This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

A Two-Stage Link Scheduling Scheme for Variable-Bit-Rate Traffic Flows in Wireless Mesh Networks

Yung-Cheng Tu, Meng Chang Chen, and Yeali S. Sun

***Abstract*—Providing end-to-end QoS for delay sensitive flows with variable-bit-rate (VBR) traffic in wireless mesh networks is a major challenge. There are several reasons for this phenomenon, including time-varied bandwidth requirements, competition for transmission opportunities from flows on the same link, and interference from other wireless links. In this paper, we propose a flexible bandwidth allocation and uncoordinated scheduling scheme, called two-stage link scheduling (2SLS), to support flow delay control in TDMA-based wireless mesh networks. The scheme is implemented in two stages: slot allocation and on-the-go scheduling. The slot allocation mechanism allocates contiguous time slots to each link in each frame based on pre-defined maximum and minimum bandwidth requirements. Then, each link's on-the-go scheduling mechanism dynamically schedules the transmissions within the allocated time slots. The objective is to maximally satisfy the demand of all flows on the link according to the bandwidth requirements and channel condition. Compared to traditional slot allocation approaches, 2SLS achieves higher channel utilization and provides better end-to-end QoS for delay sensitive flows with VBR traffic.**

***Index Terms*—wireless mesh networks, TDMA-based scheduling, flexible bandwidth allocation, 2-stage link scheduling.**

I. Introduction

Because of their affordability and ease of deployment, wireless mesh networks(WMNs) have been developed as cost-effective networking platforms to support ubiquitous broadband Internet access in several cities [1]–[3]. The Wi-Fi mesh network also helps offload data from overloaded 3G/LTE networks [4], [5]. However, providing end-to-end quality of service (QoS) guarantees for traffic flows in wireless mesh networks is a challenging task due to variations in the available bandwidth and signal interference between wireless links. In this paper, we focus on an even greater challenge: QoS control for variable-bit-rate (VBR) delay sensitive flows, such as VoIP, video conferencing and multimedia streaming, in a wireless mesh network.

It is known that contention-based protocols, like CSMA/CA used in IEEE 802.11 [6], are impaired by the problems of service unfairness and the inability to control packet delays. Although the unfairness problem can be overcome by applying enhancements, such the Bidirectional-DCF (BDCF) protocol [7], on IEEE 802.11 DCF protocol, the problem still exists in multi-hop wireless mesh networks [8], [9]. In contrast, time division multiple access (TDMA) based MAC protocols, such as IEEE 802.16j [10] and 3GPP LTE Time-Division-Duplex (LTE TDD) [11], provide collision-free communications and allow fine control of the throughput and delay of network traffic. With the introduction of TDMA-based scheduling in contention-based IEEE 802.11 networks, a standard draft called IEEE 802.11s [12], was recently developed for mesh networks. By exploiting the mesh coordination function (MCF) in IEEE 802.11s, each link can reserve some time slots for its future transmissions without competing with other links for transmission opportunities. However, the most effective way to allocate and schedule slots for each link to guarantee end-to-end QoS for delay sensitive flows is still an open issue.

Several TDMA-based scheduling algorithms have been proposed to maximize the network throughput of multi-hop wireless mesh networks [13]–[23], but only a few of them [16]–[19] consider the issue of end-to-end QoS for delay sensitive flows. Generally, the algorithms can be divided into two categories: centralized algorithms [13]–[19] and distributed algorithms [20]–[23]. For centralized algorithms, the central controller collects the bandwidth requirements of all links, executes the scheduling algorithm and delivers the scheduling results to all the wireless nodes. A centralized scheduler may be able to arrange an optimum schedule; however, collecting the network data, executing the algorithm and delivering the scheduling decisions overload the central controller and its connecting links. In contrast, a distributed scheduling algorithm applied by each wireless node can derive a locally optimal schedule without the information about the whole network. To ensure there is no conflict between the transmissions of the wireless links, the wireless nodes must use a coordination mechanism, which incurs some overhead and latency.

As mentioned above, in both centralized and distributed scheduling algorithms, there is certain amount of latency between the time a link requests bandwidth and the time it receives the scheduling result. To resolve the problem, most TDMA-based scheduling algorithms assume that each flow's traffic load is fixed or follows a specific pattern, and then allocate a fixed amount of bandwidth to each link in the WMN. However, the strategy usually results in either over-allocation of bandwidth or impairment of QoS when it is applied to VBR traffic flows.

In this paper, we propose a scheme called two-stage link scheduling (2SLS), which allows each wireless link to schedule its transmission time slots dynamically in order to meet

Yung-Cheng Tu and Meng Chang Chen are with Institute of Information Science and Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan.

Yeali S. Sun is with Department of Information Management, National Taiwan University, Taipei, Taiwan.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

the QoS requirements of VBR traffic flows on the link. The 2SLS scheme is decentralized because each link only needs the information about the network topology for scheduling without further coordination among links to deal with the situation that the traffic flow bandwidth requirements and available link capacity are time varying. In the first stage, a slot allocation mechanism allocates transmission slots to each flow to guarantee end-to-end bandwidth. The first stage slot allocation is performed when the flow is estimated before its transmission. Since the real-time traffic load and network condition are unknown in the first stage, the slots are allocated according to the bandwidth requirements estimated by the end-to-end QoS requirements and the flow specifications. Then, in the second stage, an adaptive on-the-go scheduling mechanism arranges each link's transmission schedule based on the QoS requirements of all flows on the link. The mechanism also prevents transmission collisions with other links. Unlike traditional slot allocation schemes, 2SLS allocates two types of time slots to each link: conflict-free slots and multi-access slots. It is guaranteed that transmissions in each links conflict-free slots will not be affected by interference from other links. By contrast, in multi-access slots, neighboring links may compete for transmission opportunities; however, the performance can be guaranteed by controlling the number of links that cause interference and arranging appropriate transmission time slots for those links. In the second stage, each link can adjust its transmission schedule dynamically without coordinating with other links. The results of simulations demonstrate that the proposed scheme is more flexible and efficient than traditional fixed slot allocation schemes.

The following example illustrates the advantages of the 2SLS scheme over IEEE 802.11 and TDMA-based fixed rate scheduling scheme. An optimum algorithm, which is executed for each time unit with full knowledge of whole network, s used as the best-case baseline. As shown in Figure 1, we consider a six-node chain network topology with channel capacity being 11Mbps, where users 1, 2 and 3 receive videos 1, 2 and 3 sourced from nodes na, nc and ne, respectively. The source of each video is an H.264 encoded VBR film [24] with mean rate of 630Kbps. In Figure 2, we compare the end-to-end transmission rate of video 1 under different link scheduling schemes. Under this setup, the fixed rate TDMA-based scheduling scheme can allocate up to 980kbps to each flow to fully utilize the channel capacity, and, thus, the transmission rate 980kbps is assigned in this experiment. We can see that the 2SLS provides higher transmission rate than IEEE 802.11 and TDMA-based scheduling scheme especially when the offered load peaks up. For the optimum algorithm scheme, a centralized controller is needed to gather real-time information about the whole network and compute the best schedule for all the transmission links in each time slot based on an optimal throughput algorithm [19]. The problem is NP-hard, and there is a certain latency to deliver the scheduling results to all wireless nodes. We observe that the performance of 2SLS scheme is very close to that of the optimal scheduling scheme without message exchanges once the flow is established. This approach involves much less computation than the optimal algorithm and does not incur any scheduling latency.

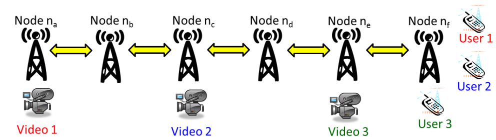

Fig. 1. An example of wireless mesh network transmissions.

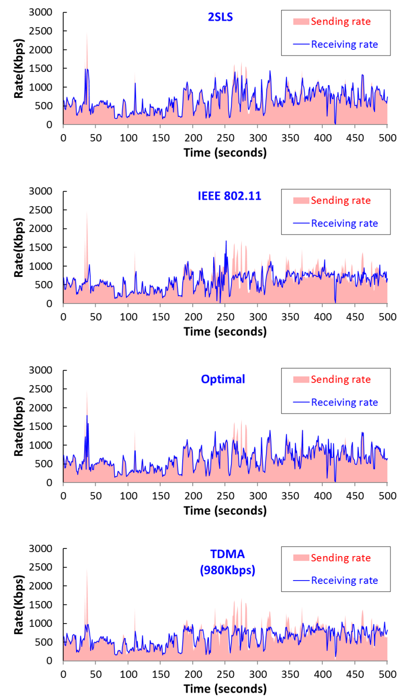

Fig. 2. The sending rate of Video 1 and receiving rate of User 1 under different scheduling schemes in the wireless mesh network as shown in Figure 1.

The remainder of this paper is organized as follows. In Section 2, we review related works on TDMA-based scheduling with end-to-end delay control. In Section 3, we define the network and system model of our approach; and in Section 4, we describe the proposed two-stage slot allocation scheme (2SLS). In Section 5, we discuss the results of simulations performed to evaluate the performance of the scheme. Section 6 contains some concluding remarks.

