PAPER

Cooperation between channel Access control and TCP Rate Adaptation in Multi-hop Ad hoc Networks

Pham Thanh GIANG[†a)], *Nonmember and* **Kenji NAKAGAWA**[††b)], *Member*

SUMMARY In this paper, we propose a new cross-layer scheme *Cooperation between channel Access control and TCP Rate Adaptation* (CATRA) aiming to manage TCP flow contention in multi-hop ad hoc network. In CATRA scheme, we collect useful information from MAC and physical layers to estimate channel utilization of the station. Based on this information, we adjust *Contention Window* (*CW*) size to control the contention between stations. Then, we can achieve the fair channel access of each station and the efficient spatial channel usage. Moreover, the fair value of bandwidth allocation for each flow is calculated and sent to the Transport layer. Then, we adjust the sending rate of TCP flow to solve the contention between flows and the throughput of each flow becomes fairer. The performance of CATRA scheme is examined on various multi-hop network topologies by using *Network Simulator* (NS-2).
key words: *Cross-layer method, per-flow/per-station fairness, flow/station contention, throughput, TCP rate control, channel access control, IEEE 802.11 back-off algorithm, multi-hop ad hoc.*

1. Introduction

Binary Exponential Back-off (BEB) mechanism [1] in IEEE 802.11 [2] seems to provide all the contending stations the same opportunity of access to the shared channel. However, in multi-hop topologies, BEB mechanism often suffers from the unfairness problem and low throughput [3]. Moreover, BEB mechanism determines the *CW* size corresponding only to the congestion condition, so it does not consider other conditions about neighbouring stations or higher layer, e.g., the number of flows in the channel or the number of users in the system. Thus, the *CW* size after some congestion may not be the optimal value for fairness.

TCP employs a window-based *Additive Increase Multiplicative Decrease* (AIMD) congestion control scheme to adjust the transmission rate [4]. Thus TCP performance in multi-hop wireless network depends critically on the congestion window in use. If the window grows too large, there are too many packets to compete for the same medium. That increases the network congestion and degrades the throughput and fairness performance [5], [6]. In this paper, we propose a new cross-layer method for determining better *CW* size in the MAC layer and TCP rate to achieve TCP fairness performance in multi-hop ad hoc networks.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 describes our cross-layer scheme. Section 4 evaluates our CATRA method. Finally, Section 5 concludes the paper.

2. Related Work

TCP challenges in 802.11 ad-hoc networks has been deeply investigated in the past several years as the report in [8]. Many factors may cause losses and affect TCP performance in multi-hop wireless networks such as: route failures caused by node mobility, random wireless loss, medium access contention. To improve the network performance, some studies tried to solve the problems at each layer independently as a layered design method, and some studies tried to cooperate some layers to exchange important information as a cross-layer design method.

In the layered design method, some studies focused on the malfunctions of IEEE 802.11 MAC layer. Li [9] investigated *Extended Inter-Frame Spacing* (EIFS) problem, i.e., the fixed EIFS value leads to unfair bandwidth allocation for each station. They proposed flexible EIFS values based on a measurement of the length of *Sensing Range* (SR) frame. However, the length of SR frame cannot be always recognized because of the spatial reuse of the bandwidth. The *three-pair* problem is introduced by Chaudet [10]. The analysis of the *three-pair* problem in [10] is based on Markov chain and gives some accurate results.

Some other studies considered to modify the BEB mechanism to improve performance of IEEE 802.11. Fang [11] reviewed unfairness problem in the BEB mechanism. Each station estimates the channel utilization of itself and that of neighboring stations then adjusts the back-off algorithm to give each station some value of bandwidth, which they call the *statistical fair access*. The statistical fair access must be predefined, but it is difficult in general because the value depends on the topology. There are some other studies on the control of *CW* size to achieve per-flow fairness [12], [13], but almost all of them consider only single-hop networks.

The others focused on the wrong behavior of TCP mechanism in multi-hop ad hoc networks. Xu [14] pro-

[†]Institute of Information Technology, Vietnam Academy of Science and Technology
[††]Department of Electrical Engineering, Nagaoka University of Technology
a) E-mail: ptgiang@ioit.ac.vn
b) E-mail: nakagawa@nagaokaut.ac.jp

posed the *Neighborhood RED* (NRED) scheme on network layer to enhance TCP fairness. By considering neighborhood channel usage, intermediate nodes detect neighborhood congestion and drop packets via NRED scheme according to flow's channel usage. Chen [15] and Fu [6] showed that the conventional TCP window grows too large, so there are too many packets competing for the channel. Their studies showed that there is the optimal value of TCP rate at which the throughput is maximum by improving spatial channel reuse. Their idea is adjusting TCP window size to achieve the optimal value.

