhaul of WMNs is fully wireless.

^{*} Corresponding author. Tel.: +1 337 482 6732.

E-mail addresses: yxh5111@cacs.louisiana.edu (Y. He), perkins@cacs.louisiana.edu (D. Perkins), sxv2719@cacs.louisiana.edu (S. Velaga).

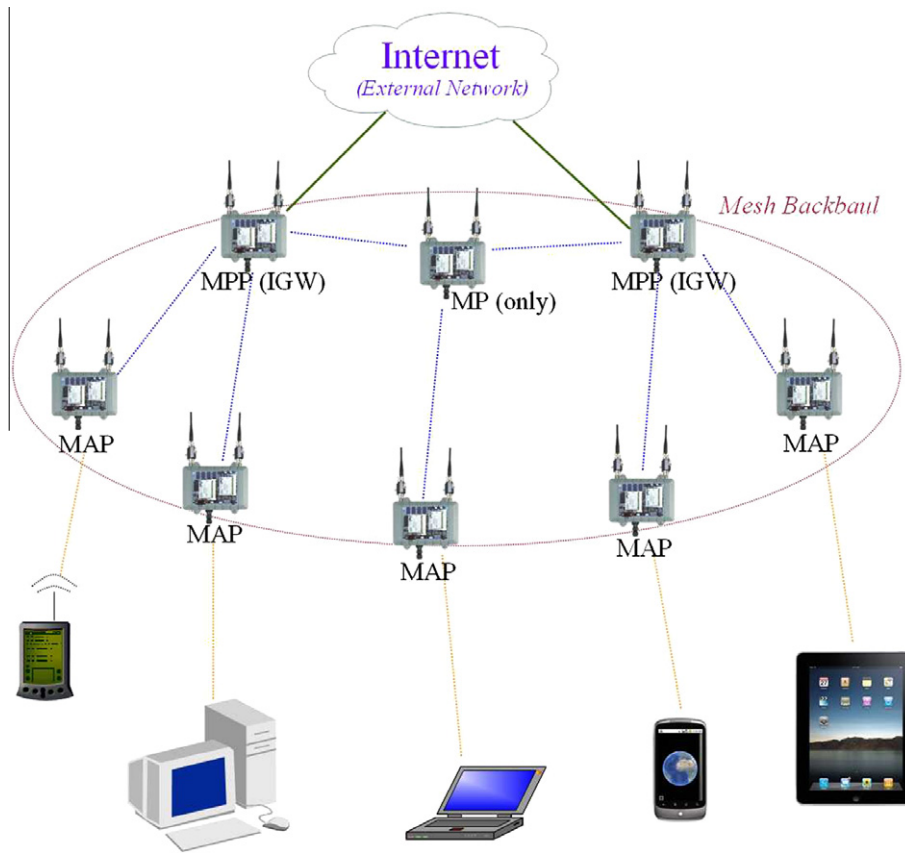


Fig. 1. A typical architecture of 802.11-based wireless mesh networks.

The IEEE 802.11 standard [4] leaves the association strategy open to the implementers. A widely-used association strategy in current implementations is to allow a Mobile Station (MS) to associate with the Access Point (AP) which has the best Received Signal Strength Indication (RSSI) value during its scanning. Previous works [5–8] have revealed that such a strongest signal strength (SSS) [8] based association scheme is unable to provide MSs with the best network performance, and have tried to optimize the AP selection in WLANs based on the access link conditions. Recent research work [9] has argued that the backhaul transportation latency should also be considered for the association in WMNs, and has used the airtime cost defined in the 802.11s draft [3] to evaluate both the access link quality and the backhaul performance for the association. However, the traffic load of the candidate MRs, the effective access link bandwidth after the association of the new MS, the impact of various packet sizes, and the asymmetric backhaul transportation costs on uplink and downlink have not been considered in that backhaul-aware association scheme.

In this work, we propose a Cross-Layer Association scheme for wireless mesh networks, called CLASS. CLASS has two fundamental design and performance goals. The first goal is to identify the MAP which will yield the highest end-to-end throughput performance for a mobile station. The second goal is to achieve a total overall association latency that is tolerable to most user classes. The notion of

tolerable latency is discussed in Section 5.4. The end-to-end airtime cost adapted from 802.11s [3] is used to determine the MAP to which the MS should associate, comprising the access link airtime cost and the backhaul airtime cost. The access link airtime cost is determined by the channel access overhead, protocol overhead, dominant packet size, frame error rate, and the expected available bandwidth after the new MS associates to the MAP. The expected available bandwidth is calculated based on both the current traffic load of an MAP and the expected traffic load generated by the new MS, which infers if the MAP will be saturated after accepting that MS. The backhaul airtime cost is a weighted average of the uplink backhaul airtime cost and the downlink backhaul airtime cost, depending on the application traffic pattern of the MS (e.g., dominant downlink traffic of Video-on-Demand or dominant uplink traffic of video surveillance).

Compared to the previous works [9–13], our proposed association scheme considers (1) the frame error rate and airtime cost for various packet size categories, (2) the impact of the traffic load generated by the new client on the saturation status and available bandwidth of the access link, and (3) the impact of the asymmetric backhaul uplink and downlink transportation costs on the end-to-end performance of the client. Previous works [9,11–13] on association in WMNs are implemented and evaluated in simulations. In our work, the implementation of CLASS is based on the off-the-shelf Wi-Fi devices and the open-

source Madwifi driver [14]. Each node (either an MR or an MS) in the WMN testbed is running a Cross-Layer Service Middleware (CLSM) module implemented in this work, which collects association-related metrics from the modified Madwifi driver and returns them to the association daemon running in the user space. Unlike the previous works, CLASS has no constraint with regard to the routing protocol used on the mesh backhaul, and is independent from the routing metric.

The remainder of this paper is organized as follows. Section 2 discusses the airtime cost introduced in the 802.11s draft and the related work. In Section 3, we describe the proposed association scheme. The implementation of CLASS is described in Section 4, followed by the performance evaluation in Section 5. Finally, we summarize our work in Section 6.

2. Background and related work

2.1. 802.11s airtime cost

Airtime cost is introduced in the 802.11s draft [3] as the cost function for establishing the radio-aware paths by the routing protocol in wireless mesh networks. Airtime cost reflects the amount of channel resources consumed by transmitting a frame over a particular link, and its calculation is designed for ease of implementation and interoperability. Therefore, we adapt the airtime cost in 802.11s to the association metric in our proposed association scheme.

$$c_a = \left[O_{ca} + O_p + \frac{B_t}{r} \right] \frac{1}{1 - e_{pt}}. \quad (1)$$

Eq. (1) shows the calculation of airtime cost. In Eq. (1), O_{ca} and O_p are channel access overhead and protocol overhead respectively, which are constant values for each modulation. B_t is the test frame size in bits, whose value is 8224. Parameter r is the bit rate in Mbit/s at the physical layer, and the frame error rate e_{pt} is the probability that a frame of the standard size (B_t) is corrupted when the frame is transmitted at the current transmission bit rate (r).

2.2. Related work

In the literature, many works have tried to improve the strongest signal strength based association scheme which is widely-used in current 802.11 networks. In the approach proposed by Mhatre and Papagiannaki [5], a client measures the receive signal strength of beacon messages from all the APs operating on the current channel and its overlapping channels, and chooses the AP based on the varying trends of these signals. Vasudevan et al. estimated the potential bandwidth of an AP based on the delay incurred by 802.11 beacon frames from that AP, and used this metric to determine the AP for a client to associate with [6]. Lee et al. also used the estimated available bandwidth as the association metric, which is proportional to the expected number of successfully transmitted data frames at a unit transmission attempt [7]. In the automatic AP discovery and selection system proposed by Nicholson et al., a client scans for

all available unencrypted APs, associates to each of them, and evaluates the quality of each AP's connection [8].