II. Related Work

Providing an end-to-end QoS guarantee for a flow, such as its bandwidth or delay, requires the QoS support of each link along its routing path. One of the major challenges in providing end-to-end QoS guarantees for VBR traffic flows in TDMA-based wireless mesh networks is how to adjust each link's bandwidth allocation rapidly based on the variations in the traffic load of flows and network transmission conditions. In [16], Djukic and Valaee formulate the problem of computing a conflict-free link schedule with end-to-end delay constraints as a mixed-integer non-linear problem, and show that the delay-aware TDMA scheduling problem can be solved by dividing it into two parts. The first part involves deciding the relative transmission order of all the links; and the second involves finding a schedule with the minimum frame size

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

based on the relative transmission order and the interference between the links. Cappanera et al. [17] propose a similar link scheduling formulation, but they assume that the traffic source of each flow is leaky-bucket constrained. Although [16] and [17] use heuristic algorithms to solve the problem, they both require a centralized controller and the time slot allocations are fixed according to the given traffic patterns of all flows. The proposed 2SLS scheme utilizes much of their problem formulation in the slot allocation stage, but it provides flexible scheduling so that each link can adapt to the dynamic traffic loads without a central controller.

Most works on TDMA-based link scheduling focus on maximizing the network throughput. It has been shown that backpressure-based algorithms can achieve optimal network throughput and low end-to-end delays for all flows [18], [19]. This kind of algorithms rely on a centralized controller to solve a maximum weighted independent set (MWIS) problem, where the weight may refer to the difference between the lengths of queues or the queueing delays in each slot time. Even if a heuristic largest-weight-first (LWF) algorithm is used [19], a centralized controller is still required, which therefore is not suitable for a network with the traffic load of each flow changes frequently. Another drawback of the above works is that they ignore the high overheads incurred by status collection and the latency in delivering the scheduling results to all links.

III. Network Model and Notations

A wireless mesh network is modeled as a directed network graph $NG = (N, V)$, where $N = \{n_1, n_2, n_3, ...\}$ is the set of all wireless nodes, and $V = \{v_1, v_2, v_3, ...\}$ is the set of all directed links in the wireless mesh network. The latter are used by the links (n_a, n_b) to indicate that the node n_a can transmit data to node n_b; that is, $v_k = (n_a, n_b) \in V$ if $|n_a, n_b| \leq D_{TR}$, where $|n_a, n_b|$ is the distance between n_a and n_b, and D_{TR} is the maximum transmission range. Two links can only transmit data successfully in the same time slot if their transmissions do not interfere with each other. Any interference model, such as K-hop interference model [25], physical model or protocol model [26], can be applied to our system to determine whether or not two links can transmit data concurrently. In this paper, we utilize the K-hop interference model because it is used by most MAC protocols, such as IEEE 802.11s and IEEE 802.16. Under the model, no two links that are within K-hops of one another can transmit successfully at the same time. Let $\langle v_k, v_l \rangle$ denote the shortest hop between the two links v_k and v_l. We can construct an undirected contention graph $CG = (V, E)$, where V is the same as in NG; and an edge (v_k, v_l) is in E if links v_k and v_l cannot transmit data simultaneously, i.e., $(v_k, v_l) \in E, if \langle v_k, v_l \rangle \leq K$. This network model can be extended to many other interference models, such as the SINR interference mode. When use the SINR model, the definition of E becomes $(v_k, v_l) \in E$, if $SINR(n_a, n_b) < \beta$ or $SINR(n_c, n_d) < \beta$ when both v_k and v_l are transmitting and all other links are idle, where $v_k = (n_a, n_b)$, $v_l = (n_c, n_d)$, $SINR(x, y)$ represents the received SINR in y from x and β is the threshold that the receiver can successfully decode the received signals. The edges in E are undirected and we regard v_l and v_k as neighbors in CG if $(v_k, v_l) \in E$. Then, we define the interference set of link v_k as follows:

$$IF(v_k) = \{v_l \,|\, (v_k, v_l) \in E\} \tag{1}$$

Figure 3 shows the network graph of the wireless mesh network in Figure 1 and its corresponding contention graph using a 2-hop interference model. In the network graph, the solid lines and dashed lines represent, respectively, the links and the interference between the nodes.

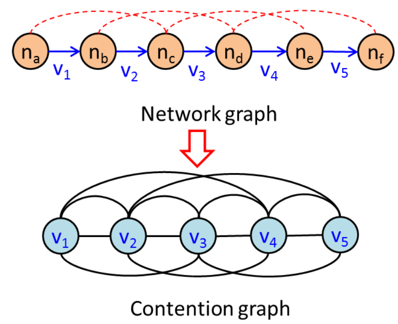

Fig. 3. An example of a network graph and its corresponding contention graph.

A transmission flow f_i in a wireless mesh network comprises a routing path, defined as $Path_i = \{v_k | f_i \text{ passing through the link } v_k \in V\}$; and the end-to-end bandwidth demand rate of the flow is in the range $(\phi_i^{min}, \phi_i^{max})$, where ϕ_i^{min} and ϕ_i^{max} are the minimum and maximum demand rates of f_i respectively. Then, the bandwidth requirement of link v_k is bounded by (r_i^{min}, r_i^{max}), where r_i^{min} is the sum of the minimum demand rates and r_i^{max} is the sum of the maximum demand rates of all flows passing through v_k. Like most TDMA-based protocols, the timeline of 2SLS is divided into recurrent frames, each comprised of M fixed-length time slots. Let the demand rate of flow f_i on link v_k in frame j be $\phi_{i,k}(j)$; then, the bandwidth requirement $r_k(j)$ and slot requirement $t_k(j)$ of link v_k in frame j are as follows:

$$r_k(j) = \sum \phi_{i,k}(j), \tag{2}$$

$$t_k(j) = \lceil M \times \frac{r_k(j)}{C_k} \rceil, \tag{3}$$

where C_k is the channel capacity of link v_k. Similarly, we can derive the minimum and maximum slot requirements, denoted as t_k^{min} and t_k^{max} respectively, of link v_k from r_k^{min} and r_k^{max} by Equation (3).

IV. The 2-Stage Link Scheduling Scheme

The system framework of the proposed 2-stage link scheduling (2SLS) scheme is shown in Figure 4. It is comprised of two stages: (1) the *slot allocation stage*, which reserves at least t_k^{min} slots and up to t_k^{max} slots for the link v_k; and (2) the *on-the-go scheduling stage*, which schedules slots for each link to transmit VBR traffic according to its real-time bandwidth demand in each frame and to avoid collisions with other links. The 2SLS scheme exploits per-flow queueing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

whereby the demand rate $\phi_{i,k}(j)$ of the flow f_i in frame j is estimated based on the flow's traffic load and end-to-end QoS requirement. A bandwidth estimation algorithm that supports end-to-end QoS control, such as the per-node delay assignment scheme [27] or the bulk scheduling scheme [28], may be used to estimate $\phi_{i,k}(j)$. As bandwidth estimation is not the focus of this study, we assume $\phi_{i,k}(j)$ is determined by an existing estimation algorithm and do not discuss the issue further. The slot allocation stage only reserves slots for a flow when the flow joins a link, and each link's on-the-go scheduler assigns transmission slots to the link in each frame to meet the dynamic VBR demands of all flows. The obvious advantage of 2SLS over other schemes is that it only needs to reserve slots once for each flow, regardless of the fluctuations in VBR demand.

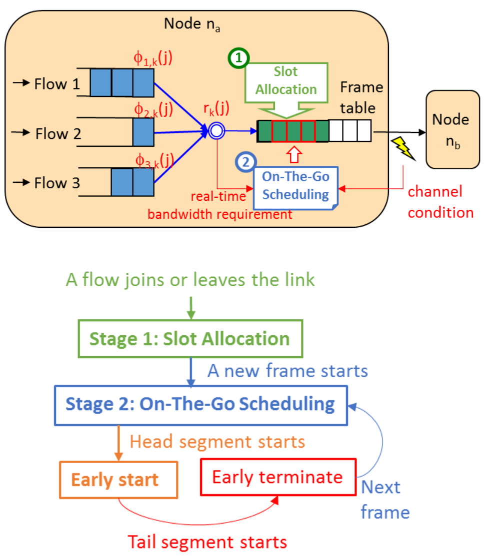

Fig. 4. The framework of the 2SLS scheme.

A. Conflict-free and multi-access slots

In the first stage of 2SLS, the objective is to allocate slots for each link v_k such that (1) the minimum slot requirement t_k^{min} can be guaranteed; and (2) the maximum slot requirement t_k^{max} can be satisfied as much as possible. As mentioned earlier, the slot allocation mechanism in 2SLS assigns *conflict-free slots* and *multi-access slots to each link*. The conflict-free slots assigned to a link v_k cannot be allocated to any of its neighbors in an undirected contention graph CG, but multi-access slots may be allocated to some of the link's neighbors. All the slots are listed in a frame table $\mathbf{F}$, and each slot is labeled in sequence, e.g., the i-th slot is labeled i. Therefore, $\mathbf{F} = \{\mathbf{1}, \mathbf{2}, \ldots, \mathbf{M}\}$, where M is the frame size. Let $\Delta(i,j)$ denote the distance between the i-th slot and the j-th slot in the frame table, i.e.,

$$\Delta(i,j) = \begin{cases} j-i, & \text{if } j \leq i \\ j-i+M, & \text{if } j < i \end{cases} \tag{4}$$

Note that based on the definition of $\Delta(i,j)$, we have $\Delta(i,j) = M - \Delta(j,i)$ and $\Delta(i,j) + \Delta(j,k) = \Delta(i,k)$ if and only if $\Delta(i,j) + \Delta(j,k) \leq M$. Then, we can define the contiguity of a set of slots as follows:

Definition 1: In a set of slots $\mathbf{Z}, \mathbf{Z} \subseteq \mathbf{F}$, the slots are contiguous if we can find a slot i in $\mathbf{Z}$ such that all the subsequent $|\mathbf{Z}| - \mathbf{1}$ slots of i in the frame table are in $\mathbf{Z}$, i.e.,

$$\begin{aligned} &\exists i \in \mathbf{Z},\ \forall j : (j \in \mathbf{F} \text{ and } 1 \leq \Delta(i,j) \leq |\mathbf{Z}| - 1) \\ &\rightarrow \mathbf{Z} \text{ is contiguous.} \end{aligned} \tag{5}$$

Let $\mathbf{Z_k}$ denote the set of all slots (i.e., conflict-free and multi-access slots) and $\mathbf{Z_k^{CF}}$ denote the set of conflict-free slots allocated to link v_k. The 2SLS model makes the following assumptions.

