Fairness Provisioning in Multi-hop Wireless Backhaul Networks: A Dynamic Estimation Approach

JENG-FARN LEE AND YUAN-NENG HSU
Department of Computer Science and Information Engineering
National Chung Cheng University
Chiayi, 621 Taiwan
E-mail: jflee@cs.ccu.edu.tw

In this paper, we consider the fairness problem for Transit Access Points (TAP) in multi-hop wireless backhaul networks. Existing approaches have implementation issues, such as the requirement for modifications to the MAC layer or queuing operations of TAPs, limitations on the network topology, high packet loss rates and overheads, and difficulty in measuring the effective link capacities between TAPs. To address these issues, we propose an effective and practical approach that enforces fairness in wireless backhaul networks by estimating the target sending rates for TAPs dynamically. We also extend our approach to multi-rate, multi-weight, non-saturated and split wireless backhaul networks. Moreover, to speed up the convergence time, we provide a good estimation for our approach’s initial sending rate based on the network topology and the transport layer protocol. We evaluate the performance of proposed approach via ns-2 simulations. The results demonstrate that the approach works well in various network topologies and achieves a rapid convergence time.

Keywords: fairness, wireless backhaul networks, TAP, sending rate estimation, 802.11

1. INTRODUCTION

Broadband wireless technologies are promising platforms for accessing the Internet due to their characteristics of low cost, robustness, and ease of deployment. In recent years, a number of studies have extended the technologies and applications from traditional one-hop network access [1-4] to multi-hop communications, such as ad hoc networks, mesh networks, and backhaul networks [5-8]. A backhaul network is concerned with transporting traffic between distributed and centralized sites. More specifically, a multi-hop wireless backhaul network has the unique advantage of providing extended coverage to wireless areas or to areas with limited wired infrastructure. In a wireless backhaul network, traffic from mobile users to the wired Internet, and vice versa, is processed through multiple wireless Transit Access Points (TAPs) via a gateway, as illustrated in Fig. 1. The TAPs may be located in different independent entities, such as restaurants, small business offices, private residences, or hot spots. Point-to-multipoint microwave access technologies, such as IEEE 802.11 WiFi or IEEE 802.16 WiMAX can be used for wireless backhauling purposes. However, under current MAC operations, users located several hops away from the gateway experience low throughput, and even starvation in some cases.

Hence, there still a need to address the fairness problem in wireless multi-hop backhaul networks. Existing approaches have implementation issues, such as the requirement

Received November 10, 2009; revised February 12 & May 25, 2010; accepted July 6, 2010.
Communicated by Rong-Hong Jan.

* The primary version of this work was published at ICST Qshine, Hong Kong, Jul. 2008 by Jeng-Farn Lee and Yuan-Neng Hsu, “Fairness in Multi-hop Wireless Backhaul Networks: A Dynamic Estimation Approach”.

835
for modifications to the MAC layer or queuing operations of TAPs, limitations on the network topology, high packet loss rates and overheads, and difficulty in measuring the effective link capacities between TAPs. In this paper, we investigate existing fairness mechanisms and propose an effective and practical approach to enforce the fairness in wireless backhaul networks. We also evaluate the performance of proposed approach via ns-2 simulations. The results demonstrate that approach works well in various network topologies and achieves a rapid convergence time.

The remainder of this paper is organized as follows. In section 2, we discuss the fairness model suitable for wireless backhaul networks and review related works on fairness mechanisms. In section 3, we present our proposed fairness mechanism. We extend our approach to multi-rate, multi-weight wireless backhaul networks and networks in which some TAPs are not always saturated and targeted at the gateway. In section 4, we propose a good estimation for our approach’s initial sending rate based on the network topology and the transport layer protocol to speed up the convergence time. We evaluate the performance of the proposed approach via simulations based on ns-2 in section 5, and present our conclusions in section 6.

2. RELATED WORKS

In this section, we consider the fairness reference model and existing fairness mechanisms in multi-hop wireless backhaul networks. Although the mechanisms can ensure fairness among TAPs, they all have some implementation issues. We classify the mechanisms according to their design and the factors they control in order to ensure fairness provisioning in backhaul networks.

2.1 Fairness Reference Model

The fairness reference model proposed by Gambiroza et al. [9] has been widely adopted in multi-hop wireless backhaul networks. It is based on four constraints. First, the granularity for fairness is a TAP-aggregated flow. Each TAP’s egress traffic should be treated as a single aggregate flow, independent of the number of local micro-flows or
mobile devices associated with the TAP. Second, maximal spatial reuse must be ensured. Third, to avoid the IEEE 802.11 performance anomaly reported in [10], we use air time rather than throughput as the network resource to be shared fairly. Throughput-based allocation leads to serious performance degradation, since the station with the lowest channel quality or link capacity determines the throughput achievable by all stations. Finally, spatial bias must be eliminated so that TAPs located farther away from the gateway will not receive disproportionately less air time than TAPs close to the gateway. This property is essential for the deployment of multi-hop wireless backhaul networks because TAPs in different locations should not be penalized; in other words, the network’s performance should be distance-independent. TAP owners pay the same amount to maintain TAP operations irrespective of their location, so each TAP should share the network resource equally.

