An Adaptive and Responsive Transport Protocol for Wireless Mesh Networks

Pasquale Pace* | Enrico Natalizio* | Vehbi C. Gungor†

*DEIS - University of Calabria, Rende, ITALY
Email: {ppace, enatalizio}@deis.unical.it
†Broadband and Wireless Networking Laboratory
Georgia Institute of Technology, Atlanta, Georgia
Email: gungor@ece.gatech.edu

Abstract—Wireless meshing has been envisioned as the economically viable networking paradigm to build up broadband and large-scale wireless commodity networks. Several different mesh network architectures have been conceived by both industry and academia; however many issues on the deployment of efficient and fair transport protocols are still open. In this paper, an adaptive and responsive transport protocol (AR-TP) is proposed for WMNs in order to fairly allocate the network resources among multiple flows, while minimizing the performance overhead. Compared to the classical end-to-end rate control mechanisms, an hop-by-hop congestion control approach is designed to keep track of dynamic multi-hop network characteristics in a responsive manner. In addition, a coarse-grained end-to-end reliability algorithm is integrated with the proposed hop-by-hop congestion control mechanism to provide packet level reliability at the transport layer. Performance evaluation via extensive simulation experiments show that the AR-TP protocol achieves high performance in terms of network throughput and fairness.

Index Terms—Wireless mesh network, reliable transport, congestion control.

I. INTRODUCTION

Wireless mesh network (WMN) is a promising wireless technology for several emerging and commercially interesting applications, including last-mile broadband Internet access, community and neighborhood networks, intelligent transportation systems and public safety applications [1]. A typical WMN consists of mesh routers and mesh clients as shown in Fig. 1. In this architecture, while static mesh routers form the wireless backbone, mesh clients access the network through mesh routers as well as directly meshing with each other. Different from mobile ad hoc networks, WMNs introduce a hierarchy in the network architecture with the implementation of dedicated and power enabled static mesh routers. Thus, the main focus of communication protocols in static mesh router domain is on improving the network throughput or the performance of individual transfers, e.g., fairness, instead of coping with mobility or minimizing power usage.

Despite the recent advances in wireless communication technologies, the limited link capacity continues to be the main problem for WMNs, since adjacent links cannot operate simultaneously due to mutual interference and the end-to-end path throughput is even lower. Enabling applications to make the most of the raw wireless link capacity, traditionally the responsibility of transport protocol, is thus especially important in these networks. An efficient transport protocol for WMNs should fairly and effectively allocate the network bandwidth among multiple competing flows, while minimizing the performance overhead it incurs. In this paper, we argue that an end-to-end congestion and rate control is inappropriate for wireless mesh networks, because it suffers from the adverse effects of multi-hop wireless environments, such as variable round-trip-times (RTT), high BER and radio interferences. We also present our arguments in terms of under-utilization of network resources and imprecise congestion detection and control.

To eliminate the drawbacks of end-to-end rate control procedures, hop-by-hop congestion control mechanisms have been studied in [9],[12],[13],[14]. Such schemes result in better performance than a corresponding end-to-end scheme by reacting to network congestion faster than end-to-end mechanisms. Although hop-by-hop strategies improve network throughput significantly, they may not recover from packet losses due to node failures or network disconnections. Hence, in addition to hop-by-hop approaches, an end-to-end reliability mechanism needs to be integrated in the transport protocol to provide data transport reliability. In this regard, we argue that hop-by-hop rate control schemes are acceptable for WMNs as long as they improve network utilization and fairness, while providing end-to-end data packet reliability with additional functionalities.

All these discussed challenges and the inherent inefficiencies of the end-to-end approaches in multi-hop wireless environments call for an efficient and responsive transport protocol for WMNs. To address this need, an adaptive and responsive transport protocol (AR-TP) for WMNs is presented in this paper. The AR-TP protocol is an adaptive transport protocol based on hop-by-hop congestion control and coarse-grained end-to-end reliability mechanisms, which are designed to achieve high throughput performance and reliable data transmission in WMNs. In summary, we make the following
contributions in this paper:

1) We study the throughput and fairness performance of existing congestion control and reliable packet delivery mechanisms in the context of static mesh router domain, and show that they are not fairly allocating network resources.

