Global Journals LaTeX JournalKaleidoscope™

Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

We TCP-AP: Wireless Enhanced TCP-AP

Luis Barreto[1]

[1] Instituto Politecnico de Viana do Castelo

Received: 11 December 2015 Accepted: 3 January 2016 Published: 15 January 2016

Abstract

Congestion control in wireless networks is strongly dependent on the dynamics and instability of wireless links. It is known that TCP experiences serious performance degradation problems in wireless networks. New approaches based on TCP try to overcome these problems but, although their performance is increased, they incur in congestion control errors, since they do not evaluate accurately the capacity and available link bandwidth in wireless networks. This is also the case of TCP-AP (Adaptive Pacing) that, although presenting clear advantages in wireless networks when compared to other TCP-based approaches, its performance is still lower than rate-based approaches. In this paper we propose a new congestion control protocol based in TCP-AP, the Wireless Enhanced TCP-AP (WE TCPAP).This protocol relies on the MAC layer information gathere by a new method to accurately estimate the available bandwidth and the path capacity over a wireless network path. The new congestion control mechanism is evaluated in different scenarios in wireless mesh and ad-hoc networks, and compared against several approaches for wireless congestion control. It is shown that the new WE TCP-AP outperforms the base TCP-AP, with a more stable behavior and better channel utilization, and its performance gets close to the one of ratebased protocols. This is a very important result, as we show that TCP-based approaches are still able to have good performance in wireless mesh and ad-hoc networks.

Index terms— CP-AP congestion control available bandwidth path capacity node count fair share performance wireless networks. cp-ap congestion control available ban

1 I. Introduction

ireless networks have major factors that limit their performance, such as limited capacity and available bandwidth [5]. This results in severe congestion collapses. The support of a congestion control scheme that provides an efficient and accurate sharing of the underlying network capacity among multiple competing applications is crucial to the efficiency and stability of wireless networks. Actively using link capacity and available bandwidth for congestion control will surely make these networks more efficient. Link capacity can vary due to a variety of factors, such as handoffs, channel allocation and, of course, channel quality. In [25] we proposed a new mechanism for measuring wireless link capacity and available bandwidth, called rt-Winf. rt-Winf uses the information already present in the network and available at the MAC layer to accurately determine the link capacity and available bandwidth. Another important characteristic of rt-Winf is that it can be used by any existing wireless equipment through a cross-layer shared database.

To address the congestion control problems of wireless networks, new congestion control techniques have been proposed, through TCP-based (AIMD -Additive Increase Multiplicative Decrease) and Ratebased schemes. In [12] we proposed new rate based congestion control protocols, based on eXplicit Congestion Protocol (XCP) [20] and Rate Control Protocol (RCP) [16], with link capacity and available bandwidth estimation through rt-Winf. The performance of these protocols is increased when compared to available wireless-enabled congestion control approaches. Although these protocols can work together with TCP, they are not able to inter-operate in an end-to-end system. Therefore, it is very important to develop a TCP-based approach that is able to efficiently work in wireless based environments, and that can provide comparable performance to rate-based approaches.

The Transmission Control Protocol with Adaptive Pacing (TCP-AP) [17] is a congestion control mechanism based on TCP [23], specifically designed for ad-hoc multi-hop wireless networks, being one of the wireless-enabled TCP protocols with better performance. TCP-AP uses a hybrid scheme between a pure rate-based transmission control and TCP's use of the congestion window. However, TCP-AP, as studied in [12], is very conservative and does not use very efficiently the medium. TCP-AP relies only on a 4-hop propagation delay technique to evaluate the link available bandwidth and capacity, not taking into consideration all the factors that influence link evaluation. New simulation results, presented in this paper, conducted in wireless ad-hoc scenarios, clearly show that TCP-AP lacks of efficiency and is not using correctly the medium; it is not evaluating correctly the parameters that are real constraints in such networks.

In this paper, an enhanced version of the work in [10], we propose a new approach to improve TCP-AP behavior, based on the integration of the on-line capacity and available bandwidth estimation technique, rt-Winf, with TCP-AP through a cross layer approach. Simulation results show that the rt-Winf integration is improving TCP-AP performance. However, it still reflects some of TCP-AP flaws, especially concerning fairness and the fact that it does not use the entire network information, as it relies on the knowledge of only 4 hop propagation delay. Thus, it is also important to improve its operations with the knowledge of all nodes along the path contending for available bandwidth and capacity, introducing the fairness factor and the network interaction behavior. Therefore, we propose a new approach to take this information into account. New simulation results show that the consideration of the node path effect and the integration of rt-Winf clear improve base TCP-AP performance. The simulation results were conducted on both ad-hoc and mesh wireless networks. These considerations represent a significant step towards congestion control in wireless networks, as they show that TCP-based schemes are able to efficiently work in these wireless adverse environments.

The remaining of this paper is organized as follows. Next section, section II, briefly presents the related work on congestion control mechanisms for wireless networks. Section III briefly describes the rt-Winf mechanism. Section IV presents a first evaluation of TCP-AP and addresses main TCP-AP problems. Then, section V describes how rt-Winf is integrated with TCPAP, and section VI presents a new approach for the node path contention count effect. Section VII depicts and discusses the results obtained through simulation, using mesh and adhoc scenarios with different characteristics. Finally, section VIII presents the conclusions and future work.

2 II. Related Work

New efforts have been made to improve congestion control in wireless networks. The Wireless Control Protocol (WCP) [24], WCP with Capacity Estimation (WCPCap) [24], Cooperative Neighborhood Airtime-limiting (CNA) [19], HOP [21], EZ-Flow [9] and Neighborhood Random Early Detection (NRED) [28] are some examples.

TCP, as the most used congestion control protocol, has also been the underlying development for some congestion mechanisms in wireless environments, such as TCP-AP [17]. More recent developments, based on rate based congestion protocols, like the eXplicit Control Protocol (XCP) [20] and the Rate Control Protocol (RCP) [16], are XCP-b [4], XCP-Winf [11] and RCP-Winf [11].

WCP is an AIMD-based rate-control protocol for multi-hop wireless networks. WCP was designed with the goal to be used on networks with arbitrary traffic pattern. During congestion, WCP signals all flows in a neighborhood of congestion and sets the control interval to the maximum Round Trip Time (RTT) of any flow in the neighborhood. WCP explicitly exchanges congestion information within a neighborhood, and all nodes within the neighborhood mark packets with congestion indicators, triggering rate reductions at the source. neighborhood, and divides this capacity to contending flows. With WCPCap it is evident that considering wireless congestion collectively over a neighborhood of a link is essential to any future design of wireless congestion control. WCPCap uses a sophisticated stochastic model for estimating the achievable rate region, given packet loss rates, topology, and flow information. It then allocates the achievable capacity fairly across flows, sending feedback to the sources.

CNA is a hybrid approach, in that it explicitly allocates the channel resources, but provides only imprecise feedback to the source. CNA achieves efficient airtime allocation by distributing available airtime within a wireless neighborhood, then monitoring the air utilization and dynamically redistributing unused airtime to improve overall airtime usage. The authors of CNA claim that it achieves transparency, low overhead, and responsiveness. CAN considers airtime to be the fraction of the time that a wireless link can occupy the shared channel; it does not consider, however, the time a node is waiting to transmit.

HOP is a clean-slate design of hop-by-hop congestion control. HOP tries to use reliable per-hop block transfer as a building block. HOP is referred by its authors as: fast, because it eliminates many sources of overhead as well as noisy end-to end rate control; robust to partitions and route changes, because of hopby-hop control as well as in-network caching; and simple, because it obviates complex end-to-end rate control as well as complex interactions between the transport and link layers.

EZ-Flow is a back-pressure congestion control mechanism which does not require explicit signaling. A back-pressure mechanism flow control allows loss-free transmission by having gateways verify that the next gateway has sufficient buffer space available before sending data, thus EZ-Flow is a cooperative congestion control. EZ-flow operates by adapting the minimum congestion window parameter at each relay node, based on an estimation of the buffer occupancy at its successor node in the mesh.

NRED identifies a subset of flows which share channel capacity with flows passing through a congested node. However, it identifies only a subset of contending flows: it misses flows that traverse 2-hop neighbors of a node without traversing its 1-hop neighbors. Moreover, the mechanism to regulate the traffic rates on these flows is quite a bit complex (it involves estimating a neighborhood queue size and using RED-style marking on packets in this queue). NRED has an important disadvantage, being intimately tied to a particular queue management technique (RED) and requires special hardware for channel monitoring.

TCP-AP uses a 4-hop propagation delay technique, and it considers a hybrid scheme between a

3 E

WCPCap is a distributed rate controller that estimates the available capacity within each pure rate-based transmission control and TCP's use of the congestion window to trigger new data packets to be sent into the network. A TCP sender adaptively sets its transmission rate using an estimate of the current 4hop propagation delay and the coefficient of variation of recently measured round-trip times. The 4-hop propagation delay describes the time elapsed between transmitting a TCP packet by the TCP source node, and receiving the packet at the node which lies 4 hops apart from the source node along the path to the destination.

XCP-b is a XCP based congestion control mechanism, as it tries to extend XCP for shared-access, multi-rate wireless networks by calculating, using very complex heuristics, the available bandwidth of the wireless channel. XCP-b uses indirect parameters such as queue sizes and the number of link layer retransmissions to obtain the desired measurements. XCP-b major drawback is that it becomes inefficient over highly dynamic wireless networks. In wireless environments with few nodes and less mobility, XCP-b can obtain good performance results in terms of stability, fairness, and convergence.

XCP-Winf and RCP-Winf are two new congestion control mechanisms that use MAC layer information through a cross layer communication process. The rt-Winf algorithm performs link capacity and available bandwidth calculations without interfering in the network dynamics, and without increasing network overhead; these parameters are then passed to the congestion control mechanisms based on explicit congestion notifications, XCP and RCP, to accurately determine the network status and act accordingly. The evaluation results of XCP-Winf and RCPWinf, obtained through ns2 [1] simulations, show that the rt-Winf algorithm improves significantly XCP and RCP behavior making them more efficient and stable. In [12] it was shown that these rate-based approaches have better performance in wireless scenarios when compared to TCP-based approaches. In this paper we will work on the enhancement of TCP-based approaches to target a Table I qualitatively compares the previous referred mechanisms along some dimensions. We will then work with TCPAP protocol, being the one TCPbased, and providing good performance when compared to other similar approaches.