Recently, some studies applied cross-layer design method to improve TCP performance. They tried to investigate information from MAC layer to adjust TCP rate to achieve a good TCP performance in terms of fairness and throughput. Cheng [16] proposed TCP-CL, which slightly modifies the legacy IEEE 802.11 MAC and TCP protocols. The standard IEEE 802.11 MAC layer provides a reliable operation over the communication channel by defining a retry limit parameter (RETL). Based on RETL value, link failure is reported to the TCP layer. Nevertheless, if the link experiences a high degree of contention, the MAC layer may mistakenly infer a link failure. Nahm [7] proposed a fractional window increment scheme for TCP (TCP-FEW) to prevent unnecessary network contention by limiting the growth rate of TCP's congestion window. Zhang [17] and Natalizio [18] considered to control the transmission rate of TCP by utilizing the MAC information by a cross-layer method. In [17], TCP adjusts the transmission rate by determining a better congestion window value based on the real MAC channel efficiency. However, they did not focus on the wireless MAC layer's contention and also can not exactly identify which part of MAC channel efficiency in the intermediate node.

In this paper, we propose a new cross-layer scheme and focus on TCP performance in multi-hop wireless network. First, we solve the unfairness problem at MAC layer by determining a suitable CW size in IEEE 802.11 for each station. Second, we control TCP sending rate to achieve fair throughput for each flow.

3. Cooperation between channel Access control and TCP Rate Adaptation

In multi-hop ad hoc networks, the unfairness problem in TCP traffic is due to both a wrong behavior of Distributed Coordination Function (DCF) mechanism in the channel access mechanism and a wrong behavior of TCP mechanism in traffic rate control. The wrong behavior of DCF mechanism makes it difficult for some stations to access the channel. In this paper, we will propose a new cross-layer scheme *Cooperation between channel Access control and TCP Rate Adaptation* (CATRA). In CATRA scheme, we collect useful information from MAC and physical layers, then adjust the CW size based on these information. By using this flexible CW size in the back-off state, the behavior of DCF mechanism will be improved and disadvantaged stations will get more chance to access the channel.

In general, the wrong behavior of TCP mechanism gives some TCP flows very large bandwidth, while it gives other flows a few bandwidth. We will propose a new method to investigate information from MAC and physical layers to adjust TCP rate dynamically. Thus the improvement of the behavior of TCP mechanism will achieve the fair share for each flow.

3.1 Channel Access control

Binary Exponential Back-off (BEB) mechanism [1] does not necessarily ensure the per-station fairness in multi-hop ad hoc networks [3]. Here, the per-station fairness means fair bandwidth allocation to every station. Moreover, the per-station fairness is not necessarily good for the per-flow fairness. For ideal per-flow fairness, we must investigate the number of flows which are transmitted from the station.

We define the ideal per-flow fairness as that all flows get the same throughput. The *Fair Bandwidth Ratio* (FBR) for per-station fairness, which we call FBR_S is defined by

$$FBR_S = \frac{n_{SEND}}{n_{total}}, \tag{1}$$

where n_{SEND} is the number of flows which are transmitted from the station. We call such flows as SEND flows. n_{total} is n_{SEND} plus the number of flows which use the channel with SEND flows.

The SEND flows at the station include the direct flow, which is generated by that station, and forwarding flows, which are required to forward from the neighboring stations. The flows are identified by MAC and IP addresses of both source and destination by decoding the header of packets. We can easily get n_{SEND} from MAC layer.

n_{total} is the total of the number of two kinds of flows at the examined station's view. The first kind of flows, denoted as TX flows, includes all flows from stations in the transmission range of the examined station and also flows generated from the examined station itself. The number of TX flows in the transmission range, denoted as n_{TX}, also can be known from MAC layer. The second kind of flows, denoted as CS flows, include flows from stations which are out of the transmission range but in the carrier sensing range of the examined station. We cannot distinguish the CS flows by the local information of the examined station. However, the station can sense the existence of CS flows by the physical power. We consider the number of CS flows which share the channel with the examined station as one if there exist some CS flows, so we define

$$n_{CS} = \begin{cases} 0, & \text{if there is no CS flow,} \\ 1, & \text{if there are one or more CS flows,} \end{cases} \quad (2)$$

and define the total number of flows n_{total} as

$$n_{total} = n_{TX} + n_{CS}. \quad (3)$$

In complex models, we cannot exactly identify the value of n_{total} due to the lack of information of hidden stations. In that case, the assumption of $n_{CS} = 1$ makes the value of FBR_S not accurate, but by adding $n_{CS} = 1$ to n_{TX} we can aware that some hidden stations require sharing the channel.