2) We design a fair hop-by-hop rate adaptation mechanism integrated with a coarse-level end-to-end reliability mechanism, which is specifically tailored according to the challenges of mesh router domains.

3) We define the required functionalities and supported capabilities of the mesh router domain, e.g., single buffer for each neighbor, to improve network throughput and fairness. Moreover, compared to the classical end-to-end rate control approaches, we show the network performance improvements of the AR-TP protocol through extensive simulation experiments.

The rest of the paper is organized as follows. In Section II, we present a review of the related works in the context of wireless mesh networks. In Section III, we introduce the design principles and functionalities of the AR-TP protocol. Performance evaluation and simulation results are presented in Section IV. Finally, the paper is concluded in Section V.

II. RELATED WORKS

To the best of our knowledge, no transport protocol has been introduced specifically for WMNs to date, although several transport protocols have been developed for both wired and wireless networks in the last decade [1]. In this section, we explain existing transport protocols with a focus on ad hoc networks, since WMNs share common features with ad hoc networks in spite of their differences.

The shortcomings of TCP in wireless ad hoc networks have been investigated in [2],[4],[6],[7],[8],[10]. Based on the TCP drawbacks, which are revealed through these performance evaluation studies, several transport layer solutions have been proposed in the literature for wireless ad hoc networks. All these solutions propose to solve the problems by improving TCP with additional functionalities, modifications, or getting support from lower layers. In [6], link level protection and ACKing mechanism is advocated to improve the TCP performance over wireless ad hoc networks. In [2], the problems of TCP in dynamic multihop wireless networks are determined and additional mechanisms at media access and routing layers are proposed to improve TCP performance. The explicit link failure notification (ELFN) technique is studied in [7], which is based on explicitly informing the TCP source of the link failures to improve TCP performance. In [8], a transport layer solution (ATCP) is proposed, which introduces a thin layer between the transport and underlying routing layers to improve TCP performance by putting TCP into persist mode whenever the network gets disconnected or there are packet losses due to high bit error rate. In [10], the authors propose a fractional window increment scheme for TCP (TCP-FEW) to prevent unnecessary network contention by limiting the growth rate of TCP’s congestion window.

It is important to note that all these protocols are based on end-to-end (e2e) rate adjustment and congestion control mechanisms and require a fine-grained end-to-end communication between the source and the destination. Therefore, they may experience significant network inefficiency in WMNs due to the dynamic characteristics of multi-hop wireless environments and e2e delayed and even obsolete receiver rate feedbacks.

In this paper, to eliminate the drawbacks of e2e rate control procedures and to avoid incipient network congestion quickly, we propose a hop-by-hop congestion control mechanism. In addition, we integrate a coarse-level e2e reliability algorithm with our hop-by-hop rate control mechanism to provide data packet level reliability at the transport layer. This way, we decouple the rate control and reliability mechanisms and eliminate the need for self-clocking through the arrivals of ACKs and fine-grained RTT-based rate estimations.

III. AR-TP: AN ADAPTIVE AND RESPONSIVE TRANSPORT PROTOCOL FOR WMNs

The proposed AR-TP protocol includes both efficient hop-by-hop rate adjustment and reliability mechanisms to achieve high performance reliable data transport in WMNs. Furthermore, compared to end-to-end rate control schemes, hop-by-hop rate adaptation strategy of the AR-TP protocol enables each router to keep track of dynamic wireless channel conditions. Note that with the use of hop-by-hop strategy, each mesh router can adapt its data transmission rate opportunistically in case of multi-channel WMNs. In the following sections, we describe the details of the AR-TP protocol operation in mesh router domain, respectively.