To determine the effects of each fairness constraint on the capacity of a multi-hop wireless backhaul network, Gambiroza et al. [9] propose a general model to compute the target end-to-end throughput of each TAP under different fairness objectives. However, they focus on backhaul networks with no spatial reuse, (i.e., only one link can be active at any given time). In [11], the authors present a model to achieve target throughputs for TAPs under the fairness constraints of a more general backhaul network, where spatial reuse is possible and some TAPs are not always backlogged.

### 2.2 Existing Fairness Mechanisms

The fairness mechanisms in [9, 12] control the sending rates of TAPs being their target throughputs, which are the desired end-to-end throughputs based on the four fairness constraints mentioned above. Therefore, each TAP needs to measure the offered load of its local traffic (i.e., the arrival rate of aggregate traffic from all local mobile users during a predefined measurement period), and the effective capacity of each link connecting to an adjacent TAP. Information about the offered load and the capacity of each link incident to each TAP is then exchanged between TAPs periodically. As a result, the end-to-end throughput for each TAP-aggregated flow with fairness constraints can be computed as a function of each TAP’s offered load. The advantage of controlling the sending rates of TAPs is that it is not necessary to modify the queuing, forwarding and contention operations of the TAPs because the finely-tuned sending rates eliminate contention behavior in the wireless network. However, the measured effective capacity must consider factors like the MAC layer overhead, the effect of wireless interference, the hidden terminal problem, and multi-rate multi-channel issues. It is therefore very difficult to measure the capacity in real networks. Moreover, the information exchanged between TAPs increases the load of wireless backhaul networks.

In contrast, the approaches proposed in [13, 14] modify the queuing, forwarding and contention behavior of TAPs to enforce the fairness model. The authors of [13] evaluate the fairness and throughput performance of various queuing schemes, and show that per-flow queuing (i.e., one queue for each TAP-aggregate flow) at intermediate TAPs is required to achieve fairness. In addition, to improve the throughput performance, TAPs that need to transmit more transit data should be allocated more bandwidth via MAC-layered QoS mechanisms. However, since these approaches do not control the sending rates of TAPs, more and more transit data will be dropped by TAPs closer to the gateway.
Thus, the bandwidth used by previous TAPs to transmit and relay the dropped frames was wasted. In addition, per-flow queuing requires more hardware and processing budget. Under the mechanism proposed in [14], there are only two queues at intermediate TAPs: one for local data from mobile users and the other for transit data. The authors investigate how MAC layer contention behavior (i.e., the probabilities that TAPs will access the wireless channel) and the forwarding probabilities of intermediate TAPs in the transit queue influence the network’s performance. They then derive the throughput and packet delay experienced by TAPs at different distances (i.e., hop counts) from the gateway. However, they assume that, on average, each TAP with a distance of \( x \)-hops to the gateway must be a relay for the same number of TAPs with \((x + 1)\)-hops to the gateway. Therefore, the wireless nodes in the network must be well located and the applied network topologies are limited. Fig. 2 shows a wireless network with 126 nodes that satisfies this assumption.

Another way to achieve fairness in a wireless backhaul network is to schedule the transmitting links in each time slot. For example, Fig. 3 shows a possible upstream scheduling

![Fig. 2. A 126-node network satisfying the assumption that each \( x \)-hop node needs to relay the same number of \((x + 1)\)-hop nodes.](image)

![Fig. 3. A possible TDMA scheduling scenario with spatial reuse for the upstream links in the backhaul network in Fig. 1. In this example, \( T = 20 \).](image)
scenario with spatial reuse that guarantees TAP-aggregate fairness in the backhaul network shown in Fig. 1. This approach has proved notoriously difficult to realize since even if global information, such as the network topology and link capacities, is available, the scheduling problem of finding the optimal link transmission set in each time slot is NP-hard [15, 16]. Although [17-19] propose low-complexity or distributed mechanisms, the link schedule needs to be re-calculated when the network topology or the traffic loads of mobile clients change. Moreover, these approaches can only be used in wireless networks that adopt TDMA-based MAC protocols, such as IEEE 802.16 WiMAX networks.

To overcome the limitations or drawbacks of the above mechanisms, we propose a practical approach for achieving fairness in wireless backhaul networks. The basic idea is to control the sending rates of TAPs to their target throughputs under the fairness constraints. However, unlike the approaches in [9, 12], we do not try to measure the effective link capacities of TAPs because it is difficult to do so in real networks. Instead, the proposed approach estimates the target throughput of each TAP in the backhaul network dynamically with a high degree of accuracy and low convergence time.