The association schemes in [5–8] are proposed for 802.11-based WLANs. Athanasiou et al. proposed a Cross-Layer Association scheme for wireless mesh networks, which combines channel conditions of the client access channel and the backhaul routing metric of RM (Radio Metric)-AODV [9]. They argued that the backhaul transportation latency should also be considered for the association in WMNs, where the wireless multi-hop backhaul could also become the performance bottleneck of the end-to-end path. Their association scheme used the airtime cost defined in the 802.11s draft [3] to evaluate both the access link quality and the backhaul performance. However, their association scheme does not consider the actual traffic load of a candidate MAP and the impact of various packet sizes on the airtime cost. Their approach relies on the routing metric of the routing protocol, and thus couples the association at the MAC-layer with some particular routing protocol at the network layer.

The association strategy proposed by Giannoulis et al. combines the channel quality measurement and the AP quality score that considers the hop count from an AP to the gateway and the average attained throughput through that AP [10]. However, the channel quality measurement only considers the received signal quality, and the AP quality score does not consider the traffic pattern (e.g., packet size) of the new client. In WCNC 2008, two association schemes for wireless mesh networks were proposed [11,12]. These two schemes both considered the access link quality and the backhaul transportation cost. One of them found [11] that the impact of packet size on the total association cost is significant, and the other [12] argued that the traffic load should be considered for the evaluation of the access link quality. In [13], Wang et al. combines the expected transmission time of a test frame on the access link and the 802.11s airtime cost based multi-hop backhaul association cost in their association scheme. However, these three schemes [11–13] did not consider the frame error rate for different packet sizes, and did not consider the impact of the traffic load generated by the new client on the saturation status and available bandwidth of the access link.

Our proposed association scheme considers the frame error rate and airtime cost for various packet size categories, and the potential available bandwidth after the client association based on the predicted traffic load. The proposed scheme uses the airtime cost to measure the backhaul performance of a candidate MR. Thus, it has no constraint on the routing protocol running on the backhaul, and is independent from the routing metric. The backhaul performance measurement in our scheme considers the uplink and the downlink separately to support the applications which have significant asymmetric uplink and downlink transportation costs.

3. The Cross-Layer Association scheme

In this section, we first discuss the association metric used by CLASS. Then we present the association procedure in CLASS.

3.1. Association metric

$$TC_{i,a} = (1 - \alpha)AC_{i,a} + \alpha BC_a. \quad (2)$$

The association metric in CLASS is based on the airtime cost introduced in the 802.11s draft [3], which reflects the average latency for the transmission of a unit size packet on a particular link. However, the association metric in CLASS is the end-to-end total airtime cost ($TC_{i,a}$) from MS i to the IGW via MR a (i.e., the whole wireless path), and its calculation considers the actual packet size of the traffic, the corresponding frame error rate for that packet size, and the potential available bandwidth on the access link after the client association. As shown in Eq. (2), $TC_{i,a}$ includes the access link airtime cost $AC_{i,a}$ between MS i and MR a , and the backhaul airtime cost BC_a between MR a and the IGW it connects to. The tunable parameter α ($0 \leq \alpha \leq 1$) weighs the influence of $AC_{i,a}$ and BC_a . The value of α is static for a particular deployment scenario. To determine the α value that maximizes the client performance after an association, we can develop the empirical model for α using the statistical design of experiments (DOE) [15] methodology, which predicts the optimal α values within the design space.

After MS i gets the end-to-end airtime cost from each candidate MR, it will associate to MR a if $TC_{i,a}$ is the lowest among the airtime costs of all candidate MRs.

3.1.1. Access link airtime cost calculation

$$AC_{i,a} = \left(O_{ca} + O_p + \frac{B_i}{R_{avl}^i} \right) \frac{1}{1 - e_{pt}^i}, \quad (3)$$

$$e_{pt}^i = 1 - \frac{N_{original} - N_{dropped}}{N_{original} + N_{retries}}, \quad (4)$$

$$R_{avl}^i = \begin{cases} \lambda_a R_{i,a} & \lambda_a \geq \lambda'_a \\ \frac{1}{\sum_{j \in C_a \cup \{i\}} R_{j,a}} & \lambda_a < \lambda'_a \end{cases}. \quad (5)$$

Eq. (3) shows the access link airtime cost between MS i and MR a , where e_{pt}^i is the frame error rate between MS i and MR a , R_{avl}^i is the available bandwidth that MS i can acquire from MR a , and B_i is the dominant packet size in bits of the expected traffic generated by MS i or the recorded average packet size in bits of the traffic generated by MS i before. When the application running on the MS has a certain dominant packet size (e.g., the MS is transmitting the real-time video captured by its video camera using the H.264 encoding), CLASS will use this dominant packet size for B_i . If the dominant packet size is unknown, CLASS will then set B_i to the average packet size in bits of the traffic generated by the MS before, which was recorded by our modified driver on the MS. If this information is also unavailable, the constant testing frame size (8224-bit) defined in the 802.11s draft [3] will be used for B_i . O_{ca} and O_p in Eq. (3) are channel access overhead and protocol overhead respectively, which are constant values defined in the 802.11s draft [3]. The calculation of e_{pt}^i is shown in Eq. (4), where $N_{original}$ is the number of original unicast data packets transmitted by MS i , $N_{retries}$ is the number of data

packet retries, and $N_{dropped}$ is the number of unicast data packets dropped.

To calculate the expected available bandwidth R_{avl}^i , we need to measure the current traffic load on the client access channel and determine if the channel will be saturated after the association of MS i . In CLASS, each MR monitors the transmission time and the receive time during a time duration T . When MS i calculates R_{avl}^i , it can get the channel idleness ratio from MR a as follows, where t_j and t_k are the transmission time (including backoff, retransmission, etc.) of Packet j and the receive time of Packet k .

$$\lambda_a = \frac{T - \left(\sum_{j=1}^{N_{sent}} t_j + \sum_{k=1}^{N_{received}} t_k \right)}{T}. \quad (6)$$

When $AC_{i,a}$ is bigger than the expected latency of applications running on MS i (as shown in Inequality (7) where $R_{i,a}$ is the data rate at the physical layer between MS i and MR a , and R_{req}^i is the bandwidth requirement of applications on MS i), the access link of MR a is saturated. Previous research has shown that under saturation the associated MSs of an MR fairly share the bandwidth with an upper bound shown in Eq. (5), where C_a is the set of existing clients of MR a [16]. By comparing λ_a of MR a with λ'_a (in Eq. (8)), MS i can determine if the access link of MR a will be saturated after its association and estimate the available bandwidth using Eq. (5).

$$\left(O_{ca} + O_p + \frac{B_i}{\lambda_a R_{i,a}} \right) \frac{1}{1 - e_{pt}^i} \geq \frac{B_i}{R_{req}^i}, \quad (7)$$

$$\lambda'_a = \frac{B_i}{R_{i,a} \left[(1 - e_{pt}^i) \frac{B_i}{R_{req}^i} - (O_{ca} + O_p) \right]}. \quad (8)$$