Assumption 1: All the slots allocated to a link are contiguous.

Assumption 2: For any link v_k that flows pass through, called an active link, its minimum slot requirement t_k^{min} is at least one slot.

Assumption 1 corresponds to the operation of IEEE 802.16 mesh networks [10], and it is also exploited in [16] [17]. Although Assumption 1 would reduce the feasibility region, it has benefits such as the improvement of channel utilization and the reduction of the message exchange overhead. With of the restriction that the allocated slots must be contiguous, the transmissions of a link are grouped so the number of scatted space can be reduced to increase the channel utilization. Besides, each node can only send the start time and end time of the grouped transmissions rather than a group of time slots in message exchanging between nodes. Therefore, the overhead of coordinating data exchange between the links can be reduced. Assumption 2 is designed to ensure that active links are not starved of slots. Based on the two assumptions, the time slots allocated to the link v_k can be represented by:

$$\mathbf{Z_k} = \{\mathbf{i} \,|\, \mathbf{\Delta}(\mathbf{s_k}, \mathbf{i}) < \mathbf{t_k}\}, \tag{6}$$

where s_k is the first slot of the contiguous slots allocated to v_k, and t_k is the number of slots allocated to v_k ($t_k^{min} \leq t_k \leq t_k^{max}$). Next, we show that the conflict-free slots allocated to each link must also be contiguous.

Theorem 1: If the slots allocated to all links are contiguous and at least one of them is conflict-free, then the conflict-free slots $\mathbf{Z_k^{CF}}$ allocated to each link v_k must also be contiguous.

Proof: Let $\vec{\mathbf{Z}}_\mathbf{k}^\mathbf{CF}$ be the ordered set of $\mathbf{Z_k^{CF}}$, where all the elements in $\vec{\mathbf{Z}}_\mathbf{k}^\mathbf{CF}$ are sorted in ascending order based on their distance from s_k; and let $\vec{z}_k^{CF}(i)$ denote the i-th element of $\vec{\mathbf{Z}}_\mathbf{k}^\mathbf{CF}$, i.e., $\Delta\left(s_k, \vec{z}_k^{CF}(i)\right) < \Delta\left(s_k, \vec{z}_k^{CF}(j)\right)$, if $i < j$. Because $\vec{\mathbf{Z}}_\mathbf{k}^\mathbf{CF} \leq \mathbf{Z_k^{CF}}$, we have

$$0 \leq \Delta\left(s_k, \vec{z}_k^{CF}(1)\right) < \Delta\left(s_k, \vec{z}_k^{CF}(2)\right) < \cdots \leq t_k. \tag{7}$$

It is clear that $\mathbf{Z_k^{CF}}$ is contiguous if $|\mathbf{Z_k^{CF}}| = 1$; therefore, in the following, we only consider the cases where $2 \leq |\mathbf{Z_k^{CF}}| \leq M$. First, we consider the slots $\vec{z}_k^{CF}(1)$ and $\vec{z}_k^{CF}(2)$. If $\Delta\left(\vec{z}_k^{CF}(1), \vec{z}_k^{CF}(2)\right) > 1$, it means the set of slots between $\vec{z}_k^{CF}(1)$ and $\vec{z}_k^{CF}(2)$,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

denoted by $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2)$, is not empty, i.e., $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2) = \{i|\Delta\left(s_k, \vec{z}_k^{CF}(1)\right) < \Delta(s_k, i) < \Delta\left(s_k, \vec{z}_k^{CF}(2)\right)\} \neq \emptyset$. Hence, we can infer that all the slots in $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2)$ are also allocated to link v_k because

$$0 \le \Delta\left(s_k, \vec{z}_k^{CF}(1)\right) < \Delta(s_k, i) < \Delta\left(s_k, \vec{z}_k^{CF}(2)\right) < t_k \\ \rightarrow i \in \mathbf{Z_k}. \tag{8}$$

The slots in $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2)$ must be multi-access slots, which means they can be allocated to the neighbors of v_k. Let slot i be a slot in $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2)$. According to Assumption 1, if slot i is allocated to a link v_l, $v_l \in IF(v_k)$, then $\mathbf{Z_l}$ must between $\vec{z}_k^{CF}(1)$ and $\vec{z}_k^{CF}(2)$ because $\mathbf{Z_l}$ is contiguous and cannot contain $\vec{z}_k^{CF}(1)$ or $\vec{z}_k^{CF}(2)$. Moreover, according to Assumption 2, $\mathbf{Z_l}$ contains at least one conflict-free slot, which means that at least one slot between $\vec{z}_k^{CF}(1)$ and $\vec{z}_k^{CF}(2)$ cannot be allocated to v_k. However, we know that all the slots between $\vec{z}_k^{CF}(1)$ and $\vec{z}_k^{CF}(2)$, i.e., $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}(1,2)$, are multi-access slots of v_k, which leads to a contradiction. Therefore, we can infer that $\Delta\left(\vec{z}_k^{CF}(1), \vec{z}_k^{CF}(2)\right) = 1$, which means the next slot of $\vec{z}_k^{CF}(1)$ in the frame table is $\vec{z}_k^{CF}(2)$.

Similarly, for any element $\vec{z}_k^{CF}(i)$ in $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}$, where $2 \le i \le |\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}| - 1$, we can derive that the next slot j of $\vec{z}_k^{CF}(i)$ in the frame table is the slot $\vec{z}_k^{CF}(i+1)$. Then, $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}$ is contiguous because for the slot $\vec{z}_k^{CF}(1)$, the subsequent $|\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}| - 1$ slots are $\vec{z}_k^{CF}(1)$, $\vec{z}_k^{CF}(2)$,..., $\vec{z}_k^{CF}\left(|\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}|\right)$, which are all in $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}$. Since $\mathbf{Z}_{\mathbf{k}}^{\mathbf{CF}}$ has the same elements as $\vec{\mathbf{Z}}_{\mathbf{k}}^{\mathbf{CF}}$, it is also contiguous. ∎

Theorem 1 also implies that allocated slots can only be arranged in an enclosed layout (i.e., conflict-free slots are enclosed by multi-access slots), as shown in Figure 5. Because link v_k requires at least t_k^{min} slots, we allocate t_k^{min} conflict-free slots to it. The conflict-free slots allocated to link v_k can be represented by

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{CF}} = \left\{i \,\middle|\, \Delta(s'_k, i) < t_k^{min}\right\}, \tag{9}$$

where s'_k is the position of the first contiguous conflict-free slot allocated to v_k. Therefore, we only need four parameters, s_k, s'_k, t_k^{min} and t_k, to represent the allocated slots of each link.

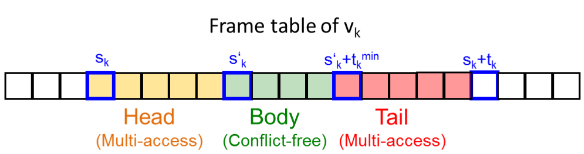

Fig. 5. The head, body and tail segments of the allocated slots of a link.

Because $\mathbf{Z}_{\mathbf{k}}^{\mathbf{CF}} \subseteq \mathbf{Z_k}$, we can derive the relation between s_k, s'_k, t_k^{min} and t_k from Equations (6) and (9) as follows:

$$0 \le \Delta(s_k, s'_k) \le t_k - t_k^{min}. \tag{10}$$

Moreover, since any slot allocated to v_k cannot be a conflict-free slot of any neighboring link of v_k, we can derive the relations between s_k, t_k, s_l and t_l if links v_k and v_l are neighbors in CG as follows:

$$\Delta(s_k, s_l) \ge \Delta(s_k, s'_k) + t_k^{min}, \text{if } (v_k, v_l) \in E \tag{11}$$

$$(M - t_l^{min} \ge \Delta(s_k, s'_l) \ge t_k) \text{ and} \\ (M - t_l \ge \Delta(s'_k, s_l) \ge t_k^{min}), \\ \text{if } (v_k, v_l) \in E \tag{12}$$

To allocate slots, we divide the link v_k into three segments, namely, the head segment $\mathbf{Z}_{\mathbf{k}}^{\mathbf{head}}$, the body segment $\mathbf{Z}_{\mathbf{k}}^{\mathbf{body}}$ and the tail segment $\mathbf{Z}_{\mathbf{k}}^{\mathbf{tail}}$ as follows:

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{head}} = \{i \,|\, \Delta(s_k, i) < \Delta(s_k, s'_k)\} \tag{13}$$

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{body}} = \mathbf{Z}_{\mathbf{k}}^{\mathbf{CF}} = \\ \left\{i \,\middle|\, \Delta(s_k, s'_k) \le \Delta(s_k, i) < \Delta(s_k, s'_k) + t_k^{min}\right\} \tag{14}$$

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{tail}} = \left\{i \,\middle|\, \Delta(s_k, s'_k) + t_k^{min} \le \Delta(s_k, i) < t_k\right\} \tag{15}$$

The body segment from slot s'_k to $s'_k + t_k^{min}$ contains all the conflict-free slots allocated to link v_k; the multi-access slots are divided into the head segment and the tail segment. In addition, we define a segment that is not an allocated segment as an idle segment of v_k, denoted by $\mathbf{Z}_{\mathbf{k}}^{\mathbf{idle}} = \{i \,|\, t_k \le \Delta(s_k, i)\}$, This leads to the following theorem.