4 III. Rt-Winf Description

The rt-Winf mechanism has been inspired by IdleGap [7], but with the purpose to mitigate IdleGap main issues and problems, and also with the intention to be compatible with all systems and being able to determine both the link capacity and available bandwidth without overloading the network. rt-Winf does not introduce any change to the OSI Model, as opposed to IdleGap, being able to obtain all the necessary times to obtain the path capacity and the available bandwidth. Another important aspect of rt-Winf, relatively to IdleGap [7], is that it does not use the DataRate value of the IEEE802.11 header [3] as the link capacity estimation.

This mechanism can be used with the Carrier Sense Multiple Access -Collision Avoidance (CSMA-CA) [13] Request To Send (RTS)/ Clear To Send (CTS) handshake enabled in the wireless communication or with probe packets when RTS/CTS packets are not present. The usage of RTS/CTS handshake is optional, but it is nowadays widely supported by all wireless equipments. Its usage, in a traditional wireless network, represents a negligible cost in terms of overhead [8]. a) RTS/CTS Packets rt-Winf with RTS/CTS control packets enabled relies on this handshake to correctly retrieve the NAV values.

In order to evaluate the accuracy of the duration field on the IEEE802.11 header, we performed a large number of captures (200). We concluded that the duration value on data packets is not reliable, because different sized packets have always the With the obtained captures, it was possible to realize how each state managed the received packets. In the case of the Sender state, the node was able to capture the CTS, DATA and ACK packets. A node in the Receiver state was able to capture the RTS and the DATA packets, while a node in the Onlooker state was able to capture the complete set of packets: RTS, CTS, DATA and ACK. This different knowledge implied the conception of different algorithms for each state. Then, we proposed that each node state uses a different method to determine the Idle Rate. In the case of the Sender, it is considered the NAV of the CTS packets on the available bandwidth calculation. For the capacity calculation, it is considered the time that the channel is busy, that is, the difference between ACK time, CTS time and the duration of the occurred Short Inter-Frame Spacing -SIFS (where ACK time is the actual clock time when the ACK packet is Received, and CTS time is the clock time when CTS packet is received). The Receiver uses the NAV of the RTS packets to obtain the Idle Rate and the difference between the DATA time, RTS time and 3 times SIFS to obtain the capacity (where DATA and RTS times are, respectively, the clock time when DATA packet is received and RTS packet is received). The On looker uses the NAV value according to the existence, or not, of the RTS packet to obtain both the available bandwidth and capacity. If a node in the Onlooker state captures a CTS packet of a communication without capturing the RTS packet, this implies that the communication is suffering from the hidden nodes problem. Thus, the algorithm will only use the NAV from the CTS packet to retrieve the correct values. The total elapsed time represents the difference between the last captured ACK time and the initial time. The packet size considered is the DATA packet size. Figure **??** shows the different approaches for each state while Figure **??** represents the state diagram of the rt-Winf tool. It is possible to observe each state's transitions. When a CTS packet is captured by the Sender, it starts to evaluate the available bandwidth and capacity, while the Receiver starts this process when a RTS packet is received. The Receiver sends the calculated available bandwidth and capacity in an ACK packet to the Sender. When the Sender receives the ACK packet with that information, from the Receiver, compares it with the available bandwidth and capacity that it has previously calculated. If the information received through the ACK packet is lower than the obtained, the sender will use the available bandwidth and capacity received in the ACK packet. Otherwise, the sender will transmit using the available bandwidth and capacity calculated before. This cooperation is a great improvement when compared to IdleGap.

5 b) Probe Packets

If RTS/CTS packets are not present, rt-Winf can use probe packets in order to retrieve the transfer time values. Probe packets can be sent between nodes. These must be UDP generated packets with altered Frame Control IEEE 802.11 header: Type Data and Subtype Reserved. We used packets with Frame Control Type set to 10 (data) and Subtype to 1001 (Reserved). This way the Sender and the Receiver can successfully differentiate these packets from the ordinary data packets. IEEE802.11 standard defines that, for each successfully received packet, it must be sent a MAC ACK packet [3]. The whole process is very similar to the one with the RTS/CTS handshake.

The generated packets are used to retrieve the capacity and available bandwidth values, according to Equation 1 and Equation 2. These packets are only sent before a node wants to start a transmission and in the absence of traffic. This allows the system to initially determine the available bandwidth and capacity. Then, the existing traffic and the MAC layer ACK will be used to trigger the calculations. As NAV values are not correctly defined in DATA packets, rt-Winf uses clock time information to determine the busy time. So, NAV values are not considered in this specific implementation with probe packets. To be fully operational, both Sender and Receiver must be running the rt-Winf mechanism. (1) where TransferTime is equal to ACKTime?DataTime.

In a normal VoIP call using G.711 codec [2], the overhead introduced by this mechanism is _1:66%. For a flow with more than 1Mbps, the overhead is less than _0:15%.

IV.

As TCP-AP tries to retain the end-to-end semantics of TCP, without any modifications on the link and routing layers or the need of cross layer information, as opposed to other proposals such as XCP-Winf or XCP-b, it is important to understand how it reacts under high density and high dynamic environments. TCPAP was developed with the main purpose to improve congestion control in ad-hoc wireless networks.

TCP-AP is a hybrid scheme that introduces the concept of a 4-hop propagation delay, which is the estimated elapsed time between the transmission of a packet by the source and its reception by a node that is 4-hops away. This estimation uses the Round Trip Time (2)

We TCP-AP: Wireless Enhanced TCP-AP TCP-AP EvaluationC = PacketSize Trans f erTime AB = 1 ? ? Trans f erTime TotalElapsedTime * C

control beyond the 4 hops. This is why TCP-AP is considered as a hybrid approach, since it is rate based as well as congestion window based.

In this section, we compare the TCP-AP approach, using ns-2 [1] simulations, against XCP-b, XCP-Winf and WCP. As WCP is also an AIMD and rate based approach, it is a good baseline for comparison purposes. The network scenario used is an adhoc network with nodes varying from 8 to 256 nodes (8,16, **??**2, **??**4, **??**28, **??**56). Nodes are distributed randomly throughout the simulation area.

Flows also vary according to the number of nodes: with 8 nodes we have 4 flows, with 16 nodes we have 8 flows, and so on. The routing protocol used is the Destination-Sequence Distance-Vector (DSDV) [22]. The configured default transmission range is 250 meters, the default interference range is 500 meters, and the channel data rate is 11 Mbps. The performance metrics used are: the throughput, the delay of the transmitted packets, and the number of received packets. Each flow presents a FTP application simulating a large file download. The mobility is emulated through the ns-2 setdest tool to provide a random node movement pattern. We configure setdest with a minimum speed of 10 m/s, a maximum speed of 30 m/s and a topology boundary of 1000x1000 meters. All results were obtained from ns-2 trace files, with the help of trace2stats [18] scripts adapted to our own needs. All simulations last 300 seconds and the simulations are repeated 30 times with different ns-2 seed values. The mean and 95% confidence intervals are presented in the results. From Figure 3, Figure 4 and Figure 5 we can observe that TCP-AP is the one with the worst results, with a poor performance when compared to the other approaches. From the Figures it is possible to observe that TCP-AP presents lower performance results in terms of delay, throughput and received packets. This is due to the fact that TCP-AP is not obtaining correctly the network's maximum capacity, thus not avoiding congestion and not using efficiently the medium. TCPAP is overestimating the available rate producing congestion, As TCPAP rate estimation technique is not using a reliable technique to evaluate the medium, the sender is generating more traffic than the medium supports, resulting in more packets queued, less packets in transit, hence its throughput is decreased, the delay is increased, and the number of received packets is also decreased. Another characteristic of TCP-AP is that it uses the standard AIMD process. This process is not suitable for wireless networks as it overloads the wireless channel. This behavior in conjunction with the estimation technique of TCP-AP results in inaccurate available bandwidth estimations and higher delays.

We can then conclude that TCP-AP is not evaluating correctly and not using efficiently the available bandwidth along the path, obtaining poor throughput and behaving very conservatively, resulting in a low number of received packets and high delay. TCP-AP is also not considering a fair share of the bandwidth to all flows, not using correctly the medium and having a significant degradation of performance.

From Figure 3 it is possible to conclude that XCP-Winf is using accurately the available bandwidth and link capacity information from the MAC layer, improving significantly network performance: it uses more efficiently the medium, resulting in better delay values. XCP-Winf, being a rate-based protocol, where bandwidth and capacity estimation is based on the MAC layer information, and providing node cooperation, it can effectively and quickly adapt to the links conditions, thus, improving network performance and making the network behave more fairly.

From the presented results, it is also possible to observe that WCP has better overall results than TCP-AP. WCP has a rate control mechanism that reacts explicitly to congestion, and a cooperative communication process between neighbor nodes that make WCP to react more efficiently to the network conditions, allowing to have a better medium usage.

XCP-b results are better than the ones obtained by TCPAP. However, its results are worse than the ones obtained by XCP-Winf and WCP. XCP-b, although a rate-based congestion control scheme, it uses complex heuristics based on measuring indirect quantities like queue sizes and the number of link layer retransmissions, to estimate the available capacity. Those direct measures are overestimating the available capacity and bandwidth, specially when the network is heavily utilized, resulting in performance degradation and instable behavior. This is shown by the good XCP-b results when the number of flows is relatively small.

Although TCP-AP scheme is a hybrid scheme of sender rate control and congestion control, TCP-AP is based on two assumptions: the rate control mechanism is efficient and the contention and spatial reuse is accurate. These assumptions may not be effective in some network topologies. This assumption is clearly not effective in high mobility wireless scenarios. The conservativeness of TCP-AP is observed in its throughput (Figure 3) results and received packets (Figure 5). While having good throughput results, they are obtained with less received packets. This is a consequence of using the hybrid scheme for congestion control.