3.1.2. Backhaul airtime cost calculation

Eq. (9) shows the backhaul airtime cost of an MR measured in CLASS, which is a weighted average of the uplink backhaul airtime cost (in Eq. (10)) and the downlink backhaul airtime cost (in Eq. (11)) between that MR and the IGW. The weight β ($0 \leq \beta \leq 1$) is determined by the application traffic pattern of the MS. When the traffic pattern is unknown, β is then set to 0.5. If the dominant traffic of the applications running on the MS is downlink (e.g., Video-on-Demand), β is assigned with a higher value than 0.5; if the dominant traffic is uplink (e.g., video surveillance), β is assigned with a lower value than 0.5 instead. In Eqs. (10) and (11), R_k is the data rate of Link k in the backhaul path, e_{pt}^k is the frame error rate of that link, and H_{uplink} and $H_{downlink}$ are the hop counts of the uplink path and the downlink path, respectively. In CLASS, the frame error rate of each backhaul link is tracked by the modified driver for various packet size categories. In our current implementation, the packet size has three categories: small packets are shorter than 250 bytes; large packets are longer than 1100 bytes; and medium packets have the size between 250 bytes and 1100 bytes. CLASS is independent from the routing metric, and in some routing protocols the uplink routing path may be different from the downlink one.

$$BC_a = (1 - \beta)BC_{uplink}^a + \beta BC_{downlink}^a, \quad (9)$$

$$BC_{uplink}^a = \sum_{k=1}^{H_{uplink}} \left(O_{ca} + O_p + \frac{B_i}{R_k} \right) \frac{1}{1 - e_{pt}^k}, \quad (10)$$

$$BC_{downlink}^a = \sum_{k=1}^{H_{downlink}} \left(O_{ca} + O_p + \frac{B_i}{R_k} \right) \frac{1}{1 - e_{pt}^k}. \quad (11)$$

3.2. Association procedure

Figs. 2 and 3 show the overall association procedure in CLASS over the network and the detailed association steps on an MS, respectively. When an MS attempts to join in the wireless mesh network through an MAP, it will scan all AP channels one by one to discover the MAPs running on each channel. If the received signal quality of the beacon message from an MAP is higher than a threshold θ (set as 15 dB for the Atheros 5212 chipset in our experiments in terms of SNR [17,18]), the MS will send this candidate MAP an Association Information Request (AIR) message, which requests the necessary information for both access link and backhaul airtime cost calculation.

When a non-IGW MAP (such as MR_n in Fig. 2) receives an AIR message from the MS, it will send the IGW (maintained in its routing table) a Backhaul Airtime Request (BAR) message attached with its own uplink backhaul airtime cost (for the link to the next hop). The header of a BAR message has a direction bit that indicates whether the message is transmitted on the uplink or the downlink. The destined IGW will send back the initiating MAP that BAR message with the accumulated uplink backhaul airtime cost and its downlink backhaul airtime cost. Any intermediate MR between the initiating MAP and the IGW will update the uplink or the downlink backhaul airtime cost contained in the BAR message according to the direction bit in the message header, when that message passes it. After the initiating MAP receives the BAR message initiated by itself, it will send an Association Information resPonse (AIP) message to the MS, which contains the metrics of its access link, and the separate uplink and downlink backhaul airtime costs. If the MAP contacted by the MS is also an IGW (such as MR_{n+2} in Fig. 2), the aforementioned backhaul process for CLASS is exempted, and the uplink and downlink backhaul airtime costs in the AIP message are both zero.

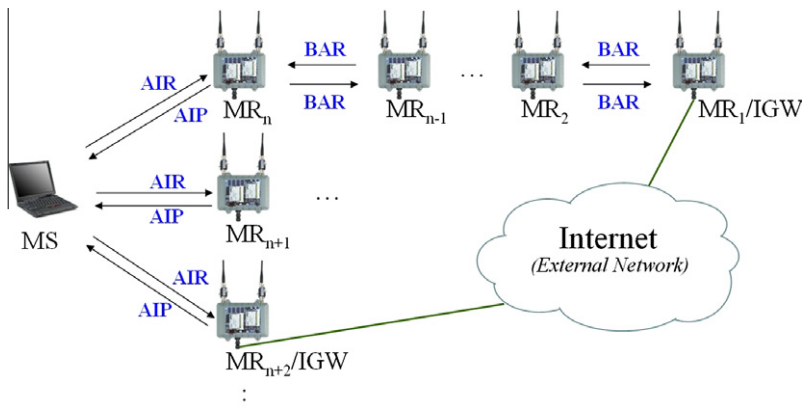


Fig. 2. Procedure of the CLASS association.

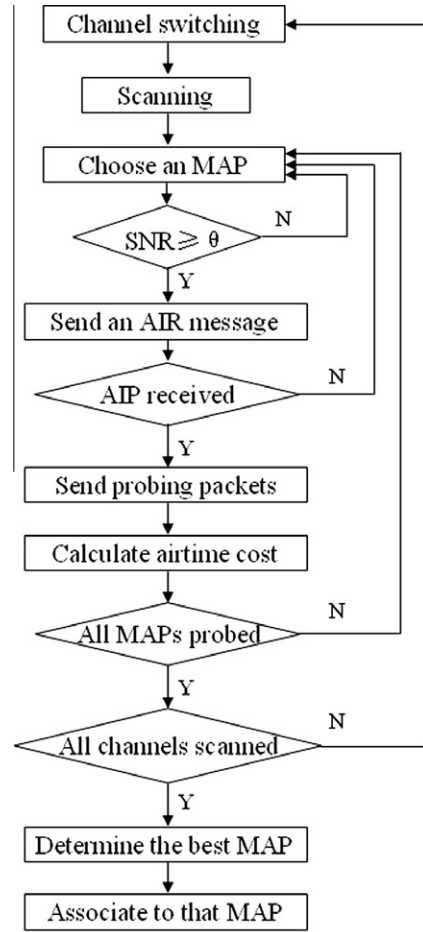


Fig. 3. Association steps of an MS.

If the MS has not received any AIP message from an MAP after several tries, it will ignore that one and turn to the next candidate. Otherwise, the MS will transmit numerous probing data packets with the size of B_i to that MAP, which are used to measure the frame error rate between the MS and the MAP. Based on this probing result and the metrics in the AIP message, the MS is able to calculate the end-to-end airtime cost which is via that MAP.

After measuring the airtime costs for all candidate MAPs, the MS will associate to the one through which the end-to-end airtime cost is the lowest. In our measurements, it took less than 1 second for the MS to acquire the metrics, probe the AP channel and calculate the airtime cost for each candidate MAP. Thus, the total association latency is tolerable to the users, even if the number of candidate MAPs is up to 10.

3.3. Reassociation support in CLASS

In the design of CLASS, the association is performed when a mobile station joins in a WMN. If the current connection is broken or the SNR of received signal strength is lower than 10 dB [18], CLASS will initiate the reassociation for the MS automatically. However, in CLASS, the reassociation will not be initiated automatically to seek better end-to-end performance for the MS when the current connection is stable. The main reason to avoid aggressive reassociations is to maintain the stability of the network and minimize the overhead induced by reassociations. The previous work in [13] shows that the frequency of automatical reassociations is very high and the performance of mobile stations may become worse if the reassociation threshold is not well chosen for the specific network configuration and traffic pattern. In the simulations in [13], 30 MAPs are randomly distributed in an area of $1000\text{ m} \times 1000\text{ m}$, and only 8 flows are presented over this mesh network. The average number of reassociations is 170 per station during 1000 s without the reassociation threshold, and is still as high as 50 per station when the reassociation threshold is tuned to achieve the best performance of mobile stations for this network configuration. When the number of active clients increases (e.g., each MAP has an active client), the overhead of aggressive reassociations may increase rapidly, and the network status may have trouble in converging. In addition, the personality of users has a large variety. Some users are satisfied with their current network performance and are unwilling to reassociate their devices to the new MAPs for a little performance enhancement, since the performance of the real-time applications using on their devices may be corrupted during the reassociation. Other users may be aggressive to improve the network performance as high as possible. In CLASS, the reassociation choice is left to users, when the connection between the MS and its current MR is still stable. Users can initiate the reassociation by CLASS if they are not satisfied with the current network performance.