Theorem 2: The transmission of slot i in the head segment of link v_k, i.e., $i \in \mathbf{Z}_{\mathbf{k}}^{\mathbf{head}}$, will only compete for transmission opportunities with link v_l, where $v_l \in IF(v_k)$ and v_l's tail segment $\mathbf{Z}_{\mathbf{l}}^{\mathbf{tail}}$ contains the slot i, i.e., $i \in \mathbf{Z}_{\mathbf{l}}^{\mathbf{tail}}$, Moreover, a transmission of slot i in a tail segment of link v_k, i.e., $i \in \mathbf{Z}_{\mathbf{k}}^{\mathbf{tail}}$, will only compete with the link v_l where $v_l \in IF(v_k)$ and $i \in \mathbf{Z}_{\mathbf{l}}^{\mathbf{head}}$.

Proof: Based on Equation (11) and the definition of $\Delta(\cdot)$, we can get that

$$\Delta(s_k, s_l) \ge \Delta(s_k, s'_k) + t_k^{min} \text{ and } \Delta(s_l, s_k) + \Delta(s_k, s_l) = M \\ \Rightarrow M - \Delta(s_k, s'_k) - t_k^{min} \ge \Delta(s_l, s_k). \tag{16}$$

If $i \in \mathbf{Z}_{\mathbf{k}}^{\mathbf{head}}$, we have $\Delta(s_k, i) < \Delta(s_k, s'_k)$ by Equation (13). Then, we can get that $\Delta(s_l, i) = \Delta(s_l, s_k) + Delta(s_k, i)$ because

$$\Delta(s_l, s_k) + \Delta(s_k, i) < \\ \left(M - \Delta(s_k, s'_k) - t_k^{min}\right) + \Delta(s_k, s'_k) < M \tag{17}$$

According to Equation (11), $\Delta(s_l, s_k) \ge \Delta(s_l, s'_l) + t_l^{min}$ and $\Delta(s_k, i) \ge 0$; the value of $\Delta(s_l, i)$ must satisfy

$$\Delta(s_l, i) = \Delta(s_l, s_k) + \Delta(s_k, i) \ge \Delta(s_l, s'_l) + t_l^{min} \tag{18}$$

Equation (18) indicates that $i \in \mathbf{Z}_{\mathbf{l}}^{\mathbf{tail}}$ (for $t_l > \Delta(s_l, i) \ge \Delta(s_l, s'_l) + t_l^{min}$) or $i \in \mathbf{Z}_{\mathbf{l}}^{\mathbf{idle}}$ (for $\Delta(s_l, i) \ge t_l$). As link v_l will not transmit packets in its idle segment, link v_k only needs to compete for transmission opportunities with link v_l if $i \in \mathbf{Z}_{\mathbf{l}}^{\mathbf{tail}}$.

If $i \in \mathbf{Z}_{\mathbf{k}}^{\mathbf{tail}}$, we have $\Delta(s_k, s'_k) + t_k^{min} \le \Delta(s_k, i) < t_k$. In the case of $\Delta(s'_k, s_l) + \Delta(s_l, i) \le M$, we can get that

$$\begin{aligned} \Delta(s_l, i) &= \Delta(s'_k, i) - \Delta(s'_k, s_l) \\ &= (\Delta(s_k, i) - \Delta(s_k, s'_k)) - \Delta(s'_k, s_l) \\ &< (t_k - \Delta(s_k, s'_k)) - \Delta(s'_k, s_l) \\ &= t_k - (\Delta(s_k, s'_k) + \Delta(s'_k, s_l)) \\ &= t_k - \Delta(s_k, s_l) \end{aligned} \tag{19}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

Since $\Delta(s_k, s_l) + \Delta(s_l, s'_l) = \Delta(s_k, s'_l)$, we know that

$$\Delta(s_l, i) < t_k - \Delta(s_k, s'_l) + \Delta(s_l, s'_l). \quad (20)$$

Moreover, according to Equation (12), $\Delta(s_k, s'_l) \geq t_k$; thus, $\Delta(s_l, i) < \Delta(s_l, s'_l)$, which means $i \in \mathbf{z_l^{head}}$. In the case of $\Delta(s'_k, s_l) + \Delta(s_l, i) > M$, we have $\Delta(s_l, i) > M - \Delta(s'_k, s_l) = \Delta(s_l, s'_k) \geq t_l$ according to Equation (12), which means $i \in \mathbf{z_l^{idle}}$. Therefore, link v_k only needs to compete for transmission opportunities with link v_l if $i \in \mathbf{z_l^{head}}$. ■

According to Theorem 2, if two links v_l and v_m are both neighbors of v_k, and v_l and v_m are neighbors of each other in CG, then only v_l or v_m will compete for transmission opportunities with v_k in each slot. This is because it is impossible for a slot to belong to the head(or tail) segments of two links if the two links are neighbors.

B. The slot allocation scheme

Based on Assumptions 1 and 2, we use four parameters, s_k, s'_k, t_k^{min} and t_k, to represent the slots allocated to each link v_k in a wireless mesh network. The relations between the slots allocated to any two links v_k and v_l are given in Equations (11) and (12). In the 2SLS slot allocation scheme, the objective is to determine s_k, s'_k and t_k, for each link v_k in order to satisfy the slot requirements of all links as much as possible. Note that the t_k represents the maximum allowed time slots for link v_k that larger t_k means v_k has higher probability to get more channel resource. In order to maximize the overall network throughput, we aim to maximize the total number of slots allocated to all the links in the network, i.e., the total number of t_k for all $v_k \in V$. Therefore, we define the global optimization problem as follows:

Global Optimization Problem

Given: $M, t_k^{min}, t_k^{max}, IF(v_k), \forall v_k \in V$

Find: $s_k, s'_k, t_k, \forall v_k \in V$

Maximize: $\sum_{k:v_k \in V} t_k$

Subject to:

$$t_k^{min} \leq t_k \leq t_k^{max}, \forall v_k \in V \quad (21)$$

$$0 \leq \Delta(s_k, s'_k) \leq t_k - t_k^{min}, \forall v_k \in V \quad (22)$$

$$\Delta(s_k, s_l) \geq \Delta(s_k, s'_k) + t_k^{min}, \forall(v_k \in V \wedge v_l \in IF(v_k)) \quad (23)$$

$$M - t_l^{min} \geq \Delta(s_k, s'_l) \geq t_k, \forall(v_k \in V \wedge v_l \in IF(v_k)) \quad (24)$$

$$M - t_l \geq \Delta(s'_k, s_l) \geq t_k^{min}, \forall(v_k \in V \wedge v_l \in IF(v_k)) \quad (25)$$

Constraint (21) is derived from the definition of t_k, and Constraints (22)-(25) are derived from Equations (10), (11) and (12). To determine the globally optimum slot allocation, it is necessary to consider all the network information and user requirements. However, the process incurs tremendous communication and computation overheads, and results in extended latency. Moreover, the slot allocations need to be re-computed if any link requirement changes. Instead of finding the global optimum solution, we adopt a distributed local optimization problem, where Constraints (22)-(25) are still held but each link v_k only maximizes its allocated time slots t_k. Since each link only competes time slots with its neighbors, Constraints (22)-(25) can be relaxed to only need to check with its neighbors. The distributed local optimal slot allocation problem is defined as follows.