3. TAP-AGGREGATE FAIRNESS MECHANISM

In this work, we consider static, cooperative TAPs in wireless multi-hop backhaul networks. We assume that data will not be exchanged between TAPs, and data sent from one TAP will not be split among different gateways at intermediate TAPs. All TAPs are always backlogged and have the same link capacity as their adjacent TAPs. However, we relax these constraints in order to extend our approach in section 4. We also assume that the gateway can (1) measure the average throughput of each TAP-aggregate flow for the predefined measurement period; (2) calculate the sending rate of each TAP in the next measurement period based on our algorithm; and (3) distribute this information to TAPs at the beginning of the next time period. When the backhaul network achieves fairness, the gateway stops the above operations. If the operations of the gateway are difficult to modify, each TAP can calculate the average throughput in the previous measurement period, and distribute the information to other TAPs in a similar way to that in the Open Shortest Path First (OSPF) protocol. The exchanged information can also be piggybacked with the routing information exchanged among TAPs to reduce the overhead for data exchange. Then, in our approach, each TAP can calculate its sending rate for next time period based on the average throughput of other TAPs.

We propose a heuristic approach, called Search of Sending-rates and Min Throughput (SSMT), to estimate the proper sending rates of TAPs based on the average throughput of TAPs in the previous measurement period. In this approach, the initial sending rate of each TAP is 500kbps, which is chosen arbitrarily, and the rate is increased by αkbps if all TAPs can achieve a throughput similar to the sending rate in the initial state. The value of the initial sending rate has a significant influence on the convergence time because, if it is not close to the target rate, SSMT would need more rounds to achieve fairness. Thus, in section 4, we propose a good estimation of the initial sending rate of our approach based on the network topology and the transport layer protocol. If there is unfairness between the TAPs, the estimation process goes into the adaptation state. The SSMT decides the new sending rate based on the sending rate and the minimum through-
put of TAPs in the previous round. Since we assume all links to have the same capacity in this section, the temporal fairness is equal to the throughput fairness. Thus, we assume that each TAP has the same throughput (i.e., it is fair for all TAPs) if the difference ratio between the maximum and minimum throughput of the current round is less than $\beta$ of the maximum throughput. SSMT terminates if the backhaul network is fair. In this work, we nominate a small value, such as 3%, for $\beta$ in order to demonstrate that our approach achieves high accuracy and converges rapidly.

An intuitive way to estimate the sending rates of TAPs in a backhaul network is to conduct a binary search of the sending rates in previous rounds. Although this strategy can eventually find the approximate target sending rates of TAPs for the fairness reference model, it may take several rounds to converge. Moreover, the sending rates of TAPs may fluctuate dramatically in different rounds since the rates may increase or decrease in the adaptation state. To address these problems, SSMT speeds up the convergence time of the search process and provides stable sending rates for TAPs in different rounds. SSMT tries to estimate the sending rate for a backhaul network by searching the sending rate and minimum throughput in the previous round based on the concept that the target sending rate should be between the sending rate and minimum throughput if there is unfairness between TAPs. Let $\text{min}_r$ and $\text{max}_r$ denote, respectively, the minimum and maximum throughput of TAPs in the previous round. The initial sending rate is 500kbps, and it is increased by $\alpha$ if there is no unfairness between the TAPs (i.e., $\text{max}_r - \text{min}_r < \beta \cdot \text{max}_r$) because the backhaul network can afford to transmit more traffic. Otherwise, SSMT goes into the adaptation state and takes the mean value of the old rate (i.e., the sending rate in the previous round) and the new rate as the new sending rate. Fig. 4 details the steps of the SSMT algorithm. Note that the new sending rate in the adaptation state will always be lower than the old sending rate; hence, the sending rates of SSMT in different rounds will not fluctuate like the rates in a binary search approach.

```plaintext
1  flag = 0; old_rate = 500;
2  for each round {
   // initial state
3    if (flag == 0 && (max_r - min_r) < $\beta \cdot \text{max}_r$)
4       new_rate = old_rate + $\alpha$;
5    else {
   // adaptation state
6       new_rate = (old_rate + min_r)/2; flag = 1;
7       if ((max_r - min_r) < $\beta \cdot \text{max}_r$)
8          break;
9       old_rate = new_rate;
```

Fig. 4. The SSMT algorithm.

There is a trade-off between correctness and the convergence time when determining the size of the measurement period. A measurement period that is too small may result in incorrect estimations due to the unstable performance of the backhaul network. On the other hand, a measurement period can yield the correct result, but the convergence time is too long. Generally, when the number of TAPs, the traffic load, the number of hops...
or contention in the interference range increases, SSMT needs a larger measurement period to derive the correct estimation. According to the simulations on different network topologies (discussed in section 5), 1,000 milliseconds is sufficient for SSMT to achieve correctness.