4. Implementation details

We have implemented a prototype version of CLASS for Linux. We modified the Madwifi driver [14] to support CLASS functions on the MS and the software-based MR. The architecture of CLASS on the MR (as shown in Fig. 4) and that on the MS (as shown in Fig. 5) both include the modified driver running in the kernel space, the daemon running in the user space, and a Cross-Layer Service Middleware. The CLSM module provides APIs to the CLASS daemon on either the MR or the MS. The daemon is able to get

the association-related metrics from CLSM through its APIs, instead of the APIs of a specific driver. CLSM will get the required metrics from the driver for the specified wireless interface. In the following, we will elaborate on the implementation details of CLASS.

4.1. The Cross-Layer Service Middleware (CLSM)

The Cross-Layer Service Middleware (CLSM) is a general platform that provides the applications running in the user space with the data from the lower layers in the networking stack and the ability to change the configurations of lower-layer protocols. The motivation to build such a platform is to facilitate the data request of applications and hide the implementation details of the lower layers. For instance, the MAC-layer protocol of an 802.11-based wireless network card can be maintained by the Madwifi driver, HostAP driver [19] or other driver depending on the chip-set of that card. Applications running in the user space such as the CLASS daemon need not know any information of the specific driver. Instead, they can get the information of the MAC and physical layers from CLSM, which is aware of the corresponding APIs and the parameter formats provided by various drivers.

CLSM accesses the networking protocols/modules in the kernel space using `ioctl` calls of Linux, and provides the applications with two types of interfaces: local APIs are available to local applications; a listening server on TCP/IP sockets is waiting for the requests from the applications running on remote machines (or local applications). Upon the data request of applications, CLSM will return the metrics acquired from the lower layers (e.g., the number of original unicast data packets transmitted, the number of data packet retries, and the number of unicast data packets dropped), and/or calculate some comprehensive metrics (e.g., frame error rate) based on those metrics. CLSM can also adjust the configurable parameters of networking protocols (such as TCP buffer size and TCP timestamp support) according to the request of applications.

4.2. Implementation of CLASS on the MR

As shown in Fig. 4, the modified Madwifi driver on the MR is responsible for tracking the access link conditions (e.g., packet transmission time during each time slot) and the backhaul conditions (e.g., the number of data packet retries of each packet size category for each link during each time slot). By default, the modified driver records the data of our customized metrics for last 20 s, in which each time slot is 1 s. That is, the CLASS daemon is able to get the statistical information for CLASS for up to 20 s, and the granularity of the time duration is 1 s.

The driver also checks each received unicast data packet to identify the Association Information Request (AIR) message from an MS or the Backhaul Airtime Request (BAR) message from its neighboring MRs. The driver will extract the information from those messages, signal the CLASS daemon to handle the received request, and deliver data to the daemon through a raw socket. An intermediate MR between the initiating MR of the BAR message and the IGW will discard the received BAR message after extracting

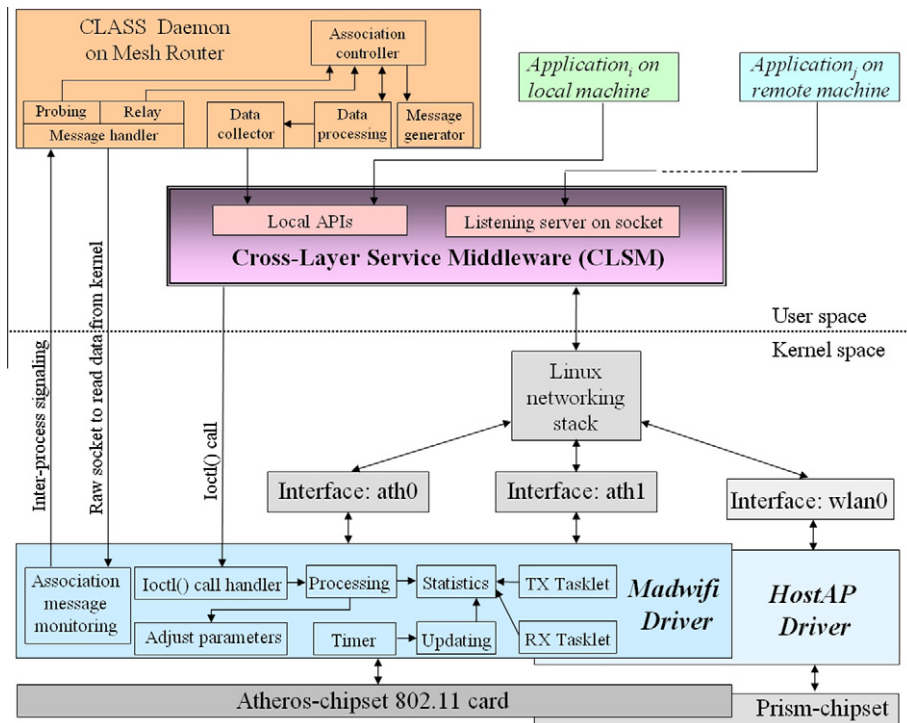


Fig. 4. Architecture of the CLASS implementation on MR.

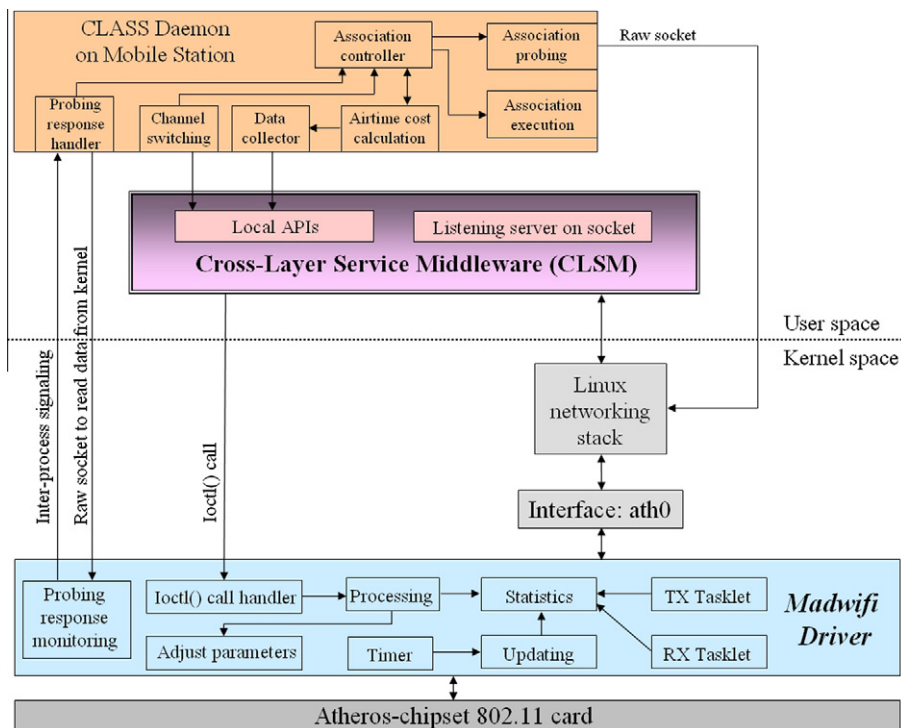


Fig. 5. Architecture of the CLASS implementation on MS.

its data (in both header and payload), and generate a new BAR message for its next hop with updated uplink or downlink backhaul airtime cost but the same source and

destination IP addresses. Both the BAR message and the Association Information resPonse (AIP) message of the MR are generated and transmitted using raw socket.