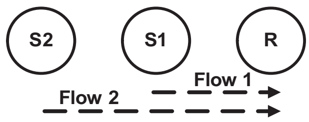

Fig. 1 A basic multi-hop wireless network model

Algorithm 1 Active time estimation

Initialization:
 $\overline{T}_{Active} = 0$
 $t = 0$
Begin
 for each packet p **do**
  if $p \to destID == localID$ **then**
   if ($p \to MacHeader \to Type == ACK$) **and** ($p \to TcpHeader \to Type == TCP_{DATA}$) **then**
    $t = t + 0.5 * CW * ST + T_{RTS} + SIFS + T_{CTS} + SIFS + T_{TCP_{DATA}} + SIFS + T_{ACK} + DIFS$
   else if ($p \to MacHeader \to Type == DATA$) **and** ($p \to TcpHeader \to Type == TCP_{ACK}$) **then**
    $t = t + 0.5 * CW * ST + T_{RTS} + SIFS + T_{CTS} + SIFS + T_{TCP_{ACK}} + SIFS + T_{ACK} + DIFS$
   end if
  end if
 end for
 for each interval time EP **do**
  $\overline{T}_{Active} = 0.8 * \overline{T}_{Active} + 0.2 * t$
  $t = 0$
 end for
End

As an example of a basic multi-hop wireless network model, see Fig. 1. Stations S1 and S2 are in a transmission range, and also stations S1 and R are in another transmission range. Stations S2 and R are out of the transmission range but in carrier-sensing range. The values of n_{SEND}, n_{TX}, n_{CS}, n_{total} and FBR_S for each station will be shown in Table 1.

Table 1 An example of Fair Bandwidth Ratio

	n_{SEND}	n_{TX}	n_{CS}	n_{total}	FBR_S
Station S1	2	3	0	3	2/3
Station S2	1	3	0	3	1/3
Station R	0	2	1	3	0

We also can obtain the *Real Bandwidth Ratio* (RBR) of the examined station, which we call RBR_S. The RBR_S is calculated by measuring the $\overline{T}_{Active}$ of the station in a predefined *Estimation Period* (EP). The $\overline{T}_{Active}$ of the station is defined as the average time for transmitting packets from the station during each EP. Algorithm 1 shows how to estimate the $\overline{T}_{Active}$ by sensing packets.

The RBR_S is defined as the ratio of the $\overline{T}_{Active}$ to the EP, i.e.,

$$RBR_S = \frac{\overline{T}_{Active}}{EP}. \quad (4)$$

Based on the original CW value of the back-off algorithm of the IEEE 802.11 and the collected information above, we will determine a better CW size in the back-off state. We propose a new CW size as follows;

$$CW' = \min\left(\frac{RBR_S}{FBR_S} CW, CW_{max}\right). \quad (5)$$

In (5), the FBR_S value is used as a threshold of channel access priority. If the station recognizes that its RBR_S is smaller than its FBR_S, CW' is smaller than the original CW. Then, the station can increase its chance to access the channel and then the bandwidth allocation will be increased. On the other hand, if the station recognizes that its RBR_S is lager than its FBR_S, CW' is larger than CW. Thus, the station decreases its chance to access channel, so other disadvantaged stations will get more chance to access the channel. The channel bandwidth is used more effectively and the fair bandwidth allocation between stations is improved.

In our mechanism, the disadvantaged stations have more chance to access channel. Throughput and fairness are often trade-off performances. However, in our proposed method, even in case that, some stations require only small offered load flows, they will access channel easily, and the remaining bandwidth is shared by other stations. Thus, the throughput performance is not degraded.

3.2 TCP Rate Adaptation

The channel access control mechanism in Sec. 3.1 can improve the Fair Bandwidth Ratio between stations but it does not necessarily ensure the per-flow fairness. In this subsection we will propose an algorithm to achieve the per-flow fairness by controlling the TCP rate. In TCP mechanism, the congestion window often grows too large in ad hoc networks [5], [6], and there are too many packets to compete for the same medium, then the throughput and fairness performance are degraded. Our idea to achieve fair share bandwidth allocation for flows is adjusting TCP rate. Let us consider the TCP flow generated from the examined station and control the rate of the flow by collecting information from MAC and physical layers. Remember we defined n_{total} in (3), which is the number of flows sharing with the TCP flow. Thus, we define the Fair Bandwidth Ratio FBR_f to the TCP flow as

$$FBR_f = \frac{1}{n_{total}}, \tag{6}$$

where the subscript f stands for $f$low.

The Real Bandwidth Ratio RBR_f for the TCP flow is defined as

$$RBR_f = \frac{\overline{T}_f^{Active}}{EP}, \tag{7}$$

where the $\overline{T}_f^{Active}$ is the average time for transmitting packets of the TCP flow at the examined station during EP.