4. ENHANCED SSMT FOR GENERAL NETWORK MODELS

In this section, we extend SSMT to multi-rate, multi-weight wireless backhaul networks, and backhaul networks in which some TAPs are not always saturated and targeted at the gateway. We also propose a good estimation for the initial sending rate of SSMT based on the network topology and the transport layer protocol to speed up the convergence time.

4.1 Multi-rate Multi-weight Backhaul Networks

First, we extend SSMT to multi-rate, multi-weight backhaul networks in which the TAPs have different link capacities and weights for sharing the wireless bandwidth. In networks with different link capacities, the target throughputs for TAPs are proportional to their link capacities, since each TAP has the same transmission time based on the fairness reference model defined in [9]. The priority or weight of a TAP’s aggregate flow can be determined according to the amount the TAP pays or the QoS requirements under different service/business models. However, a high priority service far away from the gateway causes service degradation for TAPs along the transmission path to the gateway because much more of the wireless resource is required to relay the traffic of high priority service. Combined with the concept of weighted fairness, we can use the normalized throughput defined in Eq. (1) to check whether the backhaul network is fair (i.e., TAPs should have the same normalized throughput if the network is fair).

\[ \text{nor\_throughput}_i = \frac{\text{real\_throughput}_i}{C_i \times W_i}, \]  

where \( C_i \), \( W_i \), and \( \text{real\_throughput}_i \) are the link capacity, weight, and the real average throughput of TAP\( i \), respectively. Then SSMT determines the values of max\( \_r \), min\( \_r \) according to the normalized throughputs of the TAPs (e.g., the value of min\( \_r \) is the minimum normalized throughput of the TAPs). SSMT also determines the new sending rate (i.e., new\_rate) for next round as the mean value of the old sending rate and min\( \_r \). Consequently, each TAP can use the sending rate defined in Eq. (2) to obtain its sending rate (new\_rate) for the next round.

\[ \text{new\_rate}_i = \text{new\_rate} \times C_i \times W_i \]  

4.2 Split and Non_Saturated Backhaul Networks

Next, we consider the case where some TAPs are not targeted at the gateway and saturated. In a split backhaul network, data may be exchanged between TAPs, and the
TAPs are not always targeted at the gateway, as illustrated in Fig. 5. Let \( N_i \) be the number of aggregate flows that originate from TAP\(_i\) (i.e., the values of \( N_0, N_1, \) and \( N_2 \) in Fig. 5 are 2, 1, and 1 respectively). With the temporal fairness constraint, the aggregate flows from the same TAP are all assigned the same amount of time; thus, they have the same throughput because they have the same first hop link, irrespective of the hop-length of the aggregate flow. Hence, the sending rate of each aggregate flow from TAP\(_i\) is the new sending rate derived in Eq. (2) divided by \( N_i \):

\[
\text{new rate}_i = \frac{\text{new rate}_i}{N_i}.
\]

(3)

When TAPs are non-saturated, they do not always have data to send; thus, their demand is less than their target throughputs and they should be excluded from the set when checking for fairness. Let \( S(t) \) denote the saturated TAPs at time \( t \). Our approach only needs to check the fairness for TAPs belonging to \( S(t) \) based on their normalized throughput; and only saturated TAPs use the suggested sending rates in Eq. (3) to estimate their target sending rates. The modified SSMT discussed in this section and section 4.1 is shown in Fig. 6. The additional codes are marked in boldface.

```plaintext
1  flag = 0; old_rate = 500;
2  for each TAP {new_rate = old_rate * C_i * W_i/N_i}
3  for each round {for i \in S(t)
4    nor_throughput = real_throughput * N_i/(C_i * W_i)
5    determine the values of max_r and min_r based on nor_throughput;}
6  // initial state
7  if (flag == 0 && (max_r - min_r) < \beta * max_r)
8    new_rate = old_rate + \alpha;
9  else
10    // adaptation state
11    new_rate = (old_rate + min_r)/2; flag = 1;
12    if ((max_r - min_r) < \beta * max_r)
13      for i \in S(t) {new_rate = new_rate * C_i * W_i/N_i}
14      break;
15    for i \in S(t) {new_rate = new_rate * C_i * W_i/N_i}
16    old_rate = new_rate;
```

Fig. 6. The extended SSMT algorithm.
4.3 Estimating the Initial Sending Rate

Recall that we fixed the initial sending rate of SSMT at 500kbps in section 3. However, the rate should be different with the network topology, packet size, and transport layer protocol for efficiently estimating the target sending rate. For example, the initial sending rate for a backhaul network with only 3 TAPs should be much larger than that of a network with 10 TAPs. A bad initial sending rate results in many unnecessary attempts which slow down the convergence time. Therefore, we propose a good estimation of the initial sending rate based on the network topology, packet size and the transport layer protocol to speed up the convergence time. To obtain a good initial sending rate, we can derive the time share of each TAP under the fairness constraints based on the network topology, and the effective link capacity under the frame size and the transport layer protocol.