TCP-AP is not using information from the MAC layer: it relies on the transmission of packets at the transport layer. This principle is failing effectively to transmit packets at the MAC layer, making it reacting with poor performance in terms of received packets. As TCP-AP is not relying in an effective available bandwidth and link capacity estimation mechanisms at the MAC layer, the sender assumes that the bandwidth of all links in the path is the same and the medium usage is clearly not efficient. Due to its 4 hop propagation delay assumption, TCP-AP available bandwidth and link capacity estimation is not considering the nodes along the route path, which are the nodes that contend from the available bandwidth along the path. This is specially relevant when we are dealing with a high density and high mobility network, introducing inaccuracy and lack of fairness on the TCP-AP performance.

6 V. Tcp-ap with Rt-Winf

The base TCP-AP considers network and transport layer information (RTT values) for its capacity and available bandwidth estimations. This technique is not very accurate introducing inefficiency to the congestion control process. This was already shown in wired networks, in works [15] and [6], which introduced packet dispersion to analyze the capacity and available bandwidth estimations. This problem is even increased when dealing with wireless networks, since their variation and instability increase.

We claim that it is important to have a crosslayer approach for bandwidth and link capacity estimation: using information provided by several layers, including the MAC layer, it is expected that the congestion control mechanism is more reliable and effective. Therefore, we propose TCP-AP with rt-Winf, which relies on the main functioning principles of TCP-AP, but uses information provided by rt-Winf [25] to determine the link capacity and available bandwidth.

As rt-Winf obtains the link capacity and available bandwidth in the MAC layer, this information has to be accessed by TCPAP through a cross-layer communication process. One example of such crosslayer communication process is the MobileMan [14] cross-layered network stack. This communication system uses a shared database architecture, with a set of methods to get/insert information from/in the database accessible by all protocol layers.

Our approach, when compared to the base TCP-AP, changes the way each node calculates the 4hop delay (FHD) and the average packet queuing delay per node (t q), with the rt-Winf link capacity and available bandwidth values. Thus, (3) where T RTT represents the RTT value, h represents the number of hops between the sender and receiver, Sdata is the size of the data packet and Sack is the size of the ACK packet. Finally, CWin f corresponds to the rt-Winf link capacity. The previous equation allows to update the 4-hop delay (FHD) by: (4) where AB Winf is the rt-Winf available bandwidth.

Considering that a high density and high mobility network suffers from a large number of collisions, rt-Winf mechanism was updated with the effect of collision probability. Notice that rt-Winf works on the IEEE 802.11 [3] MAC layer that uses the Distribution Coordination Function (DCF) as the access method. This function is based on the CSMA-CA principle, in which a host wishing to transmit senses the channel, waits for a period of time, and then transmits if the medium is still free. If the packet is correctly received, the receiving host sends an ACK frame after another fixed period of time. If the ACK frame is not received by the sending host, a collision is assumed to have occurred. Therefore, to improve efficiency and reliability of TCPAP with rt-Winf, collision probability is accounted for. When a sender cannot transmit due to collision, the back off mechanism is activated. This mechanism is also consuming bandwidth that is not really used by the channel. This unused channel contention bandwidth can be allocated as an extra bandwidth. This extra bandwidth, C extra , is defined by:

7 Global Journal of Computer Science and Technology

Volume XVI Issue VI Version I 7 Year 2016 ( )E

We TCP-AP: Wireless Enhanced TCP-APt q = 1 2 (T RT T h ? S data ? S ack C Win f) FHD = 4 × (t q + S data AB Win f) (5)

where T DIFS is the IEEE 802.11 DCF Inter frame Space, T backo f f is the medium backoff time, T m is the time between the transmission of two packets and W is the channel bit rate. The collision probability (P c) can then be defined as 1-C extra . Applying this result to the rt-winf inference mechanism, the available bandwidth (AB) becomes:

(6)

8 VI.

In a wireless network, nodes along a multi-hop path (NP) contend among themselves for access to the medium, i.e, they contend for available bandwidth. To obtain the contention of nodes along the path, it is important to know the contention count of each node. The contention count at a node is the number of nodes on the multi-hop path that are located within carrier sensing range of the given node, and can be obtained as described in [26]. Considering that TCP-AP only implements adaptive pacing at the sender side, available bandwidth and capacity estimation must take into consideration nodes along the path between the source and the sink, that is, the bandwidth contending successors and predecessors on the route path. However, this is not true in TCP-AP, since it is considering only 4-hop neighborhood for these contending estimations. Therefore, to eliminate this inaccuracy, we changed TCP-AP with rt-Winf to use a coefficient (R is the unused bandwidth) that represents the proportion of bandwidth contention among other nodes on the path, thus, maximizing the throughput while guaranteeing fairness. If we consider NP as all nodes along the path and if NP?1 is equal or less than 4, then TCP-AP with rt-Winf is kept unchanged; if NP?1 is higher than 4, then the FHD equation, now called the hop delay (HD) is updated to: Where then,

Algorithm 1 shows the pseudo-code of an WE TCP-AP source node.

As R represents the unused bandwidth due to node contention and queue management along the path, it introduces the fairness factor allowing an improved fair share of the available bandwidth among all contending nodes, not only the ones within the 4-hop propagation delay, improving WE TCP-AP behavior and making it behave more accurately.

9 VII. Simulation Results

This section presents simulation results of our proposed congestion control mechanism. The results are obtained using the ns-2 simulator [1]. The underlying rt-Winf mechanism is configured with enabled RTS/CTS/ACK handshake packets. The proposed mechanism is evaluated against the base TCP-AP protocol, WCP and XCP-Winf. Two different scenarios were used: the same ad-hoc scenario presented in section V, and a wireless mesh topology scenario that is presented to understand how the new proposals behave under different conditions. This scenario is

AP. Again, all simulations last 300 seconds and the simulations are repeated 30 times with different ns-2 seed values. The mean and 95% confidence intervals are presented in the results.

We TCP-AP: Wireless Enhanced TCP-AP Wireless Enhanced TCP-AP -we TCP-APAB = P c × AB Win f ? AB = (1 ?C extra W) × AB Win f HD = FHD × R R = 1 + 1 NP W HD = 4 × (NP + 1 NP ×W) × (t q + S data AB Win f)

a) TCP-AP with rt-Winf Figure 6, Figure 7 and Figure 8 show the performance metrics for the mesh topology scenario. From the observation of the results, it is possible to conclude that TCP-AP with rt-Winf integrated clearly improves TCP-AP performance behavior, but its performance is still below the one of XCP-Winf. TCP-AP with rt-Winf is only taking into consideration rt-Winf information for the last 4 hop nodes; TCP-AP, as opposed to XCP-Winf, uses the standard behavior of TCP for the other hops of the network, considering that all links have the same bandwidth.

Another important drawback of TCP-AP with rt-Winf is the fact that it does not have a fairness module, resulting in a more conservative and less fair operation. The fairness module is a native mechanism used by XCP-Winf. As TCP-AP with rt-Winf uses, in most of its functioning, the standard AIMD process of TCP and is not entirely using the available information between the source and the sink, its results are not similar to the ones of XCP-Winf. XCP-Winf also relies on the overall node path interaction, using a cooperative approach to obtain the best available bandwidth and link capacity usage. In TCP-AP with rt-Winf, as the number of nodes or flows increases, it uses conservative mechanisms, reducing its performance especially concerning received packets.

WCP obtains better results than TCP-AP with rt-Winf. WCP uses explicit congestion information between nodes that trigger rate changes, making it behave with good efficiency and fairly. As XCP-Winf uses the rt-Winf mechanism as its base estimation tool, it has a precise feedback communication mechanism between all the nodes along the path using total network cooperation, and it is able to better use the channel with less losses, resulting in a more efficient and accurate behavior.

Figure **??**, Figure **??**0 and Figure 11 show the results for the ad-hoc topology scenario. In this scenario, we can see that rt-Winf clearly improves TCP-AP performance, compared to TCPAP and WCP. However, TCP-AP with rt-Winf still reflects some With the increase of the number of flows, TCP-AP with rt-winf becomes less efficient, as it is only relying on the 4-hop propagation delay and the AIMD process, not considering the entire network topology for its rate changes. This is shown by being able to obtain good throughput results, compared to XCP-Winf, when the network is not heavily loaded. When increasing the number of nodes, number of flows and the mobility density, TCP-AP with rt-Winf becomes more inefficient, reducing significantly its throughput and the number of received packets when compared to the other approaches. TCP-AP with rt-Winf is also more fair than TCP-AP to mobility changes, but it still shows an unstable behavior. WCP has overall good results: although being an hybrid approach, it uses a more effective congestion and control interval, as all nodes within the congestion neighborhood mark packets with congestion indicators, triggering rate reductions more efficiently at the source.

10 b) WE TCP-AP

This section presents the simulation results of WE TCP-AP in both mesh and ad-hoc scenarios.

Figure **??**2, Figure 13 and Figure 14 show the performance metrics. In terms of received packets, as observed in Figure 14, it is possible to see that WE TCP-AP is able to use more efficiently TCP-AP has a very conservative behavior, as it allows a good throughput with less received packets. This behavior is clearly improved with WE TCP-AP. The delay values, in Figure 13, are also reduced reinforcing the fact that this new proposal is much more efficient and fair, with better medium usage, than the base protocol. The better results are still obtained by XCP-Winf, but it is closely followed by WE TCP-AP: it is clear that the use of MAC layer information and the node path contention count is making WE TCP-AP to react more efficiently to the network dynamics.

Figure 15, Figure 16 and Figure 17 show WE TCP-AP results in the ad-hoc network scenarios, as defined before. the medium, as it can transmit more packets increasing overall throughput results. More received packets means that more transmissions are allowed, thus WE TCP-AP is behaving more fairly. With these improvements, the network can transmit with a higher rate and incurring less losses. As more packets are transmitted, more throughput is obtained and the medium is better and more efficiently used. This allows node path contention clearly improves base TCP-AP performance behavior. It is possible to conclude that, with more nodes and flows in the network, WE TCP-AP is more efficient than the standard TCP-AP proposal. XCP-Winf uses an explicit congestion control notification mechanism for an accurate rate change and relies in the link capacity at the MAC layer information; in this scenario, it is also able to operate more efficiently than WE TCP-AP, specially concerning the number of received packets. WE TCP-AP is not a pure rate-based congestion control mechanism with explicit feedback, thus it is not reacting quickly to network changes. The AIMD process of WE TCP-AP still introduces some instability and behavior problems.