If the TCP flow has more ratio of bandwidth than (6), then we will reduce the bandwidth by giving some delay before generating new packets. In TCP mechanism, at a certain time, the number of packets which are ready to send in a TCP flow is

$$win = cwnd + highest\_ack - cur\_seqno, \tag{8}$$

where $cwnd$ is TCP congestion window size, $highest\_ack$ is the highest sequence number of received TCP ACK packet, cur_seqno is the last sequence number of sent TCP DATA packet. Assuming the TCP flow achieves the Fair Bandwidth Ratio FBR_f of (6), the time T_f for transmitting win packets is calculated as

$$\begin{aligned} T_f &= \frac{win * \overline{s}_f}{FBR_f * B} = \frac{n_{total} * win * \overline{s}_f}{B} \\ &= n_{total} * win * \overline{T}_f^{tr}, \end{aligned} \tag{9}$$

where $\overline{s}_f$ is the average packet size and $\overline{T}_f^{tr}$ is the average time for transmitting one packet by the link. $\overline{T}_f^{tr}$ is calculated as

$$\overline{T}_f^{tr} = \frac{\overline{T}_f^{Active}}{N_f}, \tag{10}$$

where N_f is the number of packets from the TCP flow which are transmitted in EP. $\overline{T}_f^{Active}$ and N_f can be easily obtained from MAC layer by monitoring the number of packets from the TCP flow in an interval time EP.

Then, if the TCP flow has more than the fair bandwidth ratio (6), we will give a delay Δ_f defined by

$$\Delta_f = \frac{RBR_f}{FBR_f} T_f \tag{11}$$

to the TCP flow before generating new packets.

Let us consider, for example, the Flow 1 in Fig. 1. The examined station is S1. In the transmission range of S1, there are 3 flows, i.e, Flow 1, Flow 2 (S2-S1) and Flow 2 (S1-R). So, we have $n_{total} = 3$, hence $FBR_f = 1/3$ by (6). Suppose $RBR_f = 1/2$, i.e., Flow 1 obtained more than the Fair Bandwidth Ratio $FBR_f = 1/3$.

Then, in order to achieve the fairness, we will reduce the obtained bandwidth by giving the delay

$$\Delta_f = \frac{RBR_f}{FBR_f} T_f = \frac{3}{2} T_f = \frac{9}{2} * win * \overline{T}_f^{tr}. \tag{12}$$

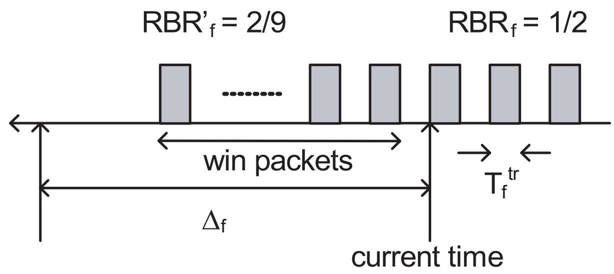

Fig. 2 TCP rate adaptation of the Flow 1

Therefore only win packets will be sent in Δ_f, the current bandwidth ratio $RBR_f = 1/2$ is reduced to

$$RBR'_f = \frac{win * \overline{T}_f^{tr}}{\Delta_f} = \frac{2}{9}, \tag{13}$$

thus we can achieve the fairness approximately by

$$\begin{aligned} \frac{1}{2}(RBR_f + RBR'_f) &= \frac{1}{2}(\frac{1}{2} + \frac{2}{9}) = \frac{13}{36} \\ &\approx \frac{1}{3} = FBR_f. \end{aligned} \tag{14}$$

Algorithm 2 TCP Rate Adaptation

Begin
  for each new received ACK packet **do**
    if $\frac{RBR_f}{FBR_f} > High_{th}$ **then**
      $\Delta_f = \frac{RBR_f}{FBR_f} T_f$ {Give the delay Δ_f before generating the next packet after the completion of win packets transmission.}
      $cwnd = max(cwnd - 1, 1)$
    else if $\frac{RBR_f}{FBR_f} < Low_{th}$ **then**
      $\Delta_f = 0$
      $cwnd = cwnd + 1$
    else
      Call original TCP mechanism
    end if
  end for
End

Algorithm 2 shows our modification in TCP mechanism. When $\frac{RBR_f}{FBR_f} > High_{th}$ with $High_{th} \geq 1$, TCP sending rate will be decreased by letting the TCP flow delay a longer time than their required time to transmit next generating packets and slightly reduce *cwnd*. When $\frac{RBR_f}{FBR_f} < Low_{th}$ with $Low_{th} \leq 1$, TCP sending rate will be increased by letting the TCP flow generate next packet without delay and have larger *cwnd*. We let TCP operate as original TCP mechanism in the other case. Thus, per-flow fairness is achieved by Algorithm 2.

The small delay time Δ_f before generating new packets will give chance for the flows of neighbouring stations can increase their throughput. Therefore, the difference of total throughput is negligible.

4. Performance Evaluation

We now evaluate the performance of our cross-layer scheme by comparing with the original scheme on various asymmetric topologies of multi-hop wireless ad hoc networks. We use *Network Simulator* (NS-2) [19] for evaluation. The simulation parameters are shown in Table 2. In CATRA scheme, we set a high threshold as $High_{th} = 1.05$, a low threshold as $Low_{th} = 0.7$ and estimation period as $EP = 2[s]$ for all simulations.