First, we adapt the model proposed in [11] for multi-rate, multi-weight backhaul networks to derive the shared transmission time for each TAP under the fairness constraints. We consider a network with $N$ TAPs and thus $N$ TAP-aggregated flows. Each aggregate flow $f$ traverses a pre-determined route $R_f$. Let $t_{l_f}$ denote the time share or the time needed to relay the data for flow $f$ on link $l$, $l \in R_f$. To maximize spatial reuse and link utilization, the time share assigned to flow $f$ on each link of route $R_f$ must satisfy the flow preservation property; that is, the time share for flow $f$ on each outbound link must be equal to the time share required to forward packets of flow $f$ on any of the previous links. Hence, we have

$$t_{l_i}^{f} C_i = t_{l_j}^{f} C_j, \forall i, j \in R_f.$$  \hspace{1cm} (4)$$

Moreover, to achieve the weighted fair requirement and avoid spatial bias for flows traversing a different number of hops, we have

$$t_{l_i}^{f} / W_f = t_{l_i}^{g} / W_g, \text{ for all flows } f \text{ and } g,$$  \hspace{1cm} (5)$$

where $l_{f_1}^{f}$ denotes the first link of flow $f$ on route $R_f$. Let flow $f_{\text{minW}}$ be the aggregate flow from the TAP with the minimum weight in the backhaul network and let its time share be $t_u$. Then, based on Eq. (5), we get

$$t_{l_i}^{f} = \frac{W_f}{W_{f_{\text{minW}}}} t_u, \text{ for all flows } f.$$  \hspace{1cm} (6)$$

To satisfy the flow preservation property for flow $f$ on its route to the gateway according to Eq. (4), we have

$$t_{l_i}^{f} = \frac{W_f}{W_{f_{\text{minW}}}} t_u \cdot \frac{C_i}{C_l} t_{l_i}^{f}, \text{ for all links } l \in R_f.$$  \hspace{1cm} (7)$$

Let $CL_i$ be the links in the interference range of link $i$. Then, we can find the bottleneck link...
\(i\), which is the link with the maximum value of \(\sum_{f \in C_F} \sum_{l \in C_L} t^f_l\), and solve the following equation to obtain the value of \(t_u\):

\[\sum_{f \in C_F} \sum_{l \in C_L} t^f_l = 1, \quad (8)\]

where \(C_{F_i}\) is the set of flows traversing the interference links for link \(i\) (i.e., \(C_L\)). Finally, the time share for \(TAP_i\) is given by \(t_u\). For example, in the multi-rate, multi-weight wireless backhaul network shown in Fig. 7, \(t_u\) would be the time share for the aggregate flow of \(TAP_0\) because it has the minimum weight in the network. Table 1 shows the time required by each link to transmit the data of the aggregate flow \(f\) according to Eqs. (5) and (7). Then, we find that \(t_u\) is equal to 1/14 according to Eq. (8). Therefore, in Fig. 7, the target transmission time for \(TAP_0, TAP_1\), and \(TAP_2\) under the fairness constraints is 1/14, 2/14, and 4/14 respectively.

Table 1. The transmission time required for flows in the backhaul network in Fig. 7.

<table>
<thead>
<tr>
<th>Flows</th>
<th>Links</th>
<th>(C_0)</th>
<th>(C_1)</th>
<th>(C_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow 0</td>
<td>(t_u)</td>
<td>4(t_u)</td>
<td>2(t_u)</td>
<td></td>
</tr>
<tr>
<td>Flow 1</td>
<td>2(t_u)</td>
<td>(t_u)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flow 2</td>
<td></td>
<td></td>
<td>4(t_u)</td>
<td></td>
</tr>
</tbody>
</table>

![Fig. 7. An illustration of the shared transmission time of TAPs for multi-rate, multi-weight, backhaul networks.](image)

Next, we consider the case some TAPs are not saturated in a non-saturated backhaul networks. Let \(TAP_i\) non-saturated and let its sending rate be \(\lambda_i\). Consequently, for a non-saturated flow originating from \(TAP_i\), the link share time \(t^f_l, \forall l \in R_{TAP_i}\) in Eqs. (4), (5), and (8) is replaced by \(t^f_l = \lambda_i/C_l\) in order to distribute the extra bandwidth to saturated TAPs in the backhaul network. Note that this calculation may be repeated several times because, after receiving the extra bandwidth, the resulting target throughputs for some TAPs may exceed their offered loads.