4.3. Implementation of CLASS on the MS

The Madwifi driver on the MS is running in the Monitor Mode during the MAP discovery and AP channel probing. The CLASS daemon generates the AIR message and the probing data packets (for measurement of e_{pr}) using raw socket so that they can be transmitted in the Monitor Mode. The probing data packets will be acknowledged by the MR in case of successful reception, although they will not be delivered to the upper layer since the MS is not associated. As long as the operating mode is the Monitor Mode, the modified Madwifi driver checks each received unicast data packet to identify the AIP message from the MR. Upon receiving the AIP message, the driver will signal the CLASS daemon to fetch the information extracted from that message via a raw socket. After the MS identifies the best MR for its association, it will switch the Madwifi driver into the Managed Mode for the standard 802.11 association messaging and the subsequent regular data communications. The modified driver records the packet size of each transmitted packet, and updates the average packet size of the traffic periodically, which is useful to future association by CLASS.

5. Performance evaluation

In this section, we compare the performance of CLASS with the strongest signal strength based association scheme (denoted as “SSS” here) and the backhaul-aware association scheme proposed in [9] (denoted as “Ath07” here), in terms of the end-to-end performance of the MS via the MR selected by each association scheme. In our experiments, the weights of the access link airtime cost and the backhaul airtime cost are set to be equal in the calculation of the end-to-end airtime cost, which excludes the impact of weights on the performance comparison between CLASS and Ath07. We also evaluate the total association latency by CLASS as well as the delay contributed by each step in the CLASS association procedure as outlined in Fig. 3.

5.1. Experiment 1: Investigating the impact of packet size on airtime cost

The objective of this experiment is to investigate the impact of packet size on airtime cost and the selection of the right MR which provides the MS with the best end-to-end performance. Fig. 6a shows the testbed used in this experiment. In the testbed, the three MRs (MR1, MR2 and MR3) use 802.11a for the backhaul and the three non-overlapping channels (Channels 1, 6 and 11) of 802.11g for their client access channels, respectively. MR3 is also an IGW, and is connected to a server on the Internet via ethernet. An existing client node associated with MR1 generates the background traffic for the experiment, which is 10 Mbps UDP flow from the client node to the server with the 8224-bit packet size. The AP channel data rate of MR3 is set as 9 Mbps in this experiment, in order to increase the access link airtime cost via it. The packet size of the probing packets varies from 100 bytes to 1400 bytes

as well as the packet size of the UDP flow generated by the new MS. The transmission powers of MR1, MR2 and MR3 are tuned in all experiments of this section so that the received signal strength of MR1 at the new MS is always the highest among all MRs.

In the experiment, the received signal strengths of MR1, MR2 and MR3 at the MS are -31 dBm, -40 dBm and -48 dBm (the experimental results shown in this section are the average values of five replicates), and the noise level is -95 dBm on their client access channels in our measurements. Thus, SSS selects MR1 for the MS all the time. Ath07 only considers the constant testing frame size (8224-bit) in the calculation of airtime cost, and the airtime costs calculated by Ath07 for MR1, MR2 and MR3 are $1446 \mu\text{s}$, $594 \mu\text{s}$ and $554 \mu\text{s}$, respectively. Thus, MR3 is selected by Ath07 all the time.

According to our measurement, the dominant data rate of the AP channel of MR2 is 48 Mbps and the dominant backhaul channel data rate between MR2 and MR3 is also 48 Mbps in the experiment. Fig. 6b shows the total (end-to-end) airtime cost of the MS via each MR, measured by CLASS for various packet sizes. The airtime cost via MR1 is always the highest, since the access link of MR1 is under high traffic load and it has the biggest hop count from the IGW (MR3). When the packet size is relatively small (i.e., 100 bytes and 250 bytes), the airtime cost via MR3 is the lowest. The reason is that the advantage of 0 backhaul airtime cost (directly connected to the Internet) overwhelms the disadvantage of a lower AP channel data rate (as shown in Fig. 6c). When the packet size raises to 500 bytes and above, the advantage of a higher AP channel data rate overwhelms the disadvantage of higher backhaul airtime cost. Hence, the total airtime cost via MR2 is the lowest for the medium and large packet sizes (i.e., 500 bytes, 1000 bytes and 1400 bytes).

Fig. 6d shows the throughput of the UDP session between the MS and the server on the Internet measured by the tool of Iperf [20]. From Fig. 6d, we can see that MR1 selected by SSS always provides the lowest throughput to the MS. MR3 selected by Ath07 provides the MS with the highest throughput for the small packet size, but does not for the medium and large packet sizes. CLASS considers the impact of packet size on the airtime cost. Thus, the MR selected by CLASS for each packet size provides the highest throughput to the MS in our investigation.

5.2. Experiment 2: Investigating the impact of traffic load on airtime cost

In this experiment, we investigate how the traffic load on the access link of an MR impacts on the airtime cost of the MS. In the testbed shown in Fig. 7a, MR1 has two client nodes which have a 10 Mbps UDP flow between them; MR3 has a client node which has a 20 Mbps UDP flow to the server; MR2 also has a client node associated but its traffic load is negligible. In this experiment and Experiment 3 in the next subsection, the packet size for both the background traffic and the traffic of the new MS is 8224 bits, in order to exclude the packet size impact which differentiates the performance of Ath07 and CLASS. The