Table 2 Parameters in the simulation

Channel data rate	11[Mbps]
Antenna type	Omni direction
Radio Propagation	Two-ray ground
Transmission range	250[m]
Carrier Sensing range	550[m]
MAC protocol	IEEE 802.11b (RTS/CTS is enable)
Contention Window	$CW_{min} = 32$, $CW_{max} = 1024$
Connection type	TCP/FTP
Buffer size	100[packet]
Packet size	1[KB]
Simulation time	300[s]

4.1 Scenario-1: Long vs. Short hop flows topology

This topology includes a station chain as Fig. 3. Distance between each station is 200 [m]. In chain topology, let the second and the last stations S1, S2 generate TCP traffic to the station R. Because the long hop flow must travel through many relay stations, and it must contend with the short hop flow, it is difficult for Flow 2 to reach the destination.

Fig. 3 Scenario-1: Long vs. Short hop flows

The simulation results in terms of throughput are shown in Fig. 4, where n is the number of stations in Fig. 3 including R, S1 and S2, TCP means original scheme and CATRA is our proposed method. Also in Fig. 4, Flow 1 and Flow 2 denote the throughput of Flow 1 and Flow 2, respectively, and $Total$ is sum of throughputs which are received at all stations. They include destination and relay stations. If no packet is dropped in Scenario-1, we have:

$$Total \approx Flow1 + (n-1)Flow2. \tag{15}$$

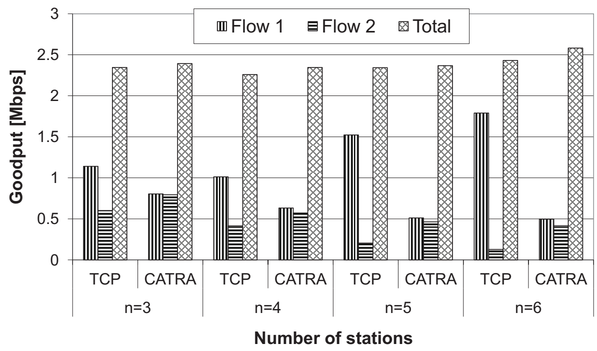

Fig. 4 Throughput in Scenario-1

Fig. 4 shows that the larger n, the more difficult the long hop flow Flow 2 reaches the destination in original scheme. While it gives fair throughput for long and short hop flows in CATRA scheme. Flow 2 in CATRA scheme achieves much better throughput than the original scheme and we still keep good total throughput of all flows in the topology.

4.2 Scenario-2: The *three-pair* topology

Fig. 5 shows the topology of Scenario-2. The problem in this scenario is also known as *three-pair* problem which was first investigated in [10]. In this scenario, stations S1 and S3 are out of the carrier sensing range, hence the two external pairs S1-R1 and S3-R3 are completely independent, i.e., they can send packets simultaneously without interference to each other. Thus, two external pairs contend bandwidth only with the central pair S2-R2, while the central pair contends with both external pairs. In this topology, the central pair cannot access the medium in the original scheme.

In Fig. 6, throughput of Flow 2 of the central pair is zero in the original scheme. In CATRA scheme, the MAC layer contention is solved hence the bandwidth allocation of S2 is increased. Thus, throughput of Flow 2 is much improved. The total throughput in CATRA scheme is smaller than the original scheme due to the channel reuse of the original scheme. If the central pair cannot access the channel bandwidth in the original scheme, then two external pairs can access channel independently and use the whole channel capacity. Thus,

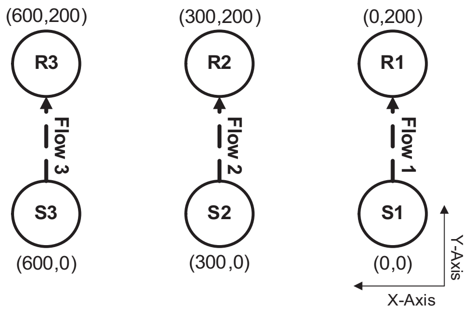

Fig. 5 Scenario-2: *Three-pair* problem

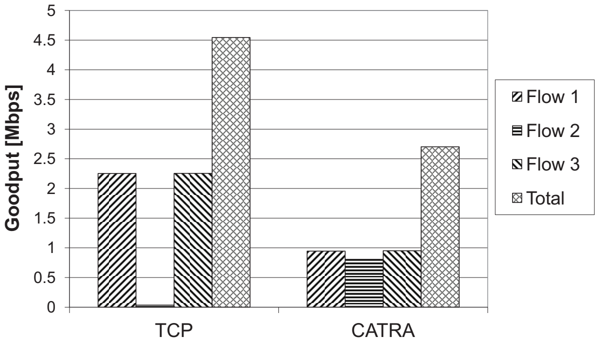

Fig. 6 Throughput in Scenario-2

the total throughput can be twice of the channel capacity. While in CATRA scheme, two external pairs must share the channel with the central pair. Thus, the central pair achieves a half of the channel capacity, then the two external pairs also have only a half of the channel capacity. Then, the maximum total throughput is one and a half of the channel capacity.