WE TCP-AP, as opposed to TCP-AP, is considering a fair share of the unused bandwidth, that results from the use of the node path contention count, making it behave more efficiently and For a better understanding of how the factor R is influencing WE TCP-AP behavior, a central network chain scenario was defined. It must be noted that the standard TCP-AP 4-hop propagation delay assumes that ”every fourth node can transmit in a multi-hop chain topology”. On this scenario, it was used the proposed version of WE TCP-AP and the TCP-AP with rt-Winf version. The chain scenario consists of a network divided in three parts. Figure 18 depicts the network topology with four chains of nodes. The application used simulates a FTP transfer. The results are shown in Figure 19, Figure **??**0 and Figure **??**1. The presented results clearly show that, with the increase of the chain nodes, TCP-AP with rt-Winf has worse results: it becomes less efficient an less accurate, as it is not considering the unused share of bandwidth. WE TCP-AP, on the other hand, is more accurate, since the available bandwidth and capacity As TCP is the most used and deployed congestion control protocol on the Internet, it is important, as described on [27], to analyze how WE TCP-AP flows interact and compete with TCP. To analyze how friendly WE TCP-AP is, we use the average data rate over time for each flow, thus allowing to observe how bandwidth is being managed between TCP and the WE TCPAP proposal. This is called the utility of a congestion control mechanism against TCP.

The evaluation scenarios consist of a 1000mx1000m area, divided in three distinct parts. In the left side area, with 250mx250m, we have two mobile source nodes: one source node is configured to use only the standard TCP, and the other source We have defined two evaluation scenarios. One scenario contains each source generating 8 FTP flows, with packets of 1500 bytes. In the other scenario we have each source generating sixteen FTP flows. The simulations last 120 seconds. The obtained results are shown in Figure 22 and Figure 23.

From the utility results, it is possible to observe that, on both situations, the TCP flow grows faster and gains more bandwidth on the beginning. However, as WE TCP-AP is a hybrid approach, keeping unchanged the AIMD process of TCP and being updated with an evaluation and measurement process, it quickly adjusts to TCP behavior, thus, allowing a fair share of network resources.

11 VIII.

This paper proposed a new approach to congestion control, based on TCP-AP and a new wireless inference mechanism, rt-Winf. rt-Winf measures the wireless capacity and the available bandwidth of wireless links, and feeds this information to TCPAP, through a cross-layer communication process. Two different improvements were also considered on the new approach: the awareness of collision probability on the available bandwidth approach, and the node path contention count on the 4-hop propagation delay approach.

The performance evaluation study of the proposed congestion control mechanism shows that the integration of rt-Winf and the proposed enhancements allow to make TCP-AP behavior more efficient, resulting in better overall network performance. Using rt-Winf, that works in the MAC layer, it is possible to perform link capacity and available bandwidth calculations without interfering in the network dynamics, allowing to significantly improve TCPAP performance. The node path contention also significantly improves TCP-AP performance, with more noticeable results for larger chains of nodes. This congestion control mechanism is denoted as WE TCP-AP.

As future work, we plan to work on the wider evaluation of the congestion control approach, using for example, new comparison baselines and protocols. An effort will also be made in creating a future test bed for understanding how the proposed mechanism is affected by different conditions and parameters, in a real environment. Year 2016 ( ) [1] [2] [3]

[2]& & & &'(&) *)) ' +

Figure 1:

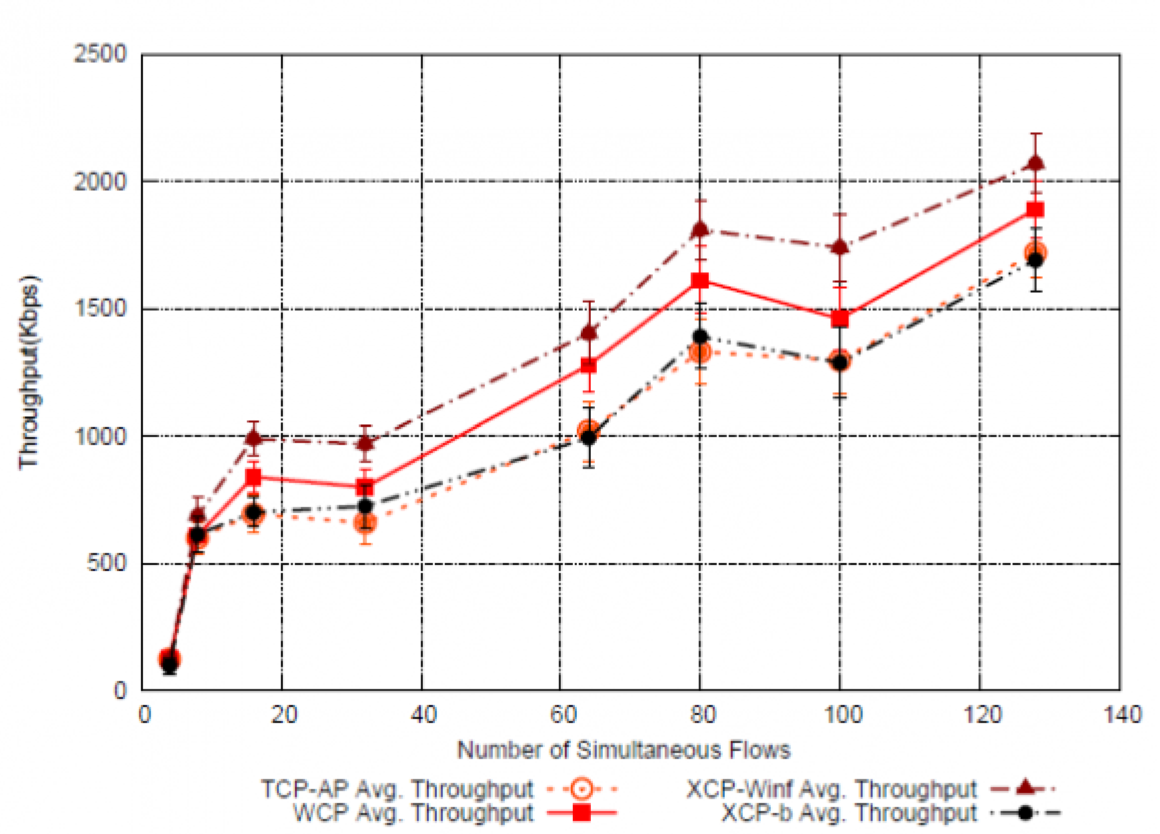
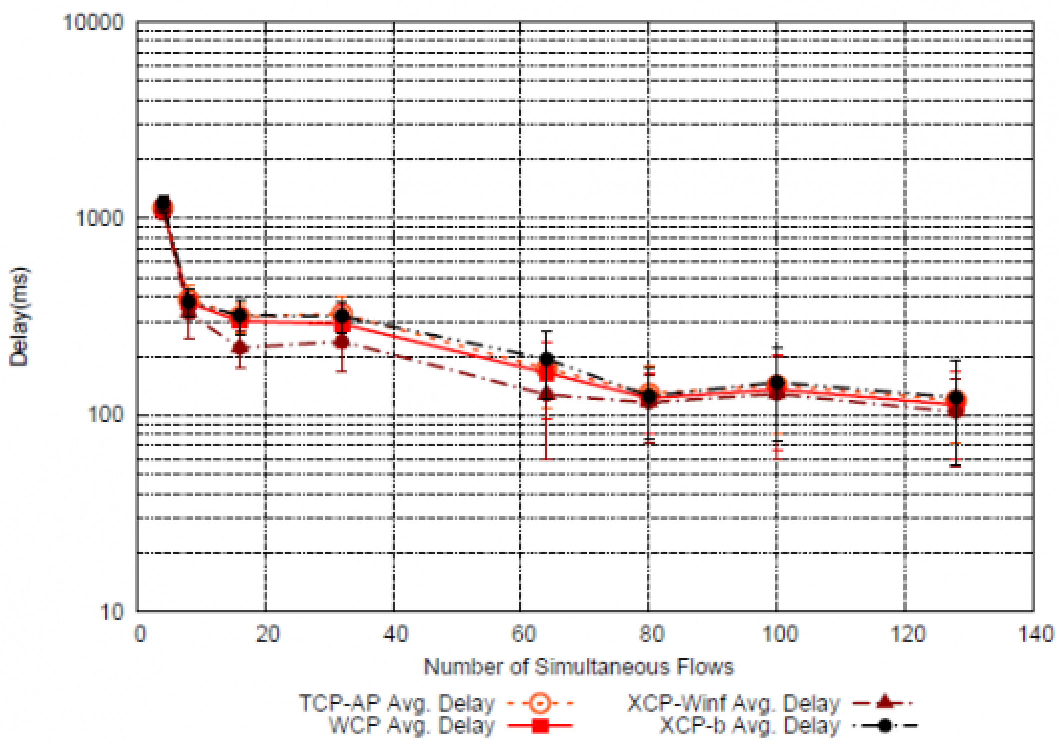


Figure 2:



3

Figure 3: Figure 3 :