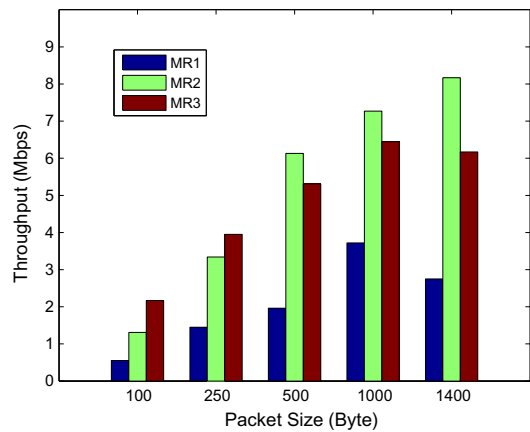
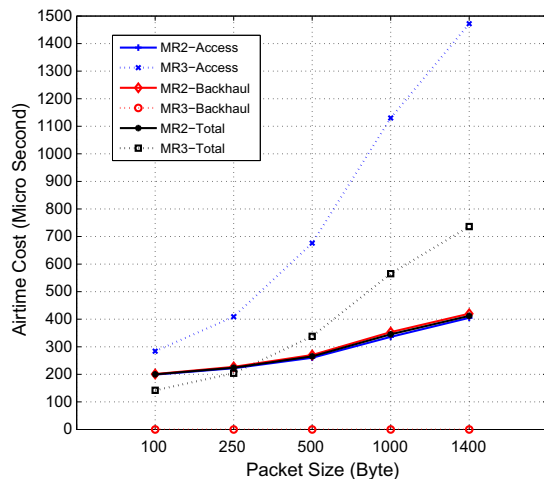
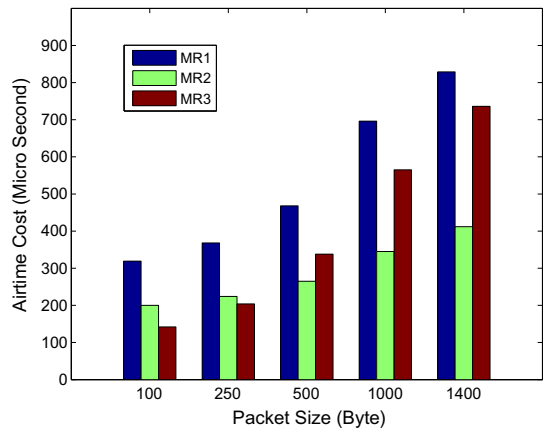
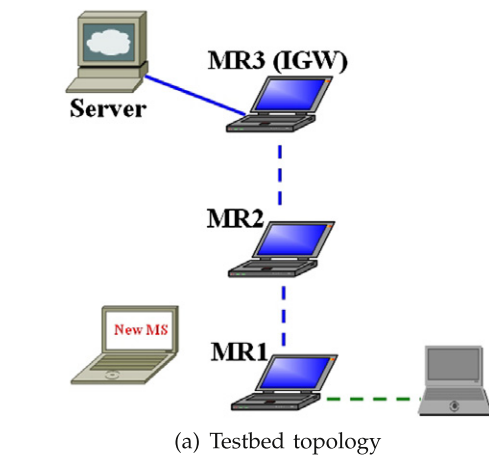


Fig. 6. Experiment 1: Investigate the impact of packet size on airtime cost.

data rates of all clients, the AP channel and the backhaul channel of each MR are set into Auto Mode.

Fig. 7b shows the access link airtime cost, the backhaul airtime cost, and the total airtime cost of Ath07 and CLASS via various MRs. The airtime cost of Ath07 and the airtime cost of CLASS cannot be compared with each other since they are calculated in different ways. The measurement of the access link airtime cost in Ath07 does not consider the actual traffic load of an MR. Thus, MR3 is selected by Ath07 since its backhaul airtime cost is zero, which results in the lowest total airtime cost among the three MRs. CLASS takes into account the traffic load on the access link, and thus determines that MR2 is the best choice for the MS to associate with. Fig. 7c shows the throughput of the UDP session between the MS and the server when it associates with various MRs. The experimental result shows that the MR selected by CLASS provides the MS with higher throughput than the MRs selected by other schemes, indicating the traffic load consideration in CLASS is significant to the correct association decision for the MS.

5.3. Experiment 3: Investigating the impact of asymmetric uplink and downlink backhaul transportation costs on airtime cost

The objective of this experiment is to investigate the impact of asymmetric uplink and downlink backhaul transportation costs on airtime cost. Fig. 8a shows the topology of the testbed used in this experiment. The WMN testbed here comprises of 6 MRs. MRs 1, 2, 4 and 5 are MAPs, and MR3 and MR6 are MP only. The backhaul of MRs 1, 2 and 3 is running on Channel 40 of 802.11a, and the backhaul of MRs 4, 5 and 6 is running on Channel 60. Thus, two non-overlapping backhaul branches are connected to the Internet for this WMN. MR1 has a client node which has 10 Mbps uploading UDP traffic to the server, and MR4 has a client node which has 10 Mbps downloading UDP traffic from the server. The traffic pattern of the new MS is the downloading UDP traffic from the server, such as the Video-on-Demand traffic.

Fig. 8b shows the access link airtime cost, the backhaul airtime cost, and the total airtime cost of Ath07 and CLASS

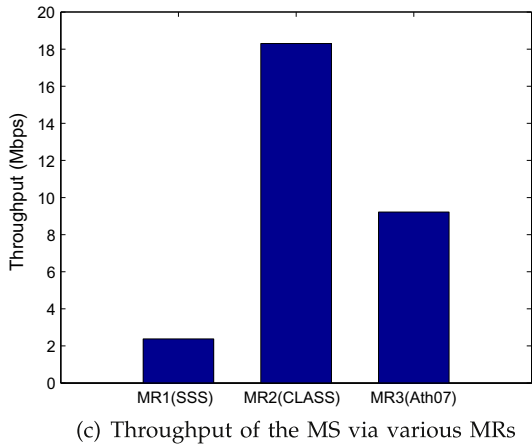
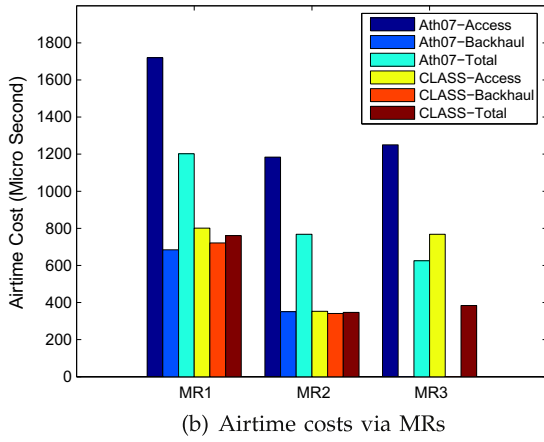
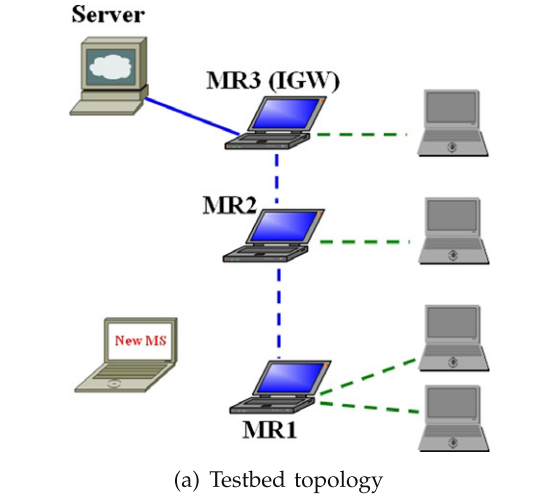


Fig. 7. Experiment 2: Investigate the impact of traffic load on airtime cost.

via various MRs. Because Ath07 only considers the uplink routing cost for the backhaul airtime cost and the uploading traffic of the MR1-MR2-MR3 branch is higher than that of the MR4-MR5-MR6 branch, MR1 (2 hops from the IGW) and MR2 (1 hop from the IGW) bring higher airtime cost to the MS than MR4 and MR5 in the measurement of Ath07, respectively. CLASS evaluates the backhaul airtime cost