4.3 Scenario-3: The cross-chain topology

The cross-chain topology, which is discussed in [20], is shown in Fig. 7. We will examine the contention between two flows that leads to short-term unfairness problem. In this topology, if one flow wins to occupy the channel, it will have an opportunity to get the whole bandwidth for long time until timeout by the intra-flow contention. Then the other flow is difficult in accessing the channel. In this simulation, we let Flow 1 be more advantageous than Flow 2 by starting Flow 2 in one second later than Flow 1.

The simulation result is shown in Fig. 8. In the original TCP scheme, the throughput of Flow 1 and Flow 2 are much unstable while in CATRA scheme the short-term fairness between two flows is quite better. Moreover, we still keep good result of the sum of two flows' throughput in CATRA scheme.

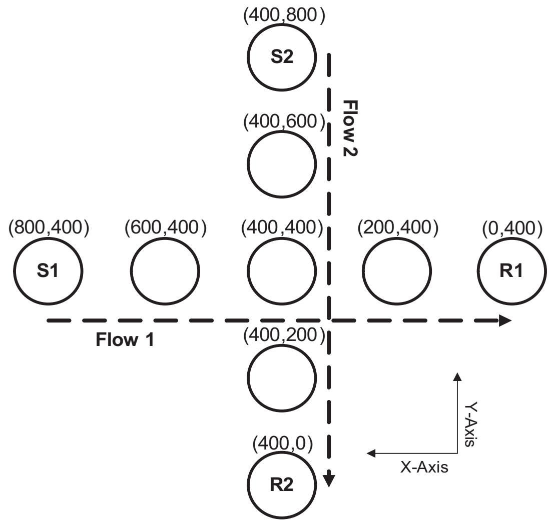

Fig. 7 Scenario-3: The cross-chain topology

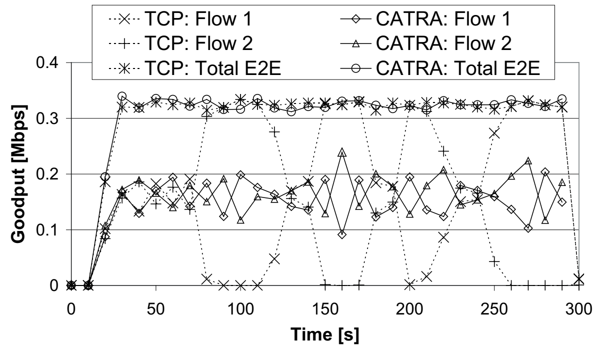

Fig. 8 Throughput in Scenario-3

4.4 Scenario-4: A TCP flow contents with a various speed UDP flow

In Scenario-4, the throughput of flows are compared between the original and our CATRA schemes. The topology is shown in Fig. 9. One UDP flow contends with one TCP flow. We evaluate the throughput of the flows by changing the offered load of the UDP flow from 0 to 6[Mbps].

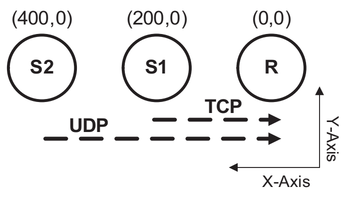

Fig. 9 Scenario-4: TCP vs. UDP flow

In the original scheme, let us call the TCP and UDP flows as "original TCP flow" and "original UDP flow", respectively. Similarly, in CATRA scheme, we call them as "CATRA TCP flow" and "CATRA UDP flow", respectively. In CATRA scheme, a UDP flow is applied the channel access control in Sec. 3.1, but not applied the TCP rate control in Sec. 3.2.

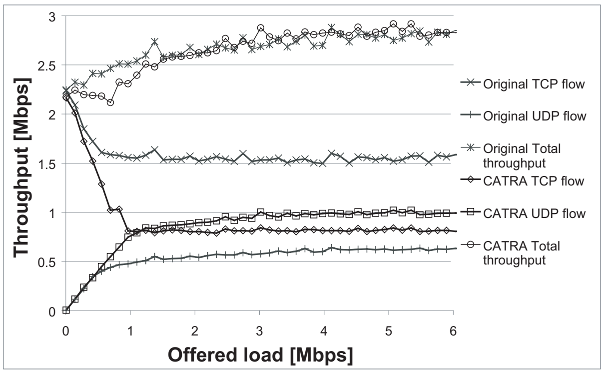

Fig. 10 Throughput in Scenario-4

The results of the throughput of each flow and the total throughput are shown in Fig. 10. For offered loads of UDP larger than 1[Mbps], the throughput of the original TCP flow is about 1.5[Mbps] and that of the original UDP flow is 0.5[Mbps], which means very unfair. While in CATRA scheme, the throughput of CATRA TCP and CATRA UDP flows are almost the same. In the original scheme, at the station S1, the sending bandwidth is much more than the receiving bandwidth, so the throughput of the original TCP flow is much larger than that of the original UDP flow. While in our CATRA scheme, due to the channel access control for both UDP and TCP flows and the TCP rate control for TCP flow, the unfairness of the original scheme was successfully dissolved.