Next, we derive the effective link capacity based on the packet size and the transport layer protocol. As mentioned earlier, the effective link capacity must account for factors like the MAC layer overhead, the effect of wireless interference, the hidden terminal problem, and the contention discrepancies between wireless nodes. Therefore, it is difficult to determine the explicit value. In the following, we derive an upper bound for the effective link capacity by only considering the transmission overhead on the MAC and transport
layers without any collisions and idle slots. However, the estimated upper bound should be close to the explicit value, since the fine-tuned sending rate can reduce the number of contention collisions caused by over-sending traffic in wireless backhaul networks.

Let $EC_i$ be the estimated effective link capacity of link $i$, $L_{data}$ be the packet size, and $L_{M-data}$ be the frame size including the MAC layer header. Then, the value of $EC_i$ can be defined as

$$EC_i = \frac{t_{tr \cdot data} \times L_{data}}{T \times L_{M-data}} \times C_i,$$  \hspace{1cm} (9)$$

where $t_{tr \cdot data}$ is the frame transmission time of the data frame; $T$ is the total time required for successful transmission of the data frame, which includes $t_{tr}$, DIFS, SIFS, the Physical Layer Convergence Protocol preamble and header transmission time, and the MAC layer acknowledgment transmission time. $T$ needs to include the time for RTS/CTS handshake if RTS/CTS is used.

However, the $EC_i$ derived by Eq. (9) is only suitable for UDP. It is not appropriate for TCP because the receiver needs to send a TCP ACK packet back to the sender. Therefore, the $EC_i$ for TCP should be modified as follows,

$$EC_{TCP_i} = \frac{t_{tr \cdot data} \times L_{data}}{T_{data + ack} \times L_{M-data}} \times C_i,$$  \hspace{1cm} (10)$$

where $T_{data}$ and $T_{ack}$ are the total times required for successful transmission of a TCP’s data and TCP ACK frames respectively. However, as the network data is not comprised of pure UDP or TCP traffic, so we let the $EC_i$ in Eq. (9) be $EC_{UDP_i}$. Then, the realistic $EC_i$ can be calculated as follows,

$$EC_i = (r_{TCP} \times EC_{TCP_i} + (1 - r_{TCP}) \times EC_{UDP_i}) \times C_i,$$  \hspace{1cm} (11)$$

where $r_{TCP}$ is the ratio of TCP traffic that can be known from the statistical data.

Finally, we can use $t_{tr \cdot EC_i}$ as the initial sending rate for the aggregate flow $f$ and skip the initial state of SSMT to speed up the convergence time. Although we only derive the effective capacity based on the transmission overhead of the MAC and transport layers, without considering the complex analytical models in the vast amount of literature on 802.11 MAC and TCP performance, the estimated initial sending rate clearly speeds up the convergence time. We validate this point in section 5.

5. PERFORMANCE EVALUATION

In this section, we evaluate the performance of SSMT via ns-2 simulations. We do not provide performance comparison with existing mechanisms since these existing mechanisms can all provide good fairness basically. However, as mentioned in section 2.2, they all suffer from implementation issues which are not easily demonstrated via simulations. In the following simulations, the wireless link rate is set at 11Mbps and each traffic flow is generated as CBR UDP traffic with a fixed packet size of 1000 bytes, including the IP header. The MAC protocol used in the simulations is IEEE 802.11 DCF.
without RTS/CTS; the duration of each measurement period is one second; and the value of $\beta$ is 3%. We first show that SSMT is effective for different wireless backhaul networks with the same link capacity and saturated TAPs. Then, we extend the network topology to multi-rate multi-weight, split and non-saturated backhaul networks. Finally, we demonstrate the efficacy of our initial rate estimation strategy.

We consider the operations and performance of SSMT in three simulation scenarios: (1) the “Parking Lot” wireless backhaul network (Fig. 8 (a)); (2) a network with 8 TAPs and one gateway located on the edge of the topology (Fig. 9 (a)); and (3) a network with 8 TAPs, but the gateway is located in the center of the topology (Fig. 10 (a)). Figs. 8 (b) to 10 (b) plot the average throughput of TAPs for SSMT, and Table 2 details the sending rates of SSMT for different rounds of measurement under the “Parking Lot” scenario. Due to space limitations, we do not show the detailed sending rates for the other scenarios. In the second round shown in Table 2, the difference between max$_r$ and min$_r$ (i.e., 976-859) is more than 3% ($\beta$) of max$_r$ such that SSMT goes into the adaptation state. Therefore, the new sending rate for TAPs in the next round is the mean of send$_r$ and min$_r$ (i.e., $(1000 + 859)/2$). In the fourth round, the difference between the maximal and minimal average throughput is less than 3%, which means that the backhaul network is fair in this situation. As a result, the SSMT algorithm terminates, as shown in line 7 of Fig. 4. The real target sending rates of the three scenarios are 913kbps, 232kbps, and