Distributed Local Optimization Problem

Given:

$M, t_k^{min}, t_k^{max}, IF(v_k), (s_l, s'_l, t_l^{min}, t_l), \forall v_l \in IF(v_k)$

Find: $s_k, s'_k, t_k,$

Maximize: t_k

Subject to:

$t_k^{min} \leq t_k \leq t_k^{max}$

$0 \leq \Delta(s_k, s'_k) \leq t_k - t_k^{min}$

$\Delta(s_k, s_l) \geq \Delta(s_k, s'_k) + t_k^{min}, \forall v_l \in IF(v_k)$

$M - t_l^{min} \geq \Delta(s_k, s'_l) \geq t_k, \forall v_l \in IF(v_k)$

$M - t_l \geq \Delta(s'_k, s_l) \geq t_k^{min}, \forall v_l \in IF(v_k)$

To solve the distributed problem, each link v_k only needs to gather the information about the head, body and tail segments from each neighbor of v_k. The 2SLS scheme utilizes the message exchange mechanism in IEEE 802.11s, whereby each link v_l periodically broadcasts its slot allocation in the form of $(s_l, s'_l, t_l^{min}, t_l)$ to all its neighbors. In the k-hop interference model, the broadcast distance is k. The slot allocations of all the neighbors of link v_k can be obtained by the following two rules, where $\mathbf{N_k^{CF}}$ and $\mathbf{N_k^{MA}}$ are the sets of slots allocated by v_k's neighbors as conflict-free slots and multi-access slots respectively.

$$\mathbf{N_k^{CF}} = \bigcup_{v_l \in IF(v_k)} \mathbf{Z_l^{body}} \quad (26)$$

$$\mathbf{N_k^{MA}} = \bigcup_{v_l \in IF(v_k)} \mathbf{Z_l^{head}} \cup \bigcup_{v_l \in IF(v_k)} \mathbf{Z_l^{tail}} - \bigcup_{v_l \in IF(v_k)} \mathbf{Z_l^{body}} \quad (27)$$

The unused segment that has not been allocated to a neighbor of v_k (as shown in Figure 6) can be regarded as the conflict-free segment of link v_k. The 2SLS slot allocation scheme selects the unused segment, which is larger than or equal to t_k^{min} slots, with the largest surrounding multi-access segment as the body segment. After determining the body segment, we select at most $t_k^{max} - t_k^{min}$ slots from its surrounding multi-access segment as its head and tail segments. The distributed problem can be solved by a linear search ($O(n)$) algorithm, where n is the number of neighbor links of the link v_k. If all unused segments are smaller than t_k^{min} slots, the problem has no solution. In this case, the flow is not admitted to enter the WMN and all the slots allocated to the flow are released from all links.

When two neighboring links execute their slot allocation scheme link obtained from the local optimization algorithm concurrently, there may be allocation conflicts. To resolve the allocation conflict, each link is assigned with a unique identification number. If a link receives a message from its k-hop neighbor with a higher identification number and finds

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

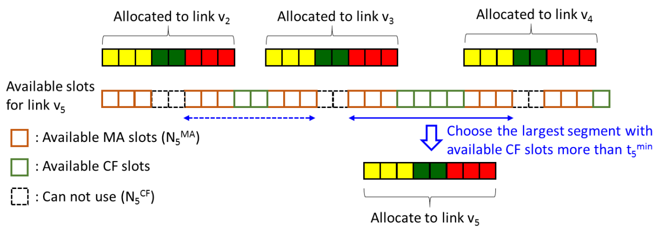

Fig. 6. An illustration of finding an allocation for link v_5.

the neighbors slot allocation conflicts with its own allocation, it discards its original allocation and executes the slot allocation scheme again. It is obvious that the network-wide slot allocations must converge in at most in $|V|$ iterations, where $|V|$ is the number of links in the WMN.

C. On-The-Go Scheduling Scheme

Although the slot allocation scheme assigns conflict-free and multi-access slots to each link v_k, we still need an efficient scheduling scheme to select the number and positions of slots from the allocated slots for transmission in each frame. The objective is to meet the immediate slot requirement $t_k(j)$ of each link v_k calculated by Equation (3). In the second stage of 2SLS, the on-the-go scheduling scheme has two functions: 1) it chooses slots dynamically for transmissions without coordinating with other links or waiting for the scheduling results provided by a central server; and 2) it prevents collisions between links and maximizes the network throughput. In practice, each link implements the on-the-go scheduling scheme to dynamically schedule the transmission time according to its immediate slot requirement and the channel condition.

1) Estimating channel availability: To prevent transmission collisions with neighbors, 2SLS uses the exponential moving average to estimate the channel availability of link v_k in the i-th slot of frame j as follows:

$$P_k^j(i) = \alpha \cdot I_k^{j-1}(i) + (1-\alpha) \cdot P_k^{j-1}(i), \quad (28)$$

where α is a parameter between 0 and 1, and $I_k^{j-1}(i)$ is an indicator function whose value is 1 if, for the link v_k, the channel is idle or the transmission is successful in slot i of frame j-1; otherwise, it is 0. The higher value of α results in higher sensitivity in detection of channel idle. Initially, $P_k^0(i)$ is set at 1 for all allocated slots of v_k and 0 for slots in $\mathbf{Z_k^{idle}}$. Clearly, $P_k^j(i)$ is always 1 if $i \in \mathbf{Z_k^{body}}$ and 0 if $i \in \mathbf{Z_k^{idle}}$. For each link, if a slot is not used by any neighboring links, the value of $P_k^j(i)$ will be increased. Then, the slot will have a higher probability of being scheduled for the link to transmit data. Conversely, if a link selects a slot for transmissions, the $P_k^j(i)$ of its neighbors will be reduced, so they will be less likely to select the slot for transmissions.

2) On-The-Go transmission time scheduling: As mentioned earlier, the slot requirement of v_k in frame j is $t_k(t)$. It has been shown that contiguous transmissions improve channel utilization [16]. Therefore, the on-the-go scheduling scheme selects at most $t_k(t)$ contiguous slots with the highest estimated channel availability for transmissions. Because of the assumption that the allocated slots are contiguous and $P_k^j(i)$ is equal to 1 if $i \in \mathbf{Z_k^{body}}$, the transmission slots must include all slots in the body segment $\mathbf{Z_k^{body}}$. Note that a heavy traffic load may result in a high collision probability and low network utilization, even every link v_k only transmits $t_k(j)$ slots. To resolve the problem, a parameter, β, is introduced as a congestion control threshold, so the slots whose $P_k^j(i)$ is lower than β will not be selected for transmission. The higher value of β results in lower collision probability but lower channel utilization. Then, the on-the-go scheduling scheme tries to arrange the starting point of the transmission slots and the number of transmission slots of v_k denoted by s_k^* and t_k^* respectively, in each frame such that the sum of the $P_k^j(i)$ from s_k^* to $s_k^* + t_k^*$ is maximal. We formulate the problem as follows.

Collision Minimization Problem

Given: $t_k(j)$ and $P_k^j(i)$ for each slot i

Find: s_k^*, t_k^*

Maximize: $$\sum_{\Delta(s_k^*, i) < t_k^*} P_k^j(i)$$

Subject to: $P_k^j(i) \geq \beta, \text{if } \Delta(s_k^*, i) < t_k^*$

$t_k^* \leq t_k(j)$

Because the selected slots must be contiguous and the slots closer to body segment must have higher $P_k^j(i)$, we can iteratively take the slot with highest $P_k^j(i)$ from the unselected slots until a constraint is violated. The computation time is $O(n \log n)$, where n is the number of the slots allocated to the link v_k, for the sorting cost. Figure 7 shows an example of on-the-go scheduling. The network topology and the slot allocation of each link in this example are shown in Figures 3 and 6 respectively. We assume that the link v_5 does not have a bandwidth requirement in frames 1 and 2, but requires 5 slots in frame 3, i.e., $t_5(1) = 0$, $t_5(2) = 0$, and $t_5(3) = 5$. In frame 1, the first two slots in the head segment of v_5 are used by link v_3, and three slots in the tail segment of v_5 are used by link v_4. Hence, in frame 2, link v_5 updates the $P_5^2(i)$ values of those slots to 0.5 based on Equation (28) with $\alpha = 0.5$. Similarly, the $P_5^3(i)$ value of a slot i that has been used by other links in frame 2 becomes 0.25; otherwise, it becomes 0.75. Then, in frame 3, we choose four slots for link v_5 to transmit data where the sum of their $P_5^3(i)$ values is the maximum and the $P_5^3(i) \geq \beta$ for the four slots with $\beta = 0.3$.

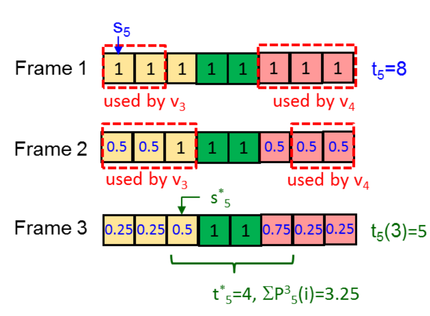

Fig. 7. An example of on-the-go scheduling for link v_5.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

3) Early start and early terminate policies: In the slot allocation and on-the-go scheduling stages, each link determines its transmission slots. When an un-scheduled slot of link v_k is detected as unused by any of its neighbor, the scheme implements an early start policy to utilize the unused slots. In other cases, if a multi-access slot of v_k conflicts with its neighbors, it may terminate its transmission early. The objective of incorporating early start and early terminate policies into the on-the-go scheduling scheme is to improve channel utilization and prevent transmission collisions. The two policies are based on the following properties of the on-the-go scheduling scheme.

Property 1: In on-the-go scheduling, if no neighbor of a link v_k transmits data in a slot $i \in \mathbf{Z_k^{head}}$, the slots after slot i in $\mathbf{Z_k^{head}}$ will not be used by any neighbor of link v_k.

Property 2: In on-the-go scheduling, if a slot $i \in \mathbf{Z_k^{tail}}$ is selected by a neighbor of v_k for transmission, the slots after slot i in the $\mathbf{Z_k^{tail}}$ will also be selected by the neighbor for transmissions.

Properties 1 and 2 can be derived from Theorem 2 and the on-the-go scheduling scheme. If slot i is in the head segment of link v_k, it must be in the tail segment or idle segment of v_k's neighbors. With the design of on-the-go scheduling scheme, any link $v_l \in IF(v_k)$ will not transmit any data after the slot i until its next head segment starts if the slot i is in the tail segment of v_l and has not been used by v_l. Therefore, Property 1 is verified. Similarly, if slot i is in the tail segment of v_k and there is a neighbor of v_k transmitting data in the slot, the slot must be in the head segment of the neighbor. Then, the neighbor will continues transmitting data in its residual head segment slots based on the on-the-go scheduling. Therefore, Property 2 is also verified.