As for the total throughput, both original and CATRA schemes achieve almost the same values. We can see that the throughput of the CATRA TCP is slightly smaller than that of the CATRA UDP flow because of the overheads of TCP mechanism such as ACK packets, and so on.

4.5 Scenario-5: The random topology

Scenario-5 is a random topology. We make topologies with 50 stations at random position in $1000[m] \times 1000[m]$ area. Among those 50 stations, n stations are chosen randomly and these n stations generate TCP traffic to one destination station. The average of Fairness Index and total end-to-end throughput are used as the metrics to compare the throughput and fairness performances, respectively. These terms of the network performance are examined versus the number of flows. Each data point is the average over 20 simulations.

The simulation result in Fairness Index is shown in Fig. 11. In there, we use Fairness Index as defined by [21] as follows;

$$\text{Fairness Index} = \frac{(\sum_{i=1}^{n} x_i)^2}{n \cdot \sum_{i=1}^{n} x_i^2}, \tag{16}$$

where n is the number of flows, x_i is the end-to-end throughput of $flow\ i$.

As Fig. 11, CATRA scheme achieves much better fairness performance than the original scheme.

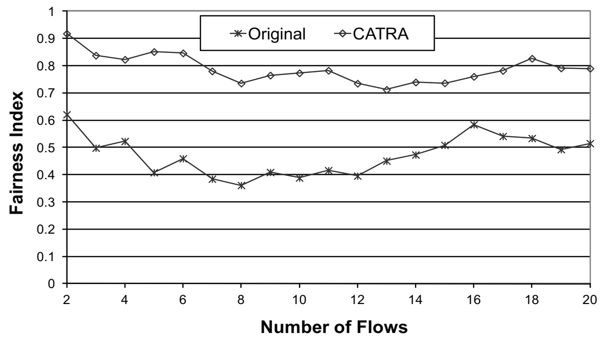

Fig. 11 Fairness Index in Scenario-5

The total end-to-end throughput is sum of all throughputs which are received at the destination station and Total throughput is sum of throughputs which are received at all stations. As Fig. 12, our total end-to-end throughput is smaller than the original scheme. The degradation of end-to-end throughput is due to sharing channel between each flow and also its forwarding flow. However, regardless of the reason of channel reuse, our total throughput Fig. 13 is similar to the original one.

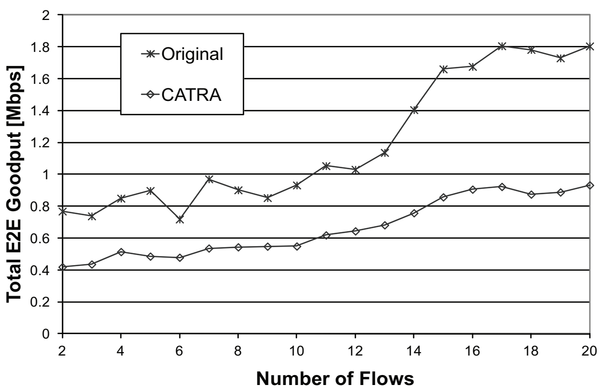

Fig. 12 Total end-to-end throughput in Scenario-5

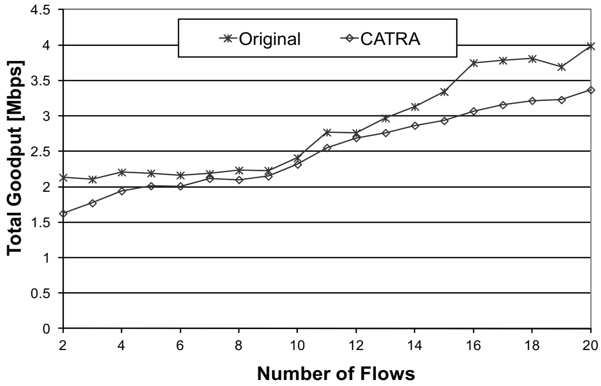

Fig. 13 Total throughput in Scenario-5

5. Conclusion

We proposed a new cross-layer scheme to enhance TCP performance in multi-hop wireless ad hoc network. We measured the channel utilization and defined the Fair Bandwidth Ratio by the number of flows in the carrier sensing range of the examined station. The IEEE 802.11 *CW* size is modified for achieving per-station fairness. The information about channel utilization was sent to Transport layer to adjust TCP sending rate to achieve the per-flow fairness. The simulations on various topologies have proved the effectiveness of CATRA scheme. In addition to fairness, CATRA scheme also achieves quite good performance in terms of throughput.