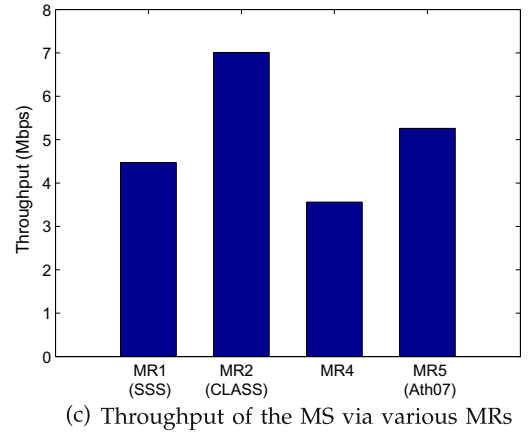
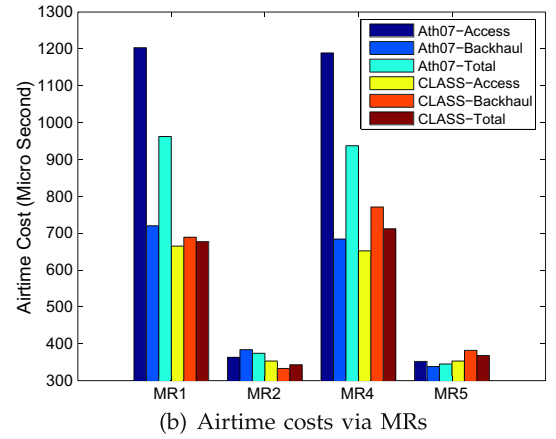
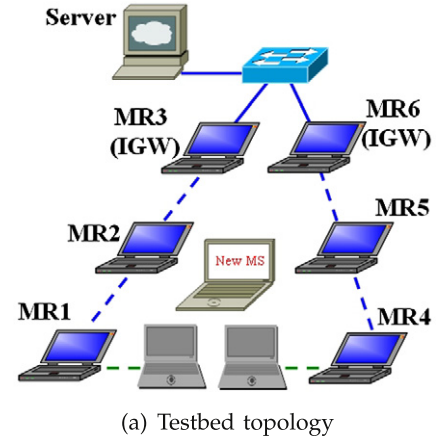


Fig. 8. Experiment 3: Investigate the impact of asymmetric uplink and downlink backhaul transportation costs on airtime cost.

via each MR based on the traffic pattern of the MS, and assigns 0.9 to β in this case. Thus, different from the conclusion of Ath07, MR1 and MR2 bring lower airtime cost to the MS than MR4 and MR5, respectively. MR2 is selected by CLASS for the lowest total airtime cost, and provides the MS with the highest throughput as shown in Fig. 8c. Hence, considering the asymmetric uplink and downlink backhaul transportation costs in the airtime cost measurement is

important to identify the correct MR which provides the MS with the best end-to-end performance.

5.4. Experiment 4: Investigating the association probing latency by CLASS

As highlighted by the previous experiments, a key design goal of CLASS is to select an MAP that yields the maximum end-to-end throughput performance for an MS. This result is determined after the actual association process. A second fundamental design goal for CLASS is to maintain an acceptably low total association latency, from the user perspective.¹

To better understand the overall latency, we investigate the latency of each step involved in the association probing procedure of CLASS. An MS will switch to each possible AP channel and scan all the MAPs on that channel. The association probing latency on each channel consists of four parts: (1) channel switching latency for the MS to switch to the AP channel; (2) channel scanning latency for the MS to discover the neighboring MAPs operating on the AP channel; (3) the latency for the MS to send the AIR messages and receive the AIP messages; and (4) the latency of transmitting probing data packets to each candidate MAP in order to measure the frame error rate. Fig. 9 shows the channel switching latency for 10 replicates. The average channel switching latency is 5.425 ms, which is consistent with the experimental results in previous research [21,22]. In our measurements, we found the beacon message of an MAP may not be successfully received by the MS when either the traffic load of the AP channel is high or the interference is high. Thus, we set the timer for channel scanning to 220 ms, which allows the MS two opportunities to receive the beacon message from each MAP² and thus enhances the effectiveness of MAP discovery. Fig. 10 shows that the association probing latency on a single AP channel as a function of the number of MAPs. In these experiments, all the MAPs are two hops away from the IGW, and the AIR-AIP messaging for each MAP takes 50–60 ms. Forty probing data packets are transmitted to each MAP with the interval of 10 ms, and the corresponding latency is around 480 ms for each MAP. As shown in Fig. 10, the total association probing latency on a single AP channel varies from 0.758 s to 1.828 s, when the number of candidate MAPs on that channel increases from 1 to 3. When the MS scans all the eleven channels of 802.11b/g, the total association probing latency can be up to 20 s if the average number of MAPs on each channel is 3. In practice, most MAPs are deployed on the three non-overlapping channels for 802.11b/g (i.e., Channels 1, 6 and 11), and thus the number of MAPs on other channels is much lower. Moreover, the candidate MAPs are selected by CLASS only if the received signal quality of their beacon messages is higher than the specified threshold θ . Therefore,

¹ In practice, the notion of *acceptable* or *tolerable* association latency may be based on user perception. Quantifying such a value is outside the scope of this work. Instead, we assume that there are various classes of users. For example, a 3-minute latency is not acceptable to any user. However, a 10-s latency may be tolerable for 95% of users while a 20-s latency may be acceptable to 60–70% users.

² The beacon broadcast interval at MAPs is about 100 ms.

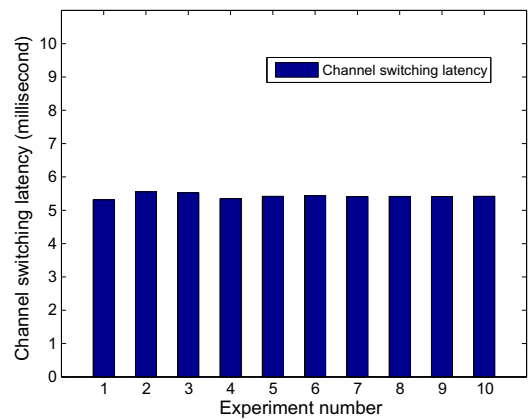


Fig. 9. Channel switching latency.

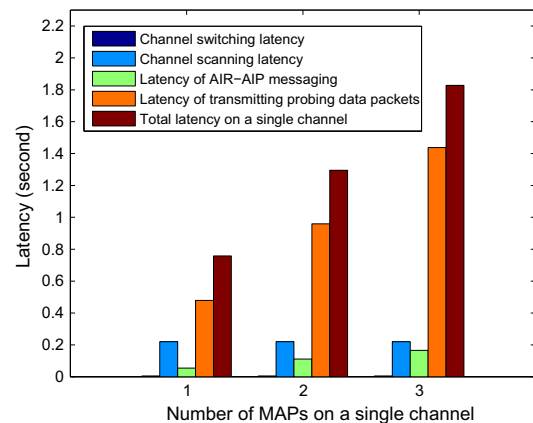


Fig. 10. Association probing latency on a single channel.

in practice, the actual number of MAPs to be probed is less than the aforementioned amount. A further optimization of association probing procedure is to skip the overlapping channels if the MS has discovered an MAP on any of the three non-overlapping channels, via which the total airtime cost of the MS is lower than a specified threshold. Assuming that most MAPs are operating on one of the three non-overlapping channels, this strategy will discover the best MAP for the MS in most cases and further reduce the association probing latency. After the MS has determined the best MAP with which to associate, it will switch to the AP channel of that MAP (if the MS is not on that channel currently), authenticate, and then associate with that MAP as specified by the IEEE 802.11 standard [4]. Fig. 11 shows that the association latencies (including channel switching latency) for 10 replicates are all under 11 ms. Thus, the association latency itself is negligible compared to the association probing latency.³

³ It should be noted that association latency may increase if a complicated authentication method (such as the full 802.1X EAP authentication [23]) is used.