The early start policy, which is based on Property 1, stipulates that any link can start transmissions if it detects an idle channel in a head segment slot before its scheduled transmission time. The early terminate policy is a collision avoidance mechanism that is based on Property 2. It stipulates that a link should stop all transmissions after a collision occurs in its tail segment until the next head segment begins. If a collision occurs in the tail segment of v_k, the residual slots in that segment will also fail because of Property 2. Therefore, we terminate the transmissions in those slots to reduce the probability of collisions between v_k and its neighbors.

In the on-the-go scheduling scheme, the early start and early terminate policies define three states for each link: a waiting state, a transmission state, and an idle state. Initially, all links are in the idle state and cannot transmit. Each link can only transmit data in its transmission state. A link will enter the waiting state when it starts its head segment. In the waiting state, the link will listen for the channel condition and enter the transmission state if the channel becomes idle during a head segment slot or slot s_k^* begins. Then, the link returns to the idle state (1) when the time slot $s_k^* + t_k^*$ begins, or (2) if a collision occurs during transmission. This mechanism enables each link v_k to obtain extra slots if there are no collisions. Under the early terminate policy, at most two collisions can occur in each frame for each link: one in slot s_k^* and the other in the slot that is being terminated early.

V. Performance Evaluation

In this section, we present the results of simulations conducted to evaluate the performance of the proposed 2SLS scheme. All the simulations were performed using the ns-2 network simulator [29] with TDMA enhancements in the MAC layer of IEEE 802.11. The performance metrics are the collision probability of the whole network, the average end-to-end delay, and the end-to-end throughput of a flow. To demonstrate the advantages of the 2SLS scheme, we compare its performance with those of the following transmission schemes: the IEEE 802.11 protocol, the throughput optimal algorithm [19], and the traditional one-stage TDMA-based slot allocation scheme. We use the slot allocation algorithm in [17] to implement the one-stage TDMA-based slot allocation scheme because it is designed for contiguous slot allocation with minimizing the end-to-end transmission delay for each flow. We also compared two other TDMA-based slot allocation algorithms, namely, the first-fit approach and the best-fit approach. Because their simulation results are close to those of the algorithm proposed in [17], we do not report them in this paper. Table I shows the system configuration used in the simulations, each of which ran for 3600 seconds. The length of each link was 200 meters in all scenarios. We set the control parameter $\alpha = 0.5$ in order to balance between the sensitivity $(\alpha \sim 1)$ and stability $(\beta \sim 0)$ on sensing of the channel condition. With $\alpha = 0.5$, we set the control parameter $\beta = 0.3$, so the collisions in a time slot will repeat at most one time, then its $P_k^j(i)$ will be smaller than β. In the following, we only show the results of chain and grid topologies to exemplify the performance of 2SLS, while the 2SLS scheme can be applied to any network topology as long as the central controller knows the interference set of all links and routing path of all flows.

TABLE I
The System Configuration Used in the Simulations

Paramter	Value
Channel Bandwidth	11Mbps
Interface Queue Size	100 packets
Maximum Packet Size	1500 bytes
Traffic Load of Flows	H.264 encoded VBR video films [24] with 630Kbps mean rate
Transmission Range	250m
Interference Range	420m
TDMA slot/frame duration	1.2ms/60ms (50slots per frame)
Control parameters	$\alpha = 0.5, \beta = 0.3$

A. Chain topology

In the first scenario, we evaluate the performance of 2SLS and the other schemes on a chain network topology with three video flows, as shown Figure 1. The traditional fixed TDMA-based slot allocation scheme can reserve 980Kbps of bandwidth at most for each flow; otherwise, it will not be schedulable. The proposed 2SLS can reserve a range of end-to-end bandwidth $(\phi_i^{min}, \phi_i^{max})$ equal to (630Kbps, 1.5Mbps), so each flow can access more bandwidth when its traffic load

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

is high and other flows have low traffic loads. Figures 8, 9, and 10 show the cumulative distribution frequency (CDF) curves of the end-to-end delays of all packets belonging to video flows 1, 2 and 3 respectively. Note that any packet that has been dropped or is delayed longer than 600ms will not been included in these figures. We observe that, in most cases, the end-to-end delays under 2SLS are shorter than those of the IEEE 802.11 protocol and the traditional fixed TDMA-based slot allocation scheme with 630Kbps (mean) and 980Kbps. If the end-to-end delay bound is 500ms for each flow, the good-put ratios (the ratio of packets whose end-to-end delay is shorter than 500ms) of video flows 1, 2 and 3 are 89.05%, 90.31% and 99.92% respectively. In this scenario, the optimal scheme has a longer average end-to-end delay than 2SLS on flow 3 because it allocates slots to flows 1 and 2 first on the link (n_e, n_f). Note that the minimum end-to-end delays of video flows 1 and 2 under 2SLS and the traditional fixed TDMA-based slot allocation scheme are 100ms and 60ms, respectively, because there are scheduling delays in frame-based slot allocation schemes [16]. The optimal scheme schedules slots one by one, so there is no scheduling delay.

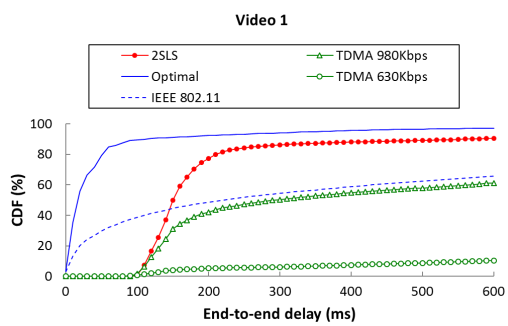

Fig. 8. The CDF of the end-to-end delays of Flow 1.

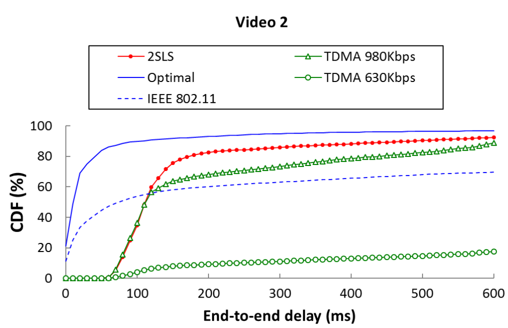

Fig. 9. The CDF of the end-to-end delays of Flow 2.

Table II shows the average end-to-end delay and packet dropping probability of each flow and the overall collision probability under the compared schemes. We observe that the TDMA-based slot allocation schemes, including 2SLS and the optimal scheme, are collision-free. 2SLS outperforms IEEE 802.11 and the traditional fixed TDMA-based slot allocation scheme. It also has a smaller dropping probability than the optimal scheme. This is because the optimal scheme assumes

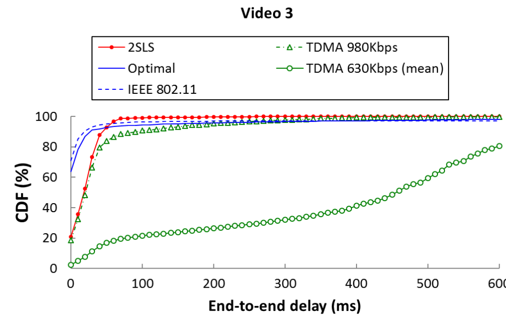

Fig. 10. The CDF of the end-to-end delays of Flow 3.

the queue length is infinite and it tends to serve the flow with the longest path first, which results in many packets being dropped by the link (n_e, n_f) due to queue overflow.

B. Grid topology

The second scenario is a wireless mesh network comprised of twenty-five wireless mesh nodes, as shown in Figure 11. There are four video flows with a mean transmission rate of 630Kbps. In this topology, each flow must compete with other flows for the channel resource in each hop. The traditional fixed TDMA-based scheme can allocate, at most, 1260Kbps of end-to-end bandwidth for each flow. In 2SLS, we set ϕ_i^{min} at the mean rate (630Kbps) and increase ϕ_i^{max} from the mean rate to 1760Kbps, which is the most end-to-end bandwidth 2SLS can access for each flow. The CDFs of the end-to-end delays of all packets belonging to video flows 1, 2, 3, and 4 are shown in Figure 12, 13, 14 and 15 respectively.

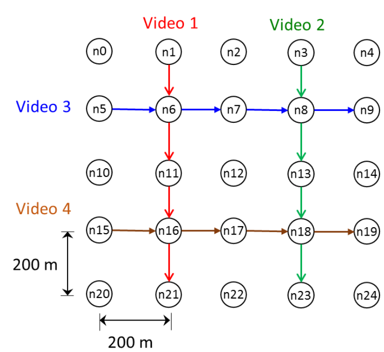

Fig. 11. The network topology in the simulation for grid topology.