A. Reliability in Wireless Mesh Router Domain

Many transport protocols employ an end-to-end loss detection and retransmission technique where receiver detects...
missing sequence numbers and requests retransmissions of lost packets from the original sender. While simple to implement due to its stateless network core, the end-to-end approach is unsuitable for mesh router domain as they suffer from several packet losses due to high bit error rates (BER) in wireless environment. Moreover, in mesh router domain, packets need to pass through multiple wireless links and thus the packet losses tend to be accumulated. When a packet is lost on an intermediate hop, end-to-end strategy requires the retransmission to traverse the entire path all over again. This leads to wastage of bandwidth on all preceding hops where the prior transmissions were successful. In addition, the packet loss probability increases as they travel more hops.

To overcome these problems, the AR-TP protocol incorporates a hybrid reliability mechanism including both hop-by-hop and end-to-end reliability strategies. Specifically, when an intermediate node detects a packet is lost at a particular hop, it immediately retransmits the packet for a fixed number of times without waiting for end-to-end feedbacks, which typically incurs a round-trip delay. This saves unnecessary retransmissions on previous hops over which the packet has already successfully traversed. In Appendix A, through an analytical approach, we derive the expression for the total number of packet transmissions necessary for hop-by-hop reliable delivery of a packet over an $N$ hop path. We analytically show that an hop-by-hop reliability strategy can provide packet reliability for a certain range of packet loss rates and number of hops.

Although hop-by-hop reliability mechanism provides packet reliability to a certain extent, data packets can still be dropped by intermediate nodes due to buffer overflows or node failures. Thus, a reliable transport protocol cannot rely on only hop-by-hop reliability mechanisms to detect such packet losses. To recover from these losses, the AR-TP protocol supports an end-to-end negative acknowledgment and retransmission scheme. That is, periodically a flow’s receiver sends back a negative ACK (NACK) that lists all the packets that it has not received and the last packet it successfully receives. Due to proactive hop-by-hop retransmission, responsive rate adaptation and congestion control, which will be explained in the following section, empirically end-to-end NACKs are infrequent and hence introduces little overhead compared to the classical TCP ACKs’ overhead.3

Here, it is important to note that with the integration of a coarse-level e2e reliability algorithm with hop-by-hop retransmission strategy, the AR-TP protocol decouples the rate control and reliability mechanism. This way, it eliminates the need for self-clocking through the arrivals of ACKs and thus it overcomes ACK bunching problem of TCP-based mechanisms.

B. Local Rate Adaptation in Wireless Mesh Router Domain

In AR-TP protocol operation, to provide service differentiation and to separate the flows coming from different routers, each router is equipped with one buffer for each neighbor. Furthermore, two different thresholds are used in mesh router buffers: i) $\Gamma_{cong}$ is the threshold over which the mesh router buffer is congested, and ii) $\Gamma_{util}$ is the threshold over which the mesh router buffer is working in an high utilization condition. Note that these thresholds are required to define a tolerance zone around the high utilization operating point for practical purposes, i.e., reducing oscillations that might lead to instability in the network. Here, our objective is to balance the tradeoff between high network utilization and non-congested network operation.

The rate control scheme of AR-TP protocol consists of two main mechanisms: i) the back pressure mechanism, which requests the previous neighbor in the data flow for a rate adaptation; and ii) the forward threshold adaptation, which asks the next hop neighbor in the data flow for the adjustment of its high utilization threshold $\Gamma_{util}$. The details of these mechanisms are described in the following.

C. Back Pressure and Forward Threshold Adaptation

The rate control procedure of the AR-TP protocol is modelled as a finite state machine with four states for each mesh router, i.e., increase, decrease, maintain and stand-by, as shown in Fig. 2. Basically, its increase is more aggressive than that of TCP, decrease is less conservative than that of TCP, and more importantly operates in a maintain state when network conditions do not change considerably. In other words, since congestion in wireless network can be a transient condition due to the extreme variability in the link quality condition, the AR-TP protocol reacts to the congestion in a smooth way for a short period providing a linear decrease and in an aggressive way after the short period, if the congestion status is still present. On the other hand, if the buffer occupancy is underutilized, the AR-TP protocol increases the transmission rate cautiously to bring the system in a high utilization operating point in terms of throughput.