4

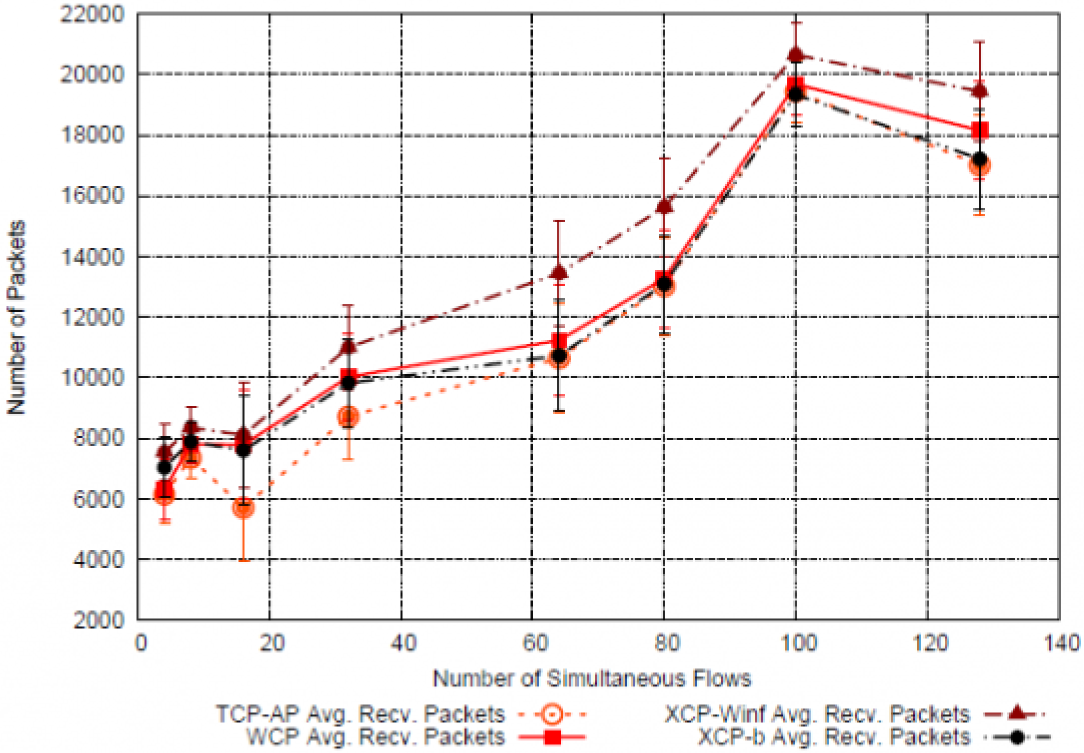


Figure 4: Figure 4 :

5

Algorithm 1: *WE TCP-AP* Source Node Operations.

foreach *ACK packet* **do**

  Node estimates node path (NP) from MAC ACK

  Node computes NP-1

  if $NP - 1 <= 4$ **then**

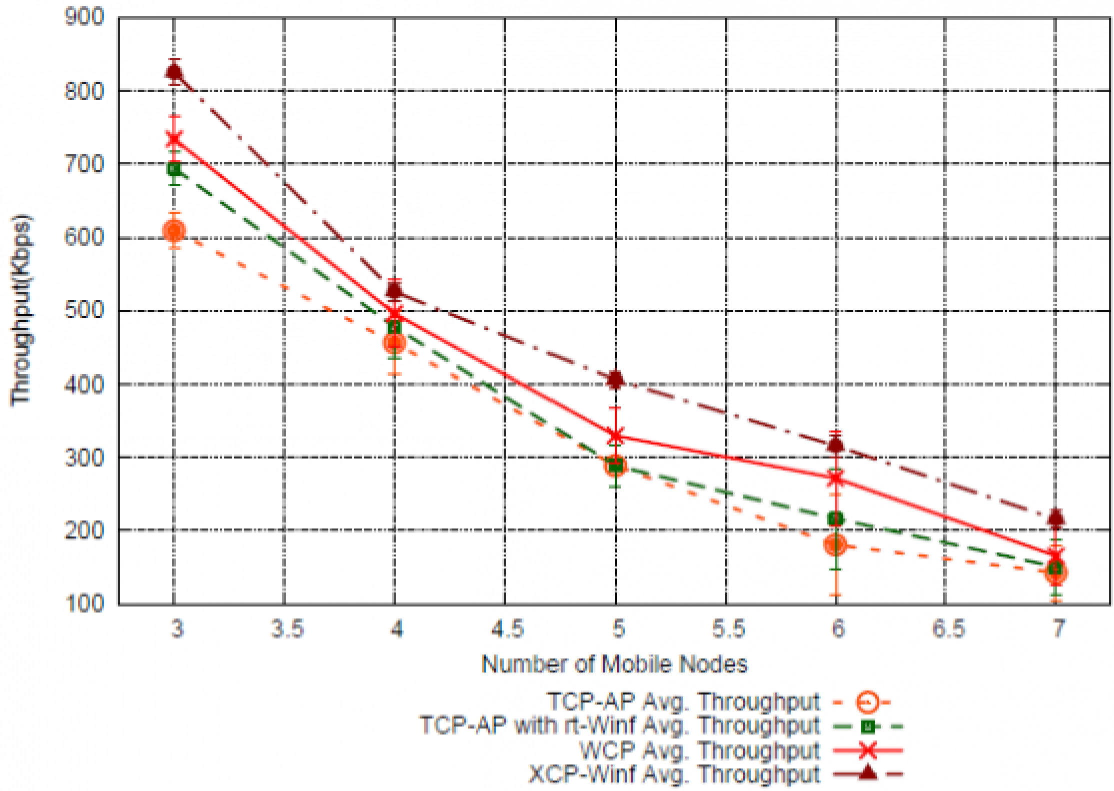
    $HD = FHD$;

  else

    $R = \frac{NP+1}{NP \times W}$;

    $HD = R \times FHD$

Figure 5: Figure 5 :



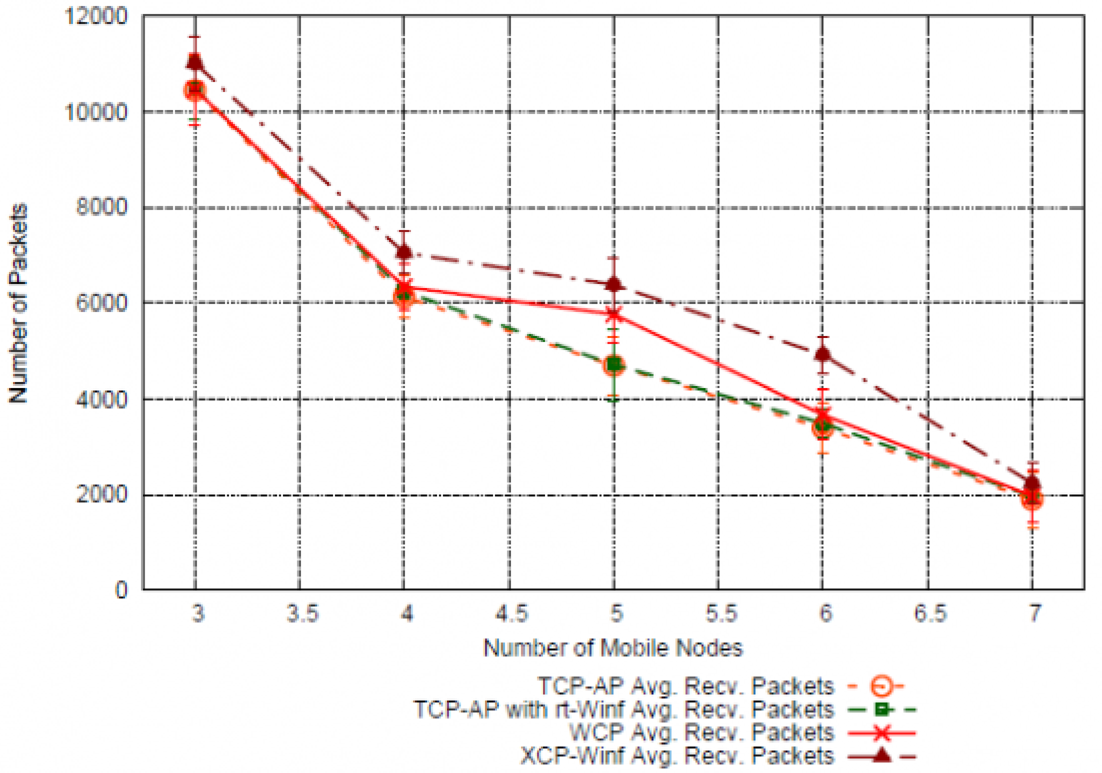
6

Figure 6: Figure 6 :

$$C_{extra} = \left(\frac{T_{DIFS}}{T_{backoff}} T_m\right) \times W$$

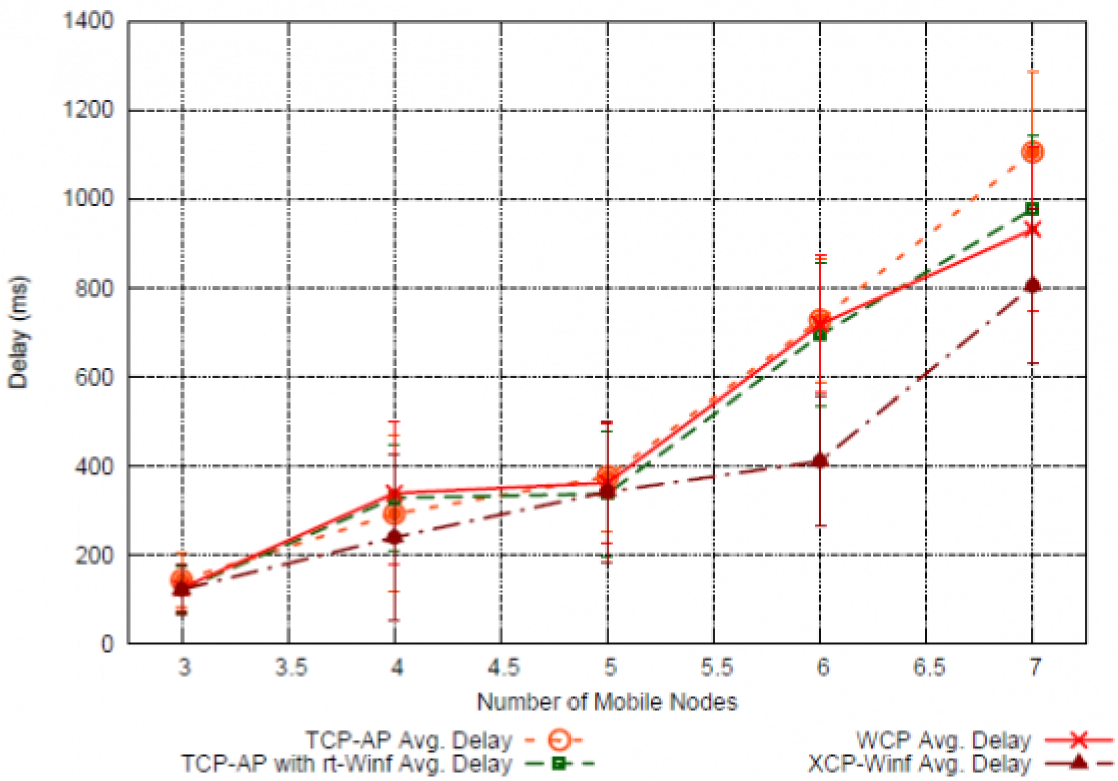
8

Figure 7: Figure 8 :