In this simulation, the traffic load of video 2 contains many long-term bursts, so its packet dropping probability is higher under the traditional fixed TDMA-based slot allocation scheme. The 2SLS scheme can allocate more slots dynamically to video flow 2 when it contains bursty traffic, and the allocation will be maintained until the burst ends. Hence, 2SLS outperforms the traditional fixed TDMA-based slot allocation scheme on video 2. In contrast, the traditional TDMA scheme outperforms 2SLS on video flows 3 and 4 because they must compete with video flow 2 for the channel resource. Even so, 2SLS achieves better fairness between different flows. We use four performance metrics in this simulation: the end-to-end

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

TABLE II
COMPARISON OF THE PERFORMANCE OF DIFFERENT SCHEDULING SCHEMES ON THE CHAIN TOPOLOGY.

		2SLS	TDMA 980Kbps	TDMA 630Kbps	IEEE 802.11	Optimal
End-to-end delay	Video 1	243.04 ms	551.64 ms	1732.02 ms	469.56 ms	57.37 ms
	Video 2	183.13 ms	220.47 ms	941.41 ms	354.13 ms	47.01 ms
	Video 3	29.86 ms	47.26 ms	359.02 ms	10.26 ms	30.10 ms
Dropping probability	Video 1	0.64 %	4.08 %	21.84 %	7.83 %	2.20 %
	Video 2	1.82 %	4.44 %	20.43 %	10.04 %	2.57 %
	Video 3	0.08 %	0.49 %	12.20 %	3.09 %	0.88 %
Collision probability		0.00 %	0.00 %	0.00 %	8.96 %	0.00 %

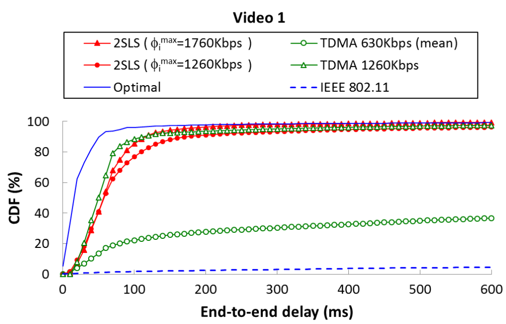

Fig. 12. The CDF of the end-to-end delays of Flow 1 in the grid topology.

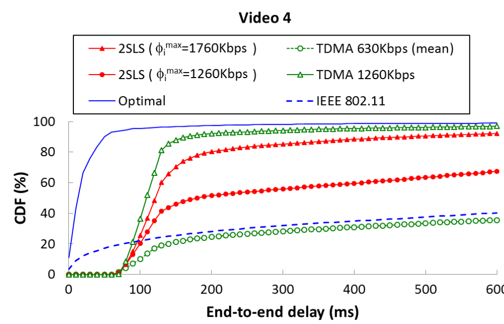

Fig. 15. The CDF of the end-to-end delays of Flow 4 in the grid topology.

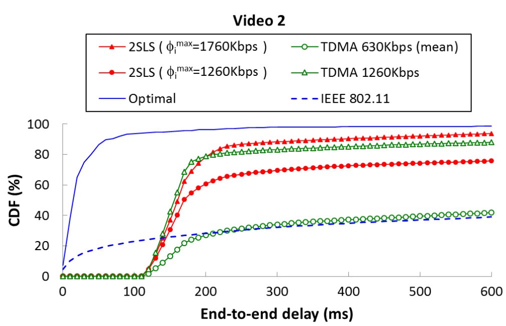

Fig. 13. The CDF of the end-to-end delays of Flow 2 in the grid topology.

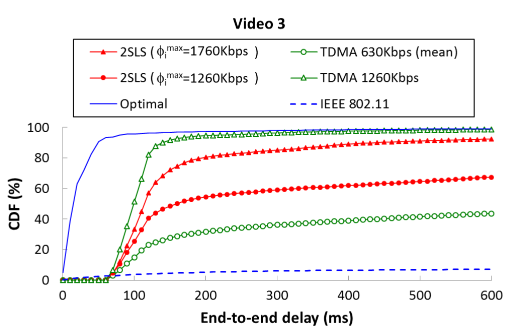

Fig. 14. The CDF of the end-to-end delays of Flow 3 in the grid topology.

delay, the dropping probability, the overall collision probability and the average available slot ratio z'_{avg} of different schemes (see Table III). The average available slot ratio z'_{avg} indicates the efficiency of channel usage, and is defined as follows:

$$z'_{avg} = \frac{\sum_{v_k \in V} |\mathbf{Z'_k}|}{M \cdot |V|}, \quad (29)$$

where $\mathbf{Z'_k} = \{i \,|\, i \notin \mathbf{Z_k} \text{ and } i \notin \mathbf{Z_l}, \forall l \in IF(v_k)\}$ is the set of slots that have not been allocated to any neighbor link of v_k. The value of Z'_k indicates the capability of link v_k to accept subsequent incoming flows. Here, we take the average ratio of Z'_k of all links to indicate the overall channel usage. A higher z'_{avg} ratio means the whole network has more unused channel resources that can be allocated to subsequent incoming flows. Note that the metric z'_{avg} cannot fully describe the channel usages of all links by itself. The distribution of channel usage can be derived from the channel usage map constructed from the Z'_k of all links.

Figure 16 shows the average available slot ratio z'_{avg} of 2SLS with different values of ϕ_i^{max}. The z'_{avg} keeps the same during some maximum demand rate intervals because the unit of channel resource allocation is slot. And the calculation of slot requirement $t_k(j)$, which is defined in Equation (3), is a step function of demand rate $r_k(j)$. We observe that the ratio decreases as ϕ_i^{max} increases. Under 2SLS, the average available slot ratio is 0.104 when ϕ_i^{max} is 1760Kbps, which is higher than that of TDMA (1260Kbps). However, 2SLS can provide more flexible bandwidth for all flows because each flow can access up to 1760Kbps of bandwidth if there is a burst in its traffic load.

In the final simulation, we evaluate the compared schemes on flows with different mean traffic rates in the grid topology. We selected four video flows from the video traces in [24]. Their mean rates are 630Kbps, 510Kbps, 340Kbps and 320Kbps respectively. For each flow, we set ϕ_i^{min} at the mean rate and ϕ_i^{max} at 3.4 and 2.5 times of ϕ_i^{min}, where $3.4 * \phi_i^{min}$ is the largest value of ϕ_i^{max} that 2SLS can handle. In the traditional fixed TDMA-based slot allocation scheme, the maximum bandwidth for each flow is 2.5 times of the mean rate. The simulation results, listed in Table IV, show

TABLE III
PERFORMANCE COMPARISON OF THE SCHEDULING SCHEMES IN THE GRID TOPOLOGY.

		2SLS ϕ_i^{max}**=1760Kbps**	**2SLS** ϕ_i^{max}**=1260Kbps**	**TDMA 1260Kbps**	**TDMA 630Kbps**	**IEEE 802.11**	**Optimal**
End-to-end delay	Video 1	82 ms	104 ms	88 ms	830 ms	3114 ms	40 ms
	Video 2	234 ms	431 ms	252 ms	830 ms	865 ms	41 ms
	Video 3	208 ms	471 ms	127 ms	795 ms	2938 ms	42 ms
	Video 4	207 ms	443 ms	149 ms	901 ms	842 ms	39 ms
Dropping probability	Video 1	0.08 %	1.45 %	0.43 %	10.84 %	43.39 %	0.40 %
	Video 2	0.52 %	3.18 %	1.80 %	14.09 %	22.07 %	1.07 %
	Video 3	0.57 %	2.83 %	0.12 %	9.44 %	40.68 %	0.41 %
	Video 4	0.84 %	2.95 %	0.27 %	11.46 %	17.57 %	0.60 %
Collision probability		1.01 %	1.42 %	0.00 %	0.00 %	24.19 %	0.00 %
Available slots ratio		0.104	0.251	0.102	0.480	N/A	N/A

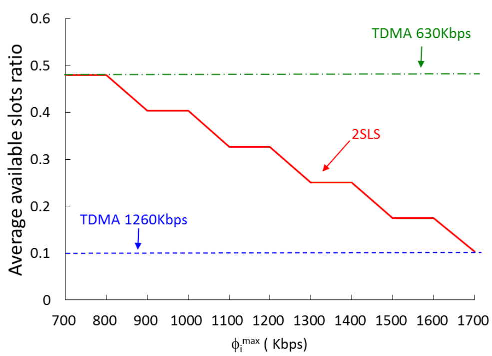

Fig. 16. The average available slot ratio of 2SLS under different values of ϕ_i^{max}.

that 2SLS performs well on flows with different mean rate traffics.

VI. CONCLUSION

We have proposed a 2-stage link scheduling scheme (2SLS) for TDMA-based wireless mesh networks to better utilize the wireless spectrum and still meet real-time application bandwidth needs. The scheme allocates a continuous region, divided into conflict-free and multi-access slots. We considered the continuity of allocated slots and the conditions that lead to transmission collisions under the scheme. The introduction of multi-access slots can accommodate the fluctuating bandwidth demands of real-time applications. To improve the transmission performance, the on-the-go scheduling mechanism selects the slots that are least likely to collide, without coordinating with other links. Two enhancements to the proposed 2SLS are designed to prevent transmission collisions. Our simulation results demonstrate that 2SLS achieves better channel utilization than the IEEE 802.11 MAC protocol, and it is more flexible and efficient than traditional TDMA-based mechanisms.