<table>
<thead>
<tr>
<th>Round</th>
<th>TAP0 Sending Rate</th>
<th>TAP1 Sending Rate</th>
<th>TAP2 Sending Rate</th>
<th>Next Sending Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>500</td>
<td>492</td>
<td>484</td>
<td>500 + 500 = 1000</td>
</tr>
<tr>
<td>2</td>
<td>1000</td>
<td>867</td>
<td>859</td>
<td>(1000 + 859)/2 = 930</td>
</tr>
<tr>
<td>3</td>
<td>930</td>
<td>882</td>
<td>882</td>
<td>(930 + 882)/2 = 906</td>
</tr>
<tr>
<td>4</td>
<td>906</td>
<td>890</td>
<td>890</td>
<td>(906 + 890)/2 = 898</td>
</tr>
</tbody>
</table>

Table 2. The sending rates of SSMT in scenario (1).
haul network, even the value of $\beta$ is small (i.e., 3%). Therefore, SSMT is also suitable for backhaul networks with dynamic traffic loads, as well as dynamic or mobile TAPs, since its sending rates in the adaptation state are stable and it converges rapidly. When the number of contending TAPs on some links in a backhaul network is large, the network performance becomes quite unstable, so it is difficult for the gateway to measure the throughput of each TAP precisely. Even so, the following simulation shows that SSMT still works well in such a network topology. In Fig. 11 (a), seven TAPs contend for the wireless link to the gateway. The throughput performance of SSMT is shown in Fig. 11 (b). Clearly, SSMT also works well in the dense topology. Next, we show the sending rates during different measurement rounds of the SSMT estimation process in Figs. 8 (c) to 11 (c) as a function of the observation window size in different scenarios. The results demonstrate that a larger observation window nearly always yields more accurate sending rates; and 1,000 milliseconds is sufficient for SSMT to estimate a good sending rate to ensure fairness, even in a dense wireless backhaul network, as illustrated in Fig. 11 (a).

![Diagram](image_url)

(a) The network topology for scenario (2).
(b) Throughput performance of SSMT for scenario (2).
(c) The SSMT estimation processes for different-size observation windows in scenario (2).

Fig. 9. Scenario (2): eight TAPs with the gateway located at the border.
In the following, we evaluate our enhanced approach for the non-saturated, multi-rate multi-weight and split backhaul networks. We first change the sending rate of TAP\(_0\) to 200kbps at time 11 in scenario (1) to simulate a non-saturated backhaul network. Fig. 12 plots the average throughput of TAPs for SSMT. The simulation results of the first ten rounds in Fig. 12 are the same as those in Fig. 8 (b). At the end of the 11th round, SSMT returns to the initial state because TAP\(_0\) reduces its sending rate to 200kbps. SSMT therefore excludes TAP\(_0\) from the saturated set (i.e., \(S(t)\)) when calculating the values of \(\max_r\) and \(\min_r\). The results show that, under this approach, the remaining saturated TAPs share the excess bandwidth equally.

We then consider the multi-rate, multi-weight, backhaul network illustrated in Fig. 7. The link capacities \(C_0\), \(C_1\), and \(C_2\) are 20Mbps, 5Mbps, and 10Mbps respectively; and the weights of TAP\(_0\), TAP\(_1\), and TAP\(_2\) are 1, 2, and 4 respectively. Consequently, the adjustment ratios for the normalized throughputs (i.e., \(C_i * W_i\)) of TAP\(_0\), TAP\(_1\), and TAP\(_2\) in SSMT are 2, 1, and 4 respectively. Fig. 13 plots the average throughput of TAPs for SSMT, and shows that each TAP can achieve its target throughput according to its link capacity and weight in the network.

Next, we consider a backhaul network in which traffic may be exchanged between TAPs, as shown in Fig. 5. The link capacities \(C_0\), \(C_1\), and \(C_2\) are 20Mbps, 5Mbps, and
Fig. 11. Scenario (4): a wireless backhaul network with 7 TAPs contending for one wireless link.

(b) Throughput performance of SSMT in scenario (4).

(c) The SSMT estimation processes for different-size observation window sizes in scenario (4).

Fig. 12. The throughput of SSMT for the network topology in Fig. 8 (a) with a non-saturated TAP0.

Fig. 13. The throughput of SSMT for the wireless backhaul network in Fig. 7.
10Mbps respectively, and all the TAPs have the same weight. \( TAP_0 \) consists of two egress aggregate flows, \( i.e., \) flow \((0, g)\) and flow \((0, 1)\). Fig. 14 plots the average throughput of TAPs for SSMT. Our approach uses the aggregate throughput from each TAP \( i.e., \) the total throughputs of flow \((0, g)\) and flow \((0, 1)\) represents the aggregate throughput of \( TAP_0 \) to calculate the TAP’s normalized throughput based on Eq. (1). Then, each TAP uses the new \( \text{rate}_i \) to assign the sending rates for its egress flow equally according to Eq. (3). Therefore, as shown in Fig. 14, the weights for the target throughput of TAPs under the fairness constraints and those for flow \((0, g)\), flow \((0, 1)\), \( TAP_1 \), and \( TAP_2 \) are 4:1:2 and 2:2:1:2 respectively.