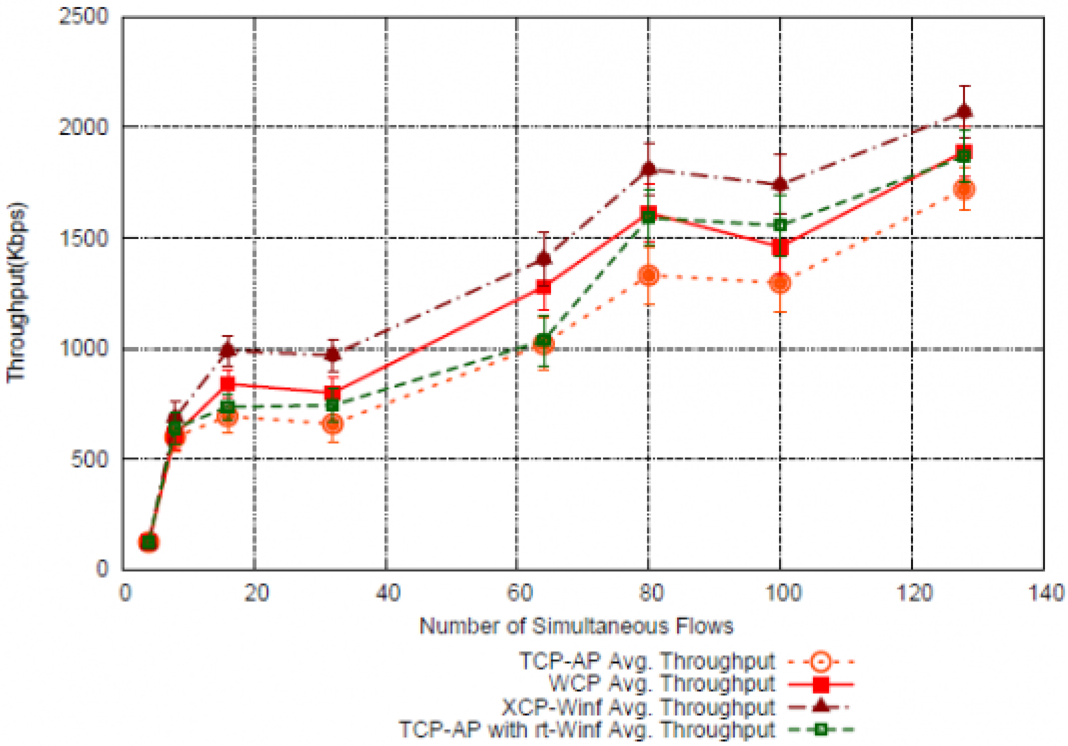
7

Figure 8: Figure 7 :



910

Figure 9: Figure 9 :Figure 10 :



11

Figure 10: Figure 11 :

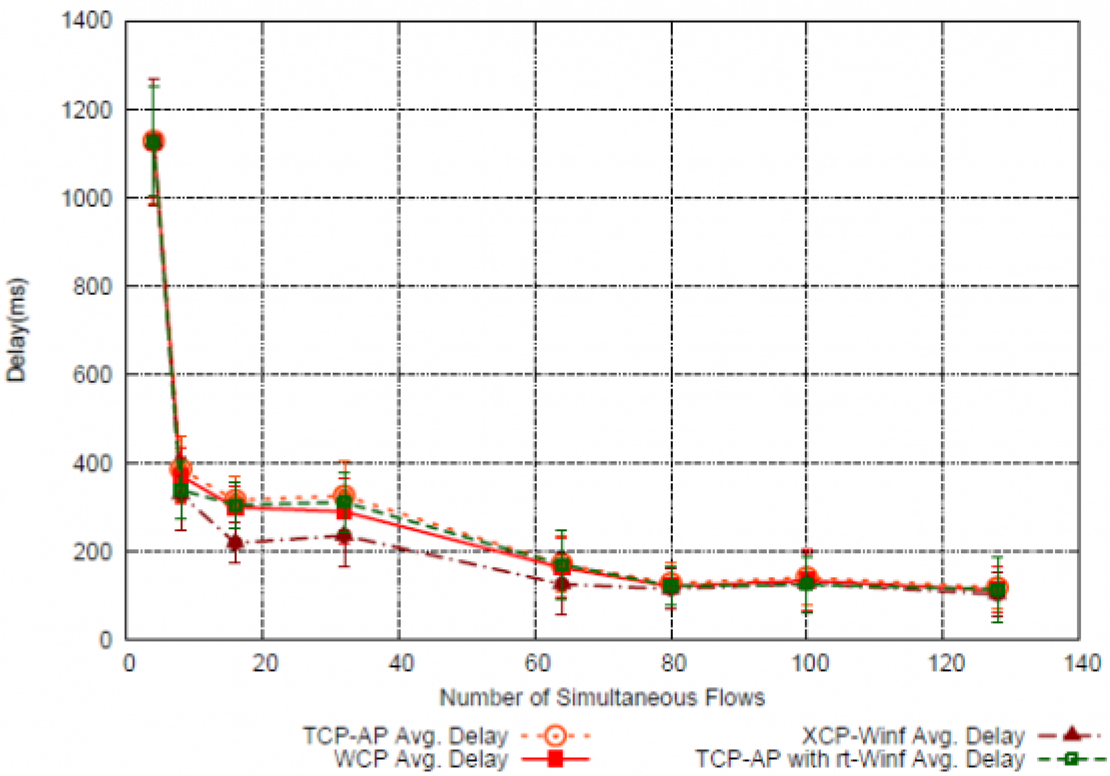
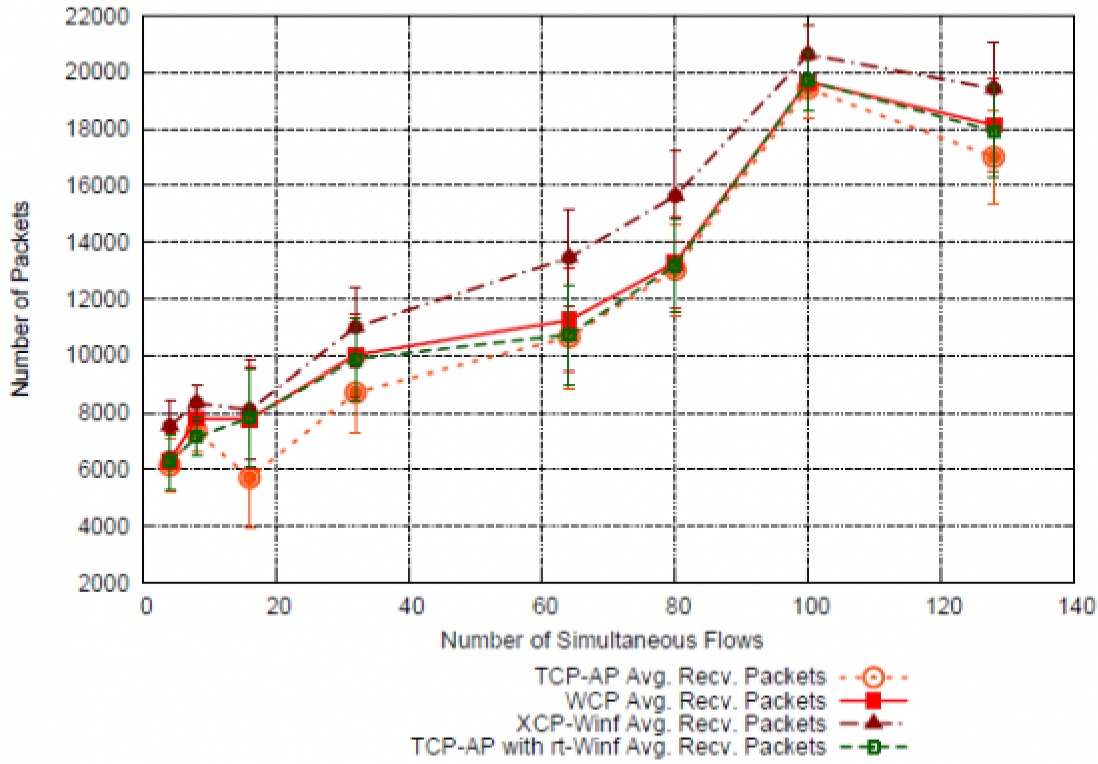
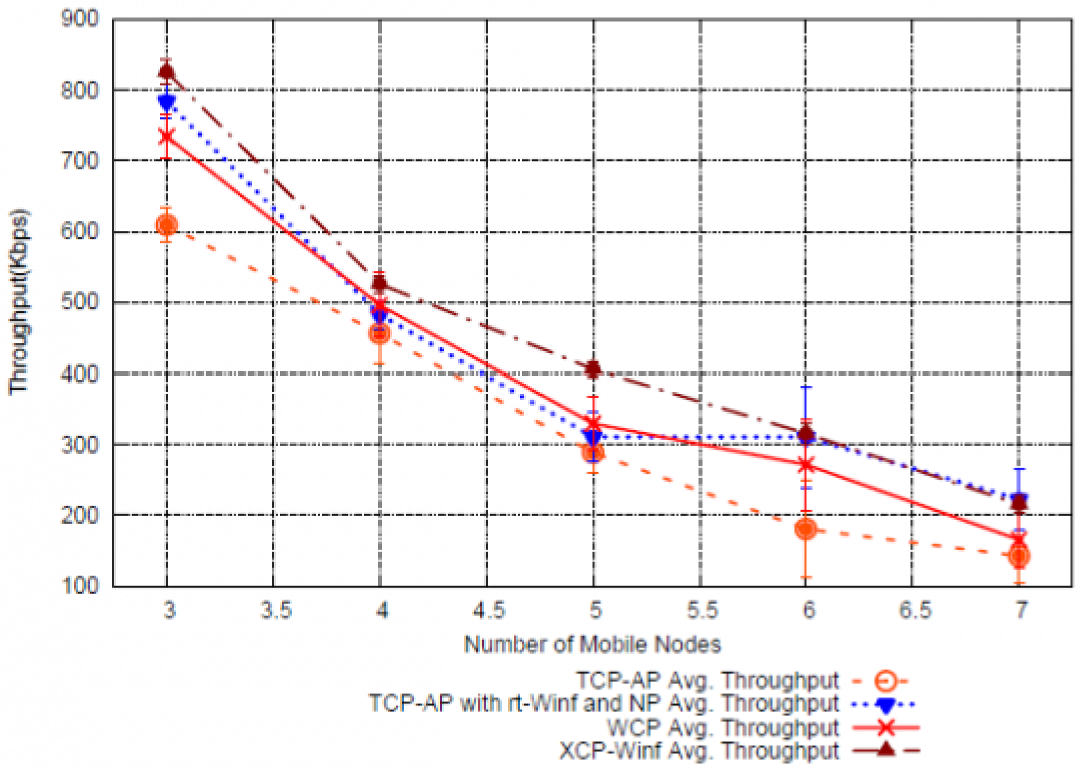