The 2SLS scheme only incurs a message overhead in the first stage, i.e., slot allocation. The size of the overhead depends on the inference model and the network topology. Moreover, the scheme only needs the information about k-hop neighbors; therefore, completing an allocation requires $4 * D^k$ messages at most in the k-hop interference model, where 4 is the number of parameters used to represent the allocation of a link, and D is the maximum degree of the network topology. The traditional static approaches, which find a global optimal solution without the restriction of using contiguous allocations, have message overheads of $O(M * N)$, where M is the frame size and N is the number of nodes in the whole network. The overhead is substantially larger than that of 2SLS. Furthermore, 2SLS does not need to exchange messages in the second stage. Thus, we can rapidly assign bandwidth to each link in order to meet frequently changing bandwidth requirements without any overhead. Note that, although 2SLS allocates slots to links, the mechanism and theory discussed in this study can be used to allocate slots to each flow. However, the overhead is considerably higher because the lifespan of flows is much shorter than that of links. Therefore, it is not advisable to allocate slots to flows.

REFERENCES

[1] Seattle wireless, (http://www.seattlewireless.net)
[2] Southampton Open Wireless Network (http://www.sown.org.uk)
[3] Rice TFA Network (http://tfa.rice.edu)
[4] Enhanced Generic Access Networks (EGAN) study (http://www.3gpp.org/ftp/Specs/html-info/43902.htm)
[5] 3GPP system to Wireless Local Area Network (WLAN) interworking; System description (http://www.3gpp.org/ftp/Specs/html-info/23234.htm)
[6] "IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999), pp.C1-1184.
[7] N. Nandiraju, et. al., Achieving fairness in wireless LANs by enhanced IEEE 802.11 DCF, in Proceedings of IEEE Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), June 2006.
[8] M. Garetto, T. Salonidis, and E. Knightly, "Modeling Per-flow Throughput And Capturing Starvation in CSMA Multi-hop Wireless Networks," in Proceedings of IEEE INFOCOM 2006, Barcelona, Spain, April 2006.
[9] J.-F. Lee, W. Liao, and M. C. Chen, "An Incentive-based Fairness Mechanism for Multi-hop Wireless Backhaul Networks with Selfish Nodes," in IEEE Transactions on Wireless Communications, Vol. 7, No. 2, Feb. 2008.
[10] "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems Amendment 1: Multiple Relay Specification," IEEE Std 802.16j-2009 (Amendment to IEEE Std 802.16-2009) , vol., no., pp.c1-290, June 2009.
[11] 3GPP TS 36.211, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation (Release 11), Sep. 2012. (http://www.3gpp.org/).
[12] "IEEE Draft STANDARD for Information Technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking," IEEE Unapproved Draft Std P802.11s/D6.0, June 2010.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2014.2358591, IEEE Transactions on Wireless Communications

TABLE IV
THE PERFORMANCE OF THE COMPARED SCHEMES ON FOUR FLOWS WITH DIFFERENT MEAN TRAFFIC RATES IN THE GRID TOPOLOGY.

		2SLS ϕ_i^{max}**=3.4***ϕ_i^{min}	**2SLS** ϕ_i^{max}**=2.5***ϕ_i^{min}	**TDMA 2.5*mean**	**TDMA mean**	**IEEE 802.11**	**Optimal**
End-to-end delay	Video 1	121 ms	156 ms	121 ms	889 ms	1329 ms	42 ms
	Video 2	123 ms	155 ms	138 ms	1477 ms	310 ms	33 ms
	Video 3	352 ms	400 ms	358 ms	3056 ms	2087 ms	48 ms
	Video 4	147 ms	191 ms	172 ms	3060 ms	246 ms	32 ms
Dropping probability	Video 1	0.00 %	0.21 %	0.18 %	10.86 %	21.84 %	0.35 %
	Video 2	0.00 %	0.42 %	0.42 %	11.99 %	9.93 %	0.09 %
	Video 3	1.73 %	3.42 %	3.42 %	40.73 %	28.41 %	0.36 %
	Video 4	0.00 %	0.06 %	0.05 %	26.13 %	9.50 %	0.08 %
Collision probability		0.06 %	0.12 %	0.00 %	0.00 %	21.42 %	0.00 %
Available slots ratio		0.111	0.215	0.199	0.612	N/A	N/A

[13] H. T. Cheng and W. Zhuang, Pareto Optimal Resource Management for Wireless Mesh Networks with QoS Assurance: Joint Node Clustering and Subcarrier Allocation, in IEEE Transactions on Wireless Communications, vol. 8, no. 3, March 2009.

[14] X.-Y. Li, A. Nusairat, Y. Wu, Y. Qi, J. Zhao, X. Chu, and Y. Liu, "Joint Throughput Optimization for Wireless Mesh Networks," in IEEE Transactions on Mobile Computing, vol. 8, no. 7, pp. 895-909, July 2009.

[15] B. Wang, and M. Mutka, Qos-Aware Fair Rate Allocation in Wireless Mesh Networks, in Elsevier Computer Communications, vol. 31, issue 7, pp. 1276-1289, May 2008.

[16] P. Djukic, and S. Valaee, Delay Aware Link Scheduling for Multi-Hop TDMA Wireless Networks, in IEEE/ACM Transactions on Networking, vol. 17, no. 3, June 2009.

[17] P. Cappanera, L. Lenzini, A. Lori, G. Stea and G. Vaglini, Optimal Link Scheduling for Real-time Traffic in Wireless Mesh Networks in Both Per-flow and Per-path Frameworks, in Proceedings of IEEE WoWMoM, 2010.

[18] B. Ji, C. Joo and N. B. Shroff, Delay-Based Back-Pressure Scheduling in Multi-Hop Wireless Networks, in Proceedings of IEEE INFOCOM, 2011.

[19] S. Zhao and X. Lin, On the Design of Scheduling Algorithms for End-to-End Backlog Minimization in Multi-hop Wireless Networks, in Proceedings of IEEE INFOCOM, 2012.

[20] A. Kabbani, T. Salonidis, and E. W. Knightly, Distributed Low-Complexity Maximum-Throughput Scheduling for Wireless Backhaul Networks, in Proceedings of IEEE INFOCOM 2007, Anchorage, AK, May 2007.

[21] C. Cicconetti, I. F. Akyildiz, and L. Lenzini, "FEBA: A Bandwidth Allocation Algorithm for Service Differentiation in IEEE 802.16 Mesh Networks," in IEEE Transactions on Networking, vol. 17, no. 3, pp. 884-897, June 2009.

[22] I. Rhee, A. Warrier, J. Min, and L. Xu, DRAND: Distributed Randomized TDMA Scheduling for Wireless Ad Hoc Networks, in IEEE Transactions on Mobile Computing, vol. 8, no. 10, Oct. 2009.

[23] Y. Hou, and K. K. Leung, A Distributed Scheduling Framework for Multi-User Diversity Gain and Quality of Service in Wireless Mesh Networks, in IEEE Transactions on Wireless Communications, vol. 8, no. 12, Dec. 2009.

[24] P. Seeling and M. Reisslein, Video Transport Evaluation With H.264 Video Traces, in IEEE Communications Surveys and Tutorials 14(4): 1142-1165, Fourth Quarter 2012.

[25] G. Sharma, R.R. Mazumdar and N.B. Shroff, On The Complexity of Scheduling in Wireless Networks, in Proceedings of ACM Mobicom 2006, pp. 227-238, 2006.

[26] P. Gupta and P. R. Kumar, The Capacity of Wireless Networks, in IEEE Transactions on Information Theory, vol. 46, no. 2, pp. 388-404, Mar. 2000.

[27] A. Vagish, T. Znati and R. Melhem, Per-Node Delay Assignment Strategies for Real-Time High Speed Network, in Proceedings of IEEE Globecom 1999, pp. 1323-1327, Dec. 1999.

[28] Y.-C. Tu, M. C. Chen, Y. S. Sun and W.-K. Shih, "Enhanced Bulk Scheduling for Supporting Delay Sensitive Streaming Applications," in Elseviser Computer Networks, Vol. 52, Issue 5, pp. 971-987, April 2008.

[29] Network Simulator ns-2, http://www.isi.edu/nsnam/ns/.

PLACE PHOTO HERE

Yung-Cheng Tu received the PhD degree from the department of Computer Science in National Tsing Hua University in 2007. He is currently a Post-doctoral Fellow of Institute of Information Science, Academia Sinica, Taiwan. His currently research interests include real-time scheduling, Quality of Service (QoS) control and wireless communication.

PLACE PHOTO HERE

Meng Chang Chen received the Ph.D. degree in Computer Science from the University of California, Los Angeles, in 1989. He was with AT&T Bell Labs from 1989 to 1992 and now he is a Research Fellow of Institute of Information Science, Academia Sinica, Taiwan. From 1999 to 2002 and from 2008 to 2010, he served as Deputy Director of the institute. From 2000 to 2003, he served as the chair of Standards and Technology Transfer committee of the National Science & Technology Program for Telecommunications Office (NTPO). His current research interests include wireless network, QoS networking, computer and network security, data and knowledge engineering, information retrieval and computer-assisted English learning.

PLACE PHOTO HERE

Yeali S. Sun received her B.S. from the Computer Science and Information Engineering department of National Taiwan University in 1982, and M.S. and Ph.D. degrees in Computer Science from the University of California, Los Angeles (UCLA) in 1984 and 1988, respectively. From 1988 to 1993, she was with Bell Communications Research Inc. In August 1993, she jointed National Taiwan University and is currently a professor of the Department of Information Management. Her research interests are in the area of Internet security and forensics, Quality of Service (QoS), cloud computing and services, and performance modeling and evaluation.