References

[1] K. Xu, M. Gerla, and S. Bae, "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks," Global Telecommunications Conference: 2002. GLOBECOM '02. IEEE, pp.72–76, 2002.

[2] I.S. Department, "IEEE 802.11 wireless LAN medium access control (MAC) and physical layer (PHY) specifications," ANSI/IEEE Standard 802.11, 1999.

[3] T. Razafindralambo and F. Valois, "Performance evaluation of backoff algorithms in 802.11 ad-hoc networks," PE-WASUN '06: Proceedings of the 3rd ACM international workshop on Performance evaluation of wireless ad hoc, sensor and ubiquitous networks, New York, NY, USA, pp.82–89, ACM Press, 2006.

[4] V. Jacobson, "Congestion avoidance and control," SIGCOMM '88: Symposium proceedings on Communications architectures and protocols, New York, NY, USA, pp.314–329, ACM, 1988.

[5] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?," Communications Magazine, IEEE, vol.39, no.6, pp.130–137, 2001.

[6] Z. Fu, H. Luo, P. Zerfos, S. Lu, L. Zhang, and M. Gerla, "The impact of multihop wireless channel on TCP performance," IEEE Transactions on Mobile Computing, vol.4, no.2, pp.209–221, 2005.

[7] K. Nahm, A. Helmy, and C.C. Jay Kuo, "Tcp over multihop 802.11 networks: Issues and performance enhancement," Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc '05, New York, NY, USA, pp.277–287, ACM, 2005.

[8] A. Al-Jubari, M. Othman, B. Mohd Ali, and N. Abdul Hamid, "Tcp performance in multi-hop wireless ad hoc networks: challenges and solution," EURASIP Journal on Wireless Communications and Networking, vol.2011, no.1, pp.1–25, 2011.

[9] Z. Li, S. Nandi, and A.K. Gupta, "ECS: An enhanced carrier sensing mechanism for wireless ad-hoc networks," Computer Communication, vol.28, no.17, pp.1970–1984, 2005.

[10] C. Chaudet, I.G. Lassous, E. Thierry, and B. Gaujal, "Study of the impact of asymmetry and carrier sense mechanism in IEEE 802.11 multi-hops networks through a basic case," PE-WASUN '04: Proceedings of the 1st ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks, New York, NY, USA, pp.1–7, ACM Press, 2004.

[11] Z. Fang, B. Bensaou, and Y. Wang, "Performance evaluation of a fair backoff algorithm for IEEE 802.11 DFWMAC," MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, New York, NY, USA, pp.48–57, ACM, 2002.

[12] M. Malli, Q. Ni, T. Turletti, and C. Barakat, "Adaptive fair channel allocation for QoS enhancement in IEEE 802.11 wireless LANs," Communications, 2004 IEEE International Conference on, pp.3470– 3475, 2004.

[13] H.S. Choi and J.T. Lim, "Per-flow fair media access control in wireless ad-hoc networks," IEICE Transactions on Communications, vol.E91.B, no.12, pp.4046–4049, 2008.

[14] K. Xu, M. Gerla, L. Qi, and Y. Shu, "Enhancing TCP fairness in ad hoc wireless networks using neighborhood red," Proceedings of the 9th annual international conference on Mobile computing and networking, pp.16–28, ACM, 2003.

[15] K. Chen, Y. Xue, and K. Nahrstedt, "On setting TCP's congestion window limit in mobile ad hoc networks," Communications, 2003. ICC '03. IEEE International Conference on, pp.1080–1084, 2003.

[16] R.S. Cheng and H.T. Lin, "A cross-layer design for {TCP} end-to-end performance improvement in multi-hop wireless networks," Computer Communications, vol.31, no.14, pp.3145 – 3152, 2008.

[17] X. Zhang, J. Lv, X. Han, and D.K. Sung, "Channel efficiency-based transmission rate control for congestion avoidance in wireless ad hoc networks," Communications Letters, IEEE, vol.13, no.9, pp.706–708, 2009.

[18] E. Natalizio, P. Pace, F. Guerriero, and A. Violi, "A reactive and dependable transport protocol for wireless mesh networks," Journal of Parallel and Distributed Computing, vol.70, no.5, pp.431 – 442, 2010.

[19] K. Fall and K. Varadhan, The Network Simulator: ns-2. http://www.isi.edu/nsnam/ns/, 2011.

[20] V. Kolar, S. Razak, N.B. Abu-Ghazaleh, P. Mhnen, and K.A. Harras, "Interference across multi-hop wireless chains," Proceedings of the 2009 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WIMOB '09, Los Alamitos, CA, USA, pp.288–294, IEEE Computer Society, 2009.

[21] R.Jain, D.M. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Technical Report TR-301, DEC Research Report, 1984.