Figure 11:



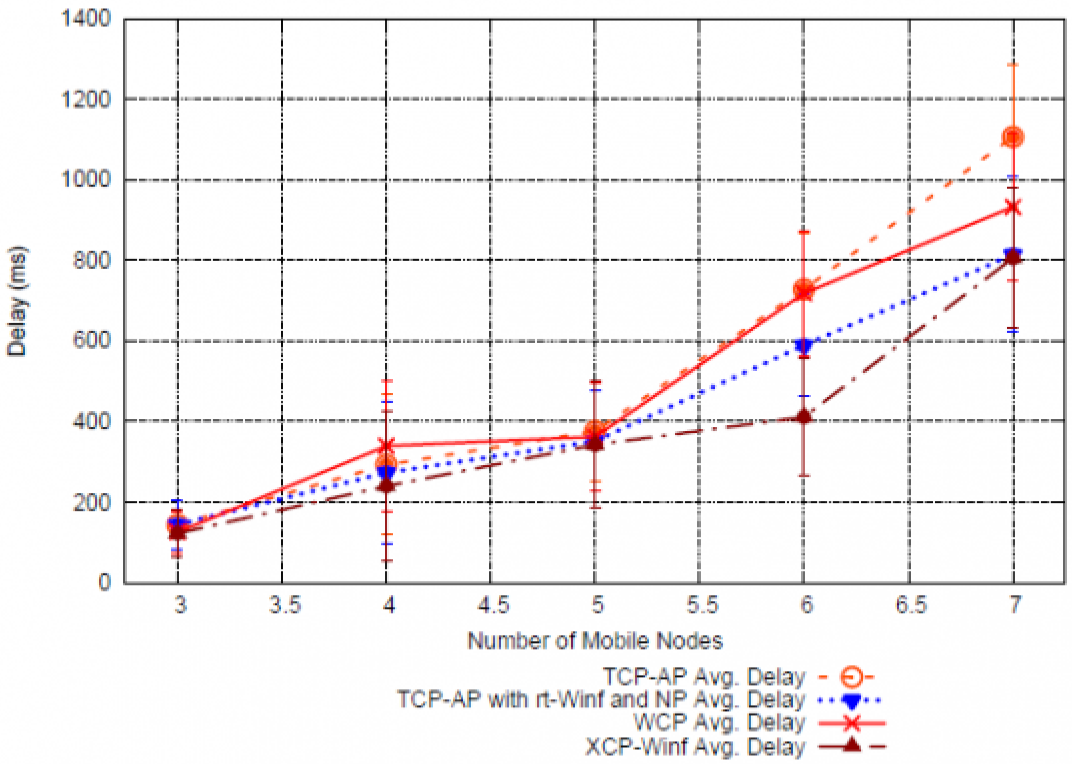
14

Figure 12: Figure 14 :



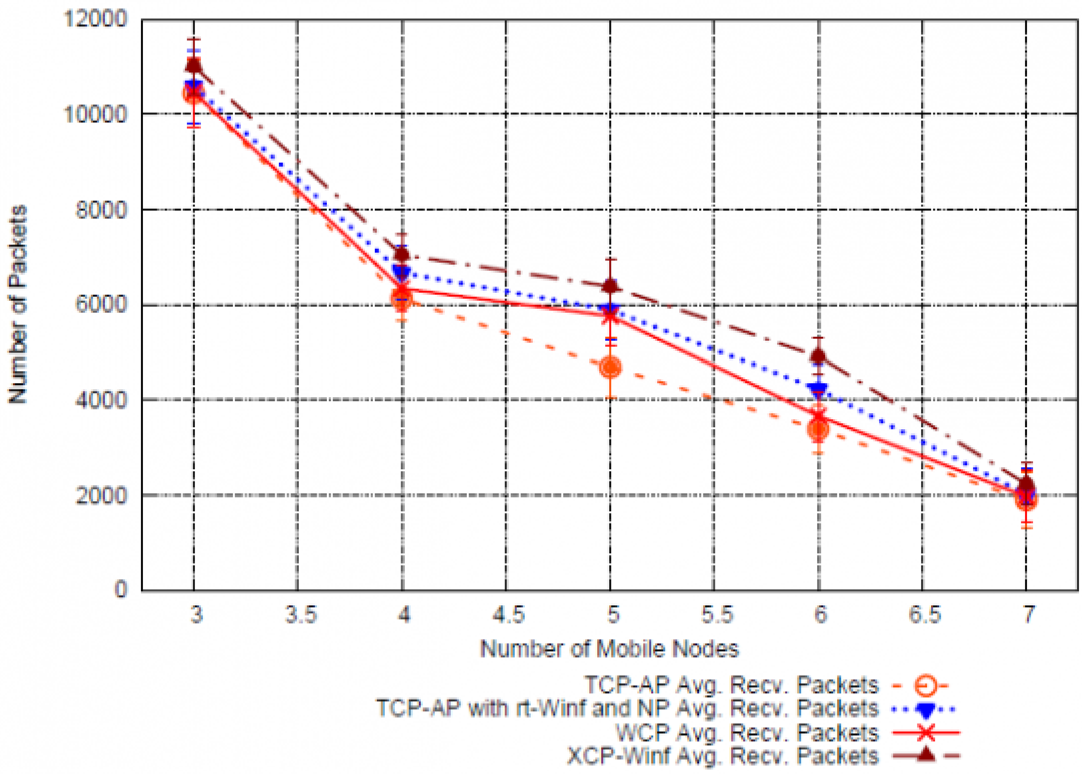
15

Figure 13: Figure 15 :



16

Figure 14: Figure 16 :



19

Figure 15: Figure 19 :

18

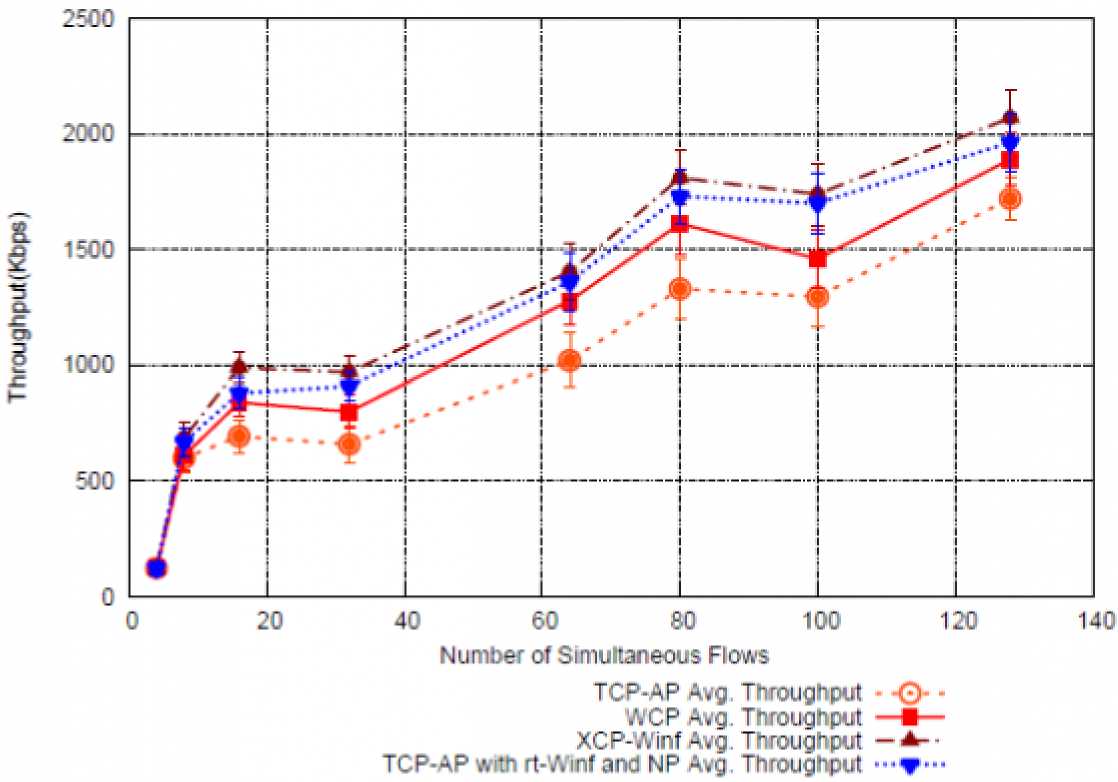


Figure 16: Figure 18 :