Finally, we demonstrate the effectiveness of our method for estimating the initial sending rate in SSMT. We first show that an unsuitable initial sending rate results in many unnecessary tries, resulting in long convergence time for the estimation process. We consider the Parking Lot scenario \( i.e., \) Fig. 8 (a) with 100kbps as the values of the initial sending rate and \( \alpha \). Fig. 15 plots the average throughput of TAPs for SSMT. We observe that SSMT spends about 10 rounds in the initial state, but it only stays in the adaptation

---

**Fig. 14.** The throughput of SSMT for the wireless backhaul network in Fig. 5.

**Fig. 15.** The throughput of SSMT for the network topology in Fig. 8 (a) with 100kbps as the values of the initial sending rate and \( \alpha \).

**Fig. 16.** The throughput of SSMT for the network topology in Fig. 8 (a) under estimated initial sending rate.

**Fig. 17.** The throughput of the SSMT network topology in Fig. 8 (a) for TCP traffic.
state for 2-3 rounds. Therefore, finding a way to reduce the number of rounds spent in the initial state is critical for the estimation process. We derive a good estimation of the initial sending rate by finding the time share of each TAP according to the network topology and an upper bound for the effective link capacity based on the frame size and transport layer protocol. Based on our approach, the estimated initial sending rate with a frame size of 1000 bytes is 1066kbps \((i.e., \frac{6400}{6}, \text{ where } 1/6 \text{ is the time share of each TAP and } 6400 \text{kbps is the effective capacity with a frame size 1000 bytes for UDP})\) for the network topology shown as Fig. 8 (a). Then, we can eliminate the initial states of SSMT to speed up the convergence time. Moreover, as shown in Fig. 16, the estimated initial sending rate is close to the target throughput for the backhaul network so that SSMT only needs 1-2 rounds in the adaptation state before convergence.

Finally, we evaluate our approach on TCP traffic. The value of \(EC_{TCP}\) with a frame size of 1000 bytes is 5.5Mbps for TCP traffic according to Eq. (10). Consequently, the initial sending is 916kbps \((i.e., \frac{5500}{6})\). Fig. 17 plots the throughput for SSMT. We observe that the target throughput for TCP traffic is much less than that for UDP traffic as TCP needs a TCP ACK for successful transmission. Moreover, our approach with good estimated initial rate still works well so that SSMT only needs 3 rounds to converge.

6. CONCLUSION

In this work, we address the problem of fairness between TAPs in multi-hop wireless backhaul networks. We propose an effective and practical approach called SSMT to enforce fairness in wireless backhaul networks. Specifically, we control the sending rates of TAPs to their target throughputs under the fairness constraints such that our approach does not need to modify the MAC layer and queuing operations of the TAPs. SSMT estimates the target throughput of each TAP efficiently, instead of measuring the link capacities of TAPs. Furthermore, we extend our approach to more general backhaul networks, such as multi-rate, multi-weight, non-saturated and split wireless backhaul networks. We also provide good estimation of the initial sending rate based on the network topology, packet size and transport layer protocol to speed up the convergence time. Performance evaluations based on ns-2 simulations demonstrate that SSMT can accurately estimate the target sending rates of TAPs. Moreover, SSMT converges rapidly and its sending rates in different rounds are stable, so it is also suitable for backhaul networks with dynamic traffic loads or mobile TAPs. Finally, SSMT can be used for both contention-based and TDMA MAC layer protocols such as IEEE 802.11 Wi-Fi and IEEE 802.16 WiMAX wireless networks.

REFERENCES


**Jeng-Farn Lee** (李正帆) received the B.S. and M.S. degrees in the Department of Information Management from National Taiwan University, Taiwan, in 1998 and 2000 respectively, and the Ph.D degree in the Department of Electrical Engineering from National Taiwan University, Taiwan, in Jan. 2007. He was a Post-doctoral fellow in the Institute of Information Science, Academia Sinica, Taiwan until July 2007, and joined Department of Computer Science and Information Engineering, National Chung Cheng University as an Assistant Professor in Aug. 2007. His current research interests include QoS networking, scheduling, and wireless access network.

**Yuan-Neng Hsu** (許淵能) received the B.S. degree in the Department of Computer Science and Information Engineering, Chung Hua University, Taiwan in 2007, and the M.S. degree in the Department of Computer Science and Information Engineering, National Chung Cheng University, Taiwan in 2009. He was an engineer in the Department of System Software of Unihan Corporation, Taiwan since March 2009.