The mechanism of rate adjustment is performed periodically at a fixed time interval. The time interval is chosen short enough, e.g., on the order of hundreds of ms [5], to keep track

![Fig. 2. State transition diagram for the AR-TP protocol operation.](image-url)
AR-TP()
If (Buffer Occupancy > \( \Gamma_{cong} \))
if (\( \Psi > \Psi_{max} \))
/*Decrease state*/
/*Permanent Congestion: Decrease aggressively*/
\[ \alpha = \min(\alpha_{max}, \alpha \times (1 - \Psi - \Psi_{Max})^{-1}) \]
\[ R_{i+1} = R_i \times (1 - \alpha) \]
else if (\( \Psi < \Psi_{Max} \))
/*Decrease state*/
/*Transient Congestion: Decrease rate cautiously*/
\[ \Psi = \Psi + 1 \]
\[ R_{i+1} = R_i \times (1 - \alpha) \]
else if (Buffer Occupancy > \( \Gamma_{util} \))
/*Maintain state*/
/*Hold the current rate*/
\[ \Psi = 0 \]
else
/*Increase state*/
/*Increase rate cautiously*/
\[ R_{i+1} = R_i \times (1 + \beta) \]
\[ \Psi = 0 \]
end;

Fig. 3. Algorithm of the AR-TP protocol operation.

The entire AR-TP protocol operation is presented in the pseudo-algorithm given in Fig. 3. In this algorithm, \( \Psi_{Max} \) is a constant parameter representing the maximum number of attempts used by the algorithm before it reduces the rate in an aggressive way. \( \Psi \) is a simple counter value adjusting the decrease amount of the rate accordingly. Moreover, \( \alpha \) and \( \beta \) represent the decrease and increase factor, respectively. Note that the AR-TP protocol seeks to detect incipient congestion and react to it in a responsive manner. The objective of the AR-TP protocol is to decrease the transmission rate multiplicatively in order to ensure fairness, less aggressively when reacting to incipient congestion in order to improve network efficiency, and more aggressively when reacting to permanent congestion in order to reduce packet loss and alleviate congestion quickly. To achieve these goals, the AR-TP protocol maintains a history of transmission rate increase or decrease in the recent past by the help of \( \Psi \) parameter.

Furthermore, when multiple flows pass through the same mesh router, the data transmission rate is equally divided among the flows passing through the same node. Here, we employ simple fair sharing principle to equally distribute the network resources. Note also that the AR-TP protocol can also work with other service disciplines such as per-flow quality-of-service (QoS) based disciplines, which can further improve the performance and are beyond the scope of the paper.

Another important aspect of the AR-TP protocol is that of the dynamics of the wireless channel. Note also that the mesh router enters the stand-by state, when no rate feedback packet is received by the sender for a certain time interval or when a route error occurs. In back-pressure mechanism, nodes signal backwardly local congestion according to the their buffer status; this reduces packet loss rates and prevents the wasteful transmissions of packets that are destined to be dropped at the downstream node.

In fact, the sending rate increase depends on the status of the \( \text{router}_{N+1} \). When the buffer utilization for this router is below the high utilization threshold \( \Gamma_{util} \), it asks the \( \text{router}_N \) to increase its sending rate, i.e., increase state. Unless it experiences congestion or under-utilization, the \( \text{router}_{N+1} \) asks the \( \text{router}_N \) to keep its corresponding transmission rate as constant, i.e., maintain state. To release network congestion promptly, the \( \text{router}_N \) can also request the \( \text{router}_{N+1} \) to raise its high utilization threshold \( \Gamma_{util} \). This way, it can have a chance to get a higher share of transmission rate to release its congestion situation. In this context, the calculation of the forward \( \text{router}_{N+1} \) high utilization threshold increase can depend on the current buffer occupancy of the \( \text{router}_N \). In other words, the closer to the total occupancy, the higher the request of threshold increase.

Note also that after network congestion is released, each mesh router, whose utilization threshold is increased, throttles its utilization threshold cautiously, until the current high utilization threshold is equal to the minimum pre-configured value. This way, each mesh router dynamically adjusts its protocol configurations to adapt to the varying congestion level of the network. In the following, we present the details of forward threshold mechanism.