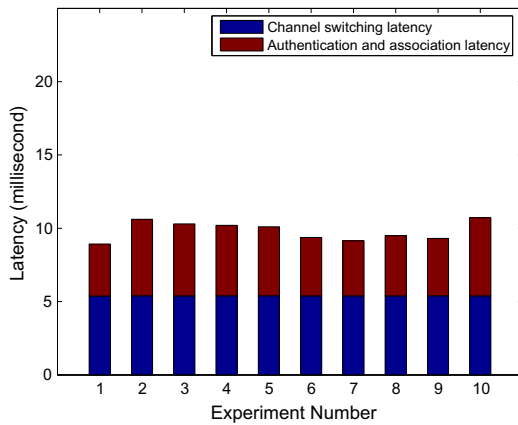


Fig. 11. Association latency.

6. Summary and future work

In this paper, we have presented a comprehensive description, including the implementation details and an experimental performance evaluation of CLASS, a Cross-Layer Association scheme for WMNs. The key attribute of CLASS is that it considers the importance of the traffic load on the access link, the dominant packet size of the client traffic, and the asymmetric uplink and downlink backhaul transportation costs in the measurement of airtime cost, the metric used for the association decision. Experimental results show that CLASS is able to identify the MAP which provides the MS with the *best* end-to-end performance with respect to throughput, and the total association latency (including probing latency) induced by CLASS is tolerable to the users. In our future work, we will evaluate CLASS on our on-campus testbed, which comprises 13 Soekris-board based mesh routers running our customized Debian Linux (Kernel 2.6.26).

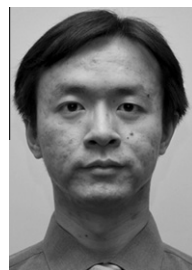
Acknowledgments

This work was supported in part by the National Science Foundation (NSF) under NSF Career Grant No. 0448055, the US Department of Energy (DOE) under Award Number DE-FG02-04ER46136, and by the State of Louisiana, Louisiana Board of Regents under Contract Numbers DOE/LEQSF(2004-07)-ULL and LEQSF(2003-06)-RD-A-35.

References

- [1] IEEE Standard for Information Technology: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, High-speed Physical Layer in the 5 GHz Band.
- [2] IEEE Standard for Information Technology: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band.
- [3] Joint SEE-Mesh/Wi-Mesh Proposal to 802.11 TGs, RWTH Aachen University, 2006.

- [4] IEEE Standard for Information Technology: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Computer Society LAN MAN Standards Committee, 1999.
- [5] V. Mhatre, K. Papagiannaki, Using smart triggers for improved user performance in 802.11 wireless networks, in: Proceedings of ACM MobiSys, June 2006, pp. 246–259.
- [6] S. Vasudevan, K. Papagiannaki, C. Diot, J. Kurose, D. Towsley, Facilitating access point selection in IEEE 802.11 wireless networks, in: Proceedings of USENIX Internet Measurement Conference (IMC'05), 2005, pp. 293–298.
- [7] H. Lee, S. Kim, O. Lee, S. Choi, S.-J. Lee, Available bandwidth-based association in IEEE 802.11 wireless lans, in: Proceedings of the 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), October 2008, pp. 132–139.
- [8] A.J. Nicholson, Y. Chawathe, M.Y. Chen, Improved access point selection, in: Proceedings of ACM MobiSys, June 2006, pp. 233–245.
- [9] G. Athanasiou, T. Korakis, O. Ercetin, L. Tassiulas, Dynamic cross-layer association in 802.11-based mesh networks, in: Proceedings of IEEE Infocom, May 2007, pp. 2090–2098.
- [10] A. Giannoulis, M. Fiore, E.W. Knightly, Supporting vehicular mobility in urban multi-hop wireless networks, in: Proceedings of ACM MobiSys, June 2008, pp. 54–66.
- [11] S. Makhoulf, Y. Chen, S. Emeott, M. Baker, A network-assisted association scheme for 802.11-based mesh networks, in: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'08), 2008, pp. 1339–1343.
- [12] L. Luo, D. Raychaudhuri, H. Liu, M. Wu, D. Li, Improving end-to-end performance of wireless mesh networks through smart association, in: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'08), 2008, pp. 2087–2092.
- [13] W.-S.S.M.M. Hui Wang, Wai-Choong Wong, Dynamic association in IEEE 802.11 based wireless mesh networks, in: Proceedings of the 6th International Symposium on Wireless Communication Systems, 2009, pp. 81–85.
- [14] Multiband Atheros Driver for WiFi (MADWiFi). <<http://madwifi.org>>.
- [15] D.C. Montgomery, Design and Analysis of Experiments, John Wiley & Sons, Inc., 2005.
- [16] A. Kumar, E. Altman, D. Miorandi, M. Goyal, New insights from a fixed point analysis of single cell IEEE 802.11 WLANs, in: Proceedings of IEEE Infocom, March 2005.
- [17] R.K. Guha, S. Sarkar, Characterizing temporal SNR variation in 802.11 networks, in: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'06), 2006, pp. 1408–1413.
- [18] J. Geier, Wi-Fi: Define Minimum SNR Values for Signal Coverage, May 2008. <<http://www.enterprisenetworkingplanet.com/netsp/article.php/3747656>>.
- [19] J. Malinen, Host AP driver for Intersil Prism2/2.5/3. <<http://hostap.epitest.fi>>.
- [20] A. Tirumala, F. Qin, J. Dugan, J. Ferguson, K. Gibbs, Iperf. <<http://dast.nlanr.net/Projects/Iperf/>>.
- [21] I. Ramani, S. Savage, Syncscan: practical fast handoff for 802.11 infrastructure networks, in: Proceedings of IEEE INFOCOM, 2005.
- [22] C. Cherred, System Architecture for Multichannel Multi-interface Wireless Networks, Master's thesis, University of Illinois at Urbana-Champaign, 2006.
- [23] RFC2865, Ietf standard for remote authentication dial in user service (radius), 2000.



Yan He received the BS degree and MS degree in Computer Science from Hunan University, China, in 2000 and 2003, respectively. After that, he was working as a software engineer in Shanghai, China. Since 2006, he has been studying in the Ph.D. program at the Center for Advanced Computer Studies in the University of Louisiana at Lafayette. His research interests are in the areas of mobile ad hoc networks, wireless mesh networks, and 700 MHz Wi-Fi.



Dmitri Perkins received the Ph.D. and MS degree in computer science from Michigan State University in 2002 and 1997, respectively. He received the BS degree in computer science in 1995 from Tuskegee University. Currently, he is an associate professor at the Center for Advanced Computer Studies (CACS), University of Louisiana (UL) at Lafayette. His research interests are in the areas of broadband multi-hop wireless systems, mobile ad hoc networks, wireless sensor networks, and wireless cyber-physical systems.

He has served as a technical committee member of several IEEE and ACM conferences and has published more than two dozen technical papers in leading journals and conference proceedings. In 2000, he received the GE Faculty of the Future Fellowship and Grant. Dr. Perkins received a National Science Foundation (NSF) CAREER Award in 2005 and, in 2008,

was named the Hardy Edmiston Endowed Professor of Computer Science by the University of Louisiana at Lafayette.



Sritej Velaga received his B.E degree in Telecommunication Engineering from M.S Ramaiah Institute of Technology, Bangalore, India in the year 2007. He received his M.S. degree in Computer Science from the Center for Advanced Computer Studies, University of Louisiana at Lafayette in 2009. He is currently a Linux device driver engineer at Qlogic Corporation. His research interests include Gigabit Networks, NIC virtualization and 802.11 wireless networks.