Let \( Q_N \) be the forecasted buffer fullness level for the next rate feedback interval. Then, the normalized \( Q_N \) value according to the buffer size \( B \) can be calculated as follows:

\[
Q_N = \frac{C_N + I_N - O_N}{B}
\]

where \( C_N \) is the current buffer level, \( I_N \) is the current input...
rate, $O_N$ is the output rate and $B$ is the buffer size of the $r_{outer}$.$N$. When this forecasted value exceeds the congestion threshold value, the $r_{outer}$.$N$ asks the $r_{outer}$.$N$+1 to raise its utilization threshold of a factor of $I_{N+1}$ calculated as follows:

$$I_{N+1} = \frac{(Q_{N} + 1)}{2}$$

(2)

The value of $I_{N+1}$ is used to determine the new high utilization threshold $\Gamma_{new} _{util}$ for the $r_{outer}$.$N$+1 by using the following formula:

$$\Gamma_{new} _{util} = (\Gamma_{cong} - \Gamma_{curr} _{util}) * I_{N+1} + \Gamma_{curr} _{util}$$

(3)

where $I_{N+1}$ represents the fraction of the interval $(\Gamma_{cong}(t) - \Gamma_{util})$, which has to be added to the current high utilization threshold $\Gamma_{curr} _{util}$.

Following this strategy, we fix a value for the high utilization threshold at the starting system phase, after that the high utilization threshold of the $r_{outer}$.$N$+1 will be automatically adapted according to the congestion condition of the $r_{outer}$.$N$. At each rate feedback interval the $r_{outer}$.$N$+1 also decreases its utilization threshold with a certain amount, until the current high utilization threshold is equal to the minimum value of $\Gamma_{util}$. This way, wireless mesh routers self-configures themselves based on the varying congestion level of the network.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the AR-TP protocol through simulations over a variety of scenarios. For the simulation experiments, we set up an evaluation environment using ns-2 network simulator [11]. In our simulations, wireless channel bandwidth is 2 Mbps and all MAC layer parameters of IEEE 802.11 are configured to provide a transmission range of 250m and a carrier sensing range as well as an interference range of 550m. Thus, our setting is consistent with real wireless networks, in which the transmission range of a node is typically smaller than its interference range. The packets generated are of size 1000 bytes and the nodes employ Dynamic Source Routing (DSR) protocol to communicate source data to the destination. In our simulations, we consider scenarios with FTP-like data transfer in different network topologies. That is, the sender transmits packets continuously, representing a large data file transfer. Moreover, for each simulation, we run 10 experiments with different seeds and take the average of the measured values. Unless specified otherwise, we use the simulation parameters listed in Table I.

Furthermore, we compare the AR-TP protocol with several other transport protocols, namely TCP-Newreno, TCP-Sack representing the most widely used transport protocol family over the Internet, TCP-AP[4], TCP-FEW[10] and TCP-ELFN[7], which are the transport protocols specifically proposed for wireless ad hoc networks. In the following sections, we present the details of the performance results under different types of topologies, such as chain, parking lot, grid and random topologies.

A. Chain Topology

In the first simulation experiment, we consider an equally spaced chain composed of $n$+1 nodes, i.e., $n$ hops, with a single flow as shown in Fig. 5. Data flow travels along the chain from the leftmost node, i.e., the sender, to the rightmost node, i.e., the receiver. Fig. 6 shows the throughput of the examined transport protocols for varying hop number.

As seen in Fig. 6, the AR-TP protocol outperforms other transport protocols under comparison, since it dynamically adjusts data rate of the flows in a responsive manner due its hop-by-hop nature and avoids network congestion, while maximizing the network utilization efficiency. For example, for 7 hop connection, we obtain that the throughput achieved by the AR-TP protocol is around 82%, 68%, 63%, 55% and 48% higher than that of TCP-NewReno, TCP-Sack, TCP-FEW[10], TCP-ELFN[7] and TCP-AP[4], respectively.

B. Parking Lot Topology

A wireless mesh network (WMN) typically has one or multiple gateways that connect wireless nodes to the wired

<table>
<thead>
<tr>
<th>Table I: NS-2 SIMULATION PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>Bandwidth</td>
</tr>
<tr>
<td>Radio range</td>
</tr>
<tr>
<td>Carrier sensing and interference ranges</td>
</tr>
<tr>
<td>MAC protocol</td>
</tr>
<tr>
<td>Routing protocol</td>
</tr>
<tr>
<td>Packet size</td>
</tr>
<tr>
<td>Buffer length</td>
</tr>
<tr>
<td>Feedback interval</td>
</tr>
<tr>
<td>Rate increase and decrease factors</td>
</tr>
<tr>
<td>Congestion and utilization thresholds</td>
</tr>
<tr>
<td>Simulation time</td>
</tr>
</tbody>
</table>

Fig. 5. N hop chain topology. The distance between each node is 200m.

Fig. 6. FTP throughput results in a chain topology.
Internet as illustrated in Fig. 1. Because different nodes in a WMN have different hop distances to these gateways, it is important that the deployed transport protocol ensures network flows with different hop counts to get an equal share of the network resources. Although most of the existing transport protocols can fairly allocate the network resources among data flows that share the same set of hops, they cannot handle flows with different number of hops due to their RTT-based rate control nature [15] and [5].

In this section, we establish a simulation scenario on a parking lot topology with different length of flows as shown in Fig. 7. The name of the scenario is intended to convey a situation analogous to many cars simultaneously attempting to leave a parking lot resulting in congestion and the lack of fairness [5]. In this simulation scenario, each flow is establishing an FTP connection to the same destination. Fig. 8 shows the FTP throughputs of the four flows under TCP-Newreno, TCP-Sack, TCP-AP[4], TCP-FEW[10], TCP-ELFN[7] and AR-TP protocols, respectively. As shown in Fig. 8, all examined transport protocols other than the AR-TP protocol showed extreme unfairness by penalizing the flow with the largest hop count, i.e., Flow 1, and giving most of the available bandwidth to the shortest flow, i.e., Flow 4. Note that the performance improvement of the AR-TP protocol over other protocols mainly comes from its local and RTT-independent rate control scheme, which is described in Section III.

C. Grid Topology

In wireless mesh networks, it is highly possible that a data flow can pass through several congested neighborhoods, although the degree of congestion is different. In this section, we evaluate the performance of the AR-TP protocol under such conditions with a grid topology having six FTP flows as shown in Fig. 9. The vertical and horizontal distance between neighboring nodes is 200m and each FTP connection traverses 4 hops.

In this topology, several bottleneck neighborhoods can be present simultaneously because of multiple crossing data traffics. The overall throughput of each flow is depicted in Fig. 10. As shown in Fig. 10, TCP-AP[4], TCP-FEW[10] and AR-TP improve the fairness among data flows compared to TCP-Sack. However, the AR-TP protocol provides more fair allocation of network resources for flow 2 and flow 5 because of its more responsive rate control scheme compared to other protocols. Note also that in case of all transport protocols flow 3 and flow 6 achieve more throughput compared to other flows, since they have less interfering flows due to their relatively far position to the senders of flow 1 and flow 4. This result is
A. Transmission attempts for reliability

The total number of link transmissions is a truncated geometric distribution with parameter $p$ because the link has reliably forwarded the packet, therefore, conditional expected number of transmissions, $E_s$, over a single link, is given by:

$$E_s = \sum_{i=1}^{r} i \cdot p^{i-1} \cdot (1-p) = \frac{1}{1-p} - \frac{r \cdot p^r}{1-p} \quad (4)$$

The corresponding conditional number of transmissions, given forwarding failure, is $E_f = r$ because link packet delivery fails only after exactly $r$ transmissions. Moreover, since each link fails to forward the packet independently with $q$, the unconditional probability of successful end-to-end delivery (without other source transmission) is given by $P_s = (1-q)^N$, and the unconditional probability of unsuccessful end-to-end delivery is given by:

$$P_f = 1 - (1-q)^N \quad (5)$$

Then, we calculate $E_s^N$ that represents the expected number of total packet transmissions considering that the e2e forwarding attempt failed. Since a downstream router or client forwards packets only when all the upstream routers or clients successfully transmitted the packet, it is easy to determine that the conditional probability that failure occurs at the $i^{th}$ link is given by:

$$P_f(i | e2e - failure) = \frac{P_f(i)}{P_f} = \frac{(1-q)^{i-1} \cdot q}{P_f}$$

The expected number of total link-layer transmissions (over all the upstream nodes) is $(i-1) \cdot E_s + E_f$ when a failure happens on the $i^{th}$ link. Consequently, the conditional mean number of total link-layer transmissions during link failure is:

$$E_f^N = E_f + E_s \cdot (1-q) \cdot \left\{ \frac{1-N \cdot (1-q)^{N-1} + (N-1) \cdot (1-q)^N}{q \cdot (1-(1-q)^N)} \right\} \quad (6)$$

When, instead, a packet has been received at the destination, the total expected number of transmissions is

$$E_s^N = N \cdot E_s \quad (7)$$

Clearly end-to-end transmission and end-to-end retransmissions are independent from each other, so the total number of end-to-end transmissions for reliable delivery is geometrically distributed with a mean of $\frac{1}{1-P_f}$; hence, for successfully transmit a packet, we will have, an average $\frac{1}{1-P_f} - 1$ failed end-to-end transmissions. In conclusion, the total expected number of distinct packet transmissions is

$$E = E_f^N \cdot \frac{P_f}{1-P_f} + E_s^N \quad (8)$$

where $E_f^N$, $E_s^N$ and $P_f$ are given by equations $6$, $7$ and $5$ respectively. Accordingly with the previous analysis we can show the results in term of expected number of transmission packets varying the number of retransmission at the link layer from 1 to 7 times.

Fig. 12 shows that using a number of link layer retransmission greater than 4 there is no improvement and the system provide the same reliability also in high packet loss probability condition (e.g., 20%) and in long multi-hop connections (e.g., 10 hops).

B. Acknowledgment overhead in TCP

In this section we derive the equation for the percentage ACK overhead in TCP with and without delayed-ACK option. Note that Delayed-ACK scheme reduces this overhead to one ACK for every two data segments. Despite enabling delayed-ACK, the ACK traffic imposes substantial overhead on connection throughput.

D. Random Topology

To get intuition on the performance of the AR-TP protocol in more realistic scenarios, we consider a random topology of 100 nodes randomly distributed over a 1000m x 1000m deployment field. The effect of traffic load on the static mesh router domain is studied by investigating scenarios with 1, 5 and 10 connections, respectively. The source-destination pairs are randomly chosen from the set of 100 nodes in the network. Each source also runs an FTP file transfer during the simulation.

In Fig. 11, we present the average throughput results of the AR-TP protocol and other transport protocols under comparison, i.e., TCP-Newreno, TCP-Sack, TCP-AP[4], TCP-FEW[10], TCP-ELFN[7]. Consistent with the previous results, the AR-TP protocol outperforms other transport protocols under comparison. For example, for 5 flow, we obtain that the average throughput achieved by the AR-TP protocol is around 35%, 30%, 28%, 25% and 15% higher than that of TCP-Newreno, TCP-Sack, TCP-AP[4], TCP-FEW[10] and TCP-ELFN[7], respectively. This is because the AR-TP protocol dynamically shapes data traffic based on network congestion level. In addition, the proper reaction of AR-TP to congestion related and non-congestion related losses, such as route failures, prevents it from any performance degradation.

V. Conclusions

In this paper, we proposed a new adaptive and responsive transport protocol (AR-TP) for wireless mesh networks (WMNs). The AR-TP protocol is based on a hop-by-hop rate control and congestion detection mechanism in order to address the shortcomings of classical end-to-end TCP-based solutions and the unique characteristics of WMNs. We have shown through simulation analysis that the AR-TP protocol has significant performance benefits in terms of throughput increase and fairness compared with classical and previous transport protocols in static mesh router domains. Future work includes extending the AR-TP protocol to different application scenarios, e.g., multi-radio mesh routers and mobile mesh client domain, and investigating the impact of different network parameters on the design of transport protocols of WMNs. We also plan to implement the developed mechanisms on a physical testbed.

Appendix

A. Transmission attempts for reliability

In this section, through an analytical approach, we derive the expression for the total number of packet transmissions necessary for hop-by-hop reliable delivery of a packet over an $N$ hops path. We consider that the packet error rate for each hop is $p$ and the maximum number of retransmissions at the link layer is $r$.

We know that only when all $r$ transmissions fail, also the reliable link forwarding fails, thus the unconditional probability of link packet transmission failure, which we call $q$, is given by $q = p^r$; the corresponding probability of reliable link delivery is then $1 - q$. Reasonable because flows with fewer interfering flows, i.e., less constraining bottlenecks, achieve higher throughput, which should not be recognized as unfairness.

$$E_s = \sum_{i=1}^{r} i \cdot p^{i-1} \cdot (1-p) = \frac{1}{1-p} - \frac{r \cdot p^r}{1-p} \quad (4)$$

The corresponding conditional number of transmissions, given forwarding failure, is $E_f = r$ because link packet delivery fails only after exactly $r$ transmissions. Moreover, since each link fails to forward the packet independently with $q$, the unconditional probability of successful end-to-end delivery (without other source transmission) is given by $P_s = (1-q)^N$, and the unconditional probability of unsuccessful end-to-end delivery is given by:

$$P_f = 1 - (1-q)^N \quad (5)$$

Then, we calculate $E_s^N$ that represents the expected number of total packet transmissions considering that the e2e forwarding attempt failed. Since a downstream router or client forwards packets only when all the upstream routers or clients successfully transmitted the packet, it is easy to determine that the conditional probability that failure occurs at the $i^{th}$ link is given by:

$$P_f(i | e2e - failure) = \frac{P_f(i)}{P_f} = \frac{(1-q)^{i-1} \cdot q}{P_f}$$

The expected number of total link-layer transmissions (over all the upstream nodes) is $(i-1) \cdot E_s + E_f$ when a failure happens on the $i^{th}$ link. Consequently, the conditional mean number of total link-layer transmissions during link failure is:

$$E_f^N = E_f + E_s \cdot (1-q) \cdot \left\{ \frac{1-N \cdot (1-q)^{N-1} + (N-1) \cdot (1-q)^N}{q \cdot (1-(1-q)^N)} \right\} \quad (6)$$

When, instead, a packet has been received at the destination, the total expected number of transmissions is

$$E_s^N = N \cdot E_s \quad (7)$$

Clearly end-to-end transmission and end-to-end retransmissions are independent from each other, so the total number of end-to-end transmissions for reliable delivery is geometrically distributed with a mean of $\frac{1}{1-P_f}$; hence, for successfully transmit a packet, we will have, an average $\frac{1}{1-P_f} - 1$ failed end-to-end transmissions. In conclusion, the total expected number of distinct packet transmissions is

$$E = E_f^N \cdot \frac{P_f}{1-P_f} + E_s^N \quad (8)$$

where $E_f^N$, $E_s^N$ and $P_f$ are given by equations $6$, $7$ and $5$ respectively. Accordingly with the previous analysis we can show the results in term of expected number of transmission packets varying the number of retransmission at the link layer from 1 to 7 times.

Fig. 12 shows that using a number of link layer retransmission greater than 4 there is no improvement and the system provide the same reliability also in high packet loss probability condition (e.g., 20%) and in long multi-hop connections (e.g., 10 hops).
If \( \rho_{ACK} \) is the percentage overhead of TCP acknowledgements when there is one TCP ACK for every two TCP segments, then it follows,
\[
\frac{1}{2} \cdot \frac{(1084 + 656/R)}{(1084 + 12656/R)} \% \leq \rho_{ACK}
\]

Obviously, in the worst case, delayed-ACK degenerates to per-packet ACK.
\[
\rho_{ACK} \leq \frac{(1084 + 656/R)}{(1084 + 12656/R)} \%
\]

Combining the last two inequalities, it is clear that the percentage of acknowledgement overhead (\( \rho_{ACK} \)) increases substantially as the link layer uses more sophisticated encoding to improve channel output.

**References**