22

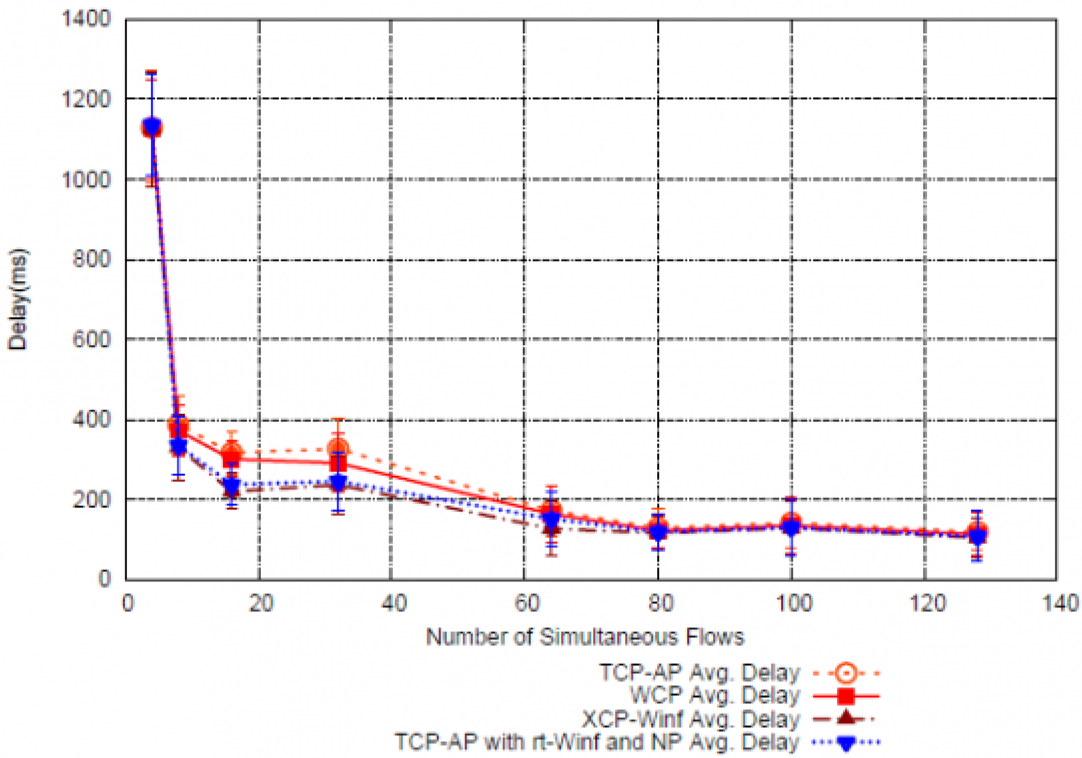
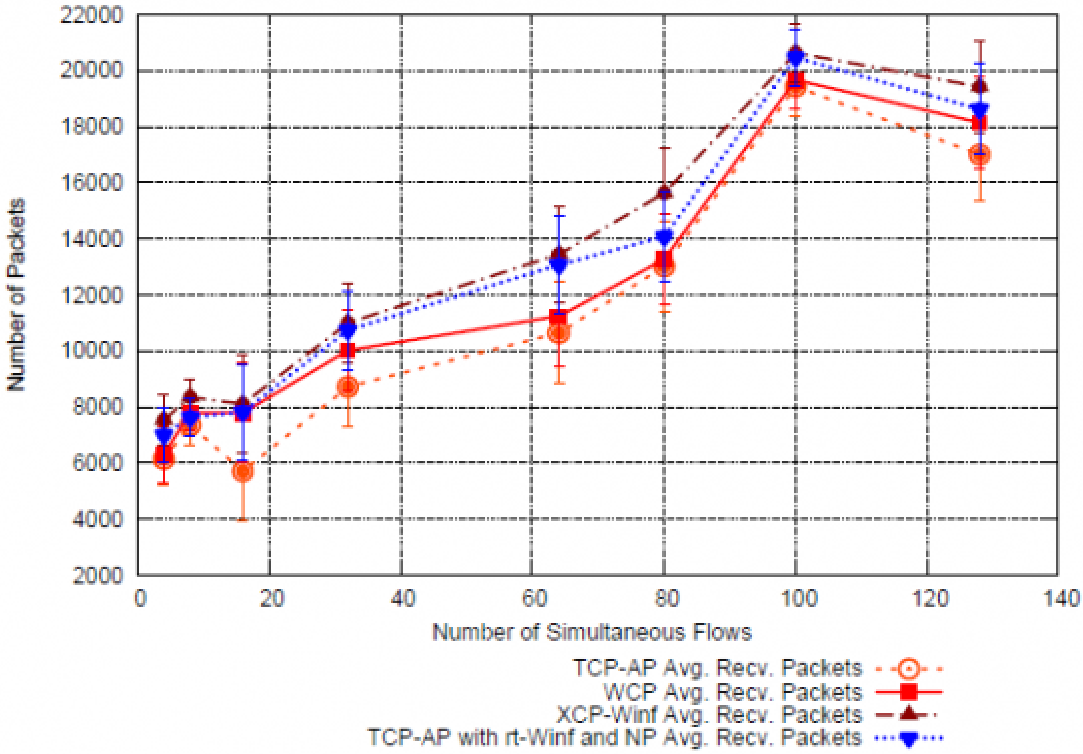


Figure 17: Figure 22 :



23

Figure 18: Figure 23 :

I

We TCP-AP: Wireless Enhanced TCP-AP

similar performance than rate-based in end-to-end TCP compatible approaches. Cross-Layer Control Param

WCPCap	Rate	
CNA	Window	
EZ-Flow	YesBuffer	
HOP	Window	
TCP-AP	Window	Yes
XCP-b	YesBuffer	

[Note: same duration. The RTS/CTS packets have accurate duration values, which can be used in the calculations.]

Figure 19: Table I :

[Fiore ()] , M Fiore . http://www.tlc-networks.polito.it/fiore/index.html 2009. p. e2.

[11 -IEEE standard for information technology -telecommunications and information exchange between systems -local and metrop ‘11 -IEEE standard for information technology -telecommunications and information exchange between systems -local and metropolitan area networks specific requirements. part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications’. http://standards.ieee.org/getieee802/download/802.11-2007.pdf *IEEE std* 2007. 802.

[Abrantes and Ricardo ()] ‘A simulation study of XCP-b performance in wireless multi-hop networks’. F Abrantes , M Ricardo . *Proceedings of the 3rd ACM workshop on QoS and security for wireless and mobile networks, Q2S Winet ’07*, (the 3rd ACM workshop on QoS and security for wireless and mobile networks, Q2S Winet ’07) 2007. ACM.

[Brenner ()] *A technical tutorial on the IEEE 802*, P Brenner . http://sss-mag.com/pdf/802n11tut.pdf 1997.

[Bandwidth estimation in wireless LANs for multimedia streaming services Advances in Multimedia ()] ‘Bandwidth estimation in wireless LANs for multimedia streaming services’. *Advances in Multimedia* 2007. 2007. (1) p. .

[Barreto ()] L Barreto . 10.5296/npa.v7i2.7677. *TCP-AP enhanced behaviour with rt-Winf and node count: boosted-TCP-AP. Network Protocols and Algorithms*, 2015. 7.

[Li et al. ()] ‘Blockswitched networks: A new paradigm for wireless transport’. M Li , D Agrawal , D Ganesan , A Venkataramani . *6th ACM/USENIX Symposium on Networked Systems Design and Implementation*, 2009. 2009.

[Katabi et al. ()] *Congestion control for high bandwidth-delay product networks*, D Katabi , M Handley , C Rohrs . 2002. ACM SIGCOMM.

[Conti et al. ()] ‘Crosslayering in mobile ad hoc network design’. M Conti , G Maselli , G Turi , S Giordano . 10.1109/MC.2004.1266295. DOIhttp://doi.Ieeecomputersociety.org/10.1109/MC.2004.1266295 *Computer* 2004. 37 p. .

[Sanzgiri et al. ()] ‘Determining intraflow contention along multihop paths in wireless networks’. K Sanzgiri , I Chakeres , E Belding-Royer . *Proceedings. First International Conference on*, (First International Conference on) 2004. 2004. 2004. p. . (Broadband Networks)

[Xu et al. ()] ‘Enhancing TCP fairness in ad hoc wireless networks using neighbor hood red’. K Xu , M Gerla , L Qi , Y Shu . *Proc. ACM MobiCom*, (ACM MobiCom) 2003. ACM Press. p. .

[Aziz et al. ()] ‘Ez-Flow: Removing turbulence in IEEE 802.11 wireless mesh networks without message passing’. A Aziz , D Starobinski , P Thiran , A E Fawal . *5th International Conference on emerging Networking EXperiments and Technologies*, 2009.

[Perkins and Bhagwat ()] *Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers*, C E Perkins , P Bhagwat . 1994. ACM SIGCOMM.

[Incentive compatible medium access control in wireless networks ()] *Incentive compatible medium access control in wireless networks*, 2006. ACM GAMENETS.

[Dovrolis et al. ()] ‘Packetdispersion techniques and a capacity-estimation methodology’. C Dovrolis , P Ramanathan , D Moore . *IEEE/ACMTrans. Netw* 2004. 12 p. .

[Res et al.] ‘rt-winf: Real time wireless inference mechanism’. B Res , L Barreto , S Sargento . *IEEE Globecom 2010 Workshop on Mobile Computing and Emerging Communication Networks (MCECN2010)*,

[Young Jang et al. ()] ‘Simple yet efficient,transparent airtime allocation for tcp in wireless mesh networks’. K Young Jang , K Psounis , R Govindan . URLhttp://www-bcf.usc.edu/\protect\char"0024\relaxnsim\protect\char"0024\relaxkpsounis/Papers/cna.pdf *6th International Conference on emerging Networking Experiments and Technologies (CoNEXT)*, 2010.

[Elrakabawy et al. ()] ‘TCP with pacing for multihop wireless networks’. S M Elrakabawy , A Klemm , C Lindemann . *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing MobiHoc ’05*, (the 6th ACM international symposium on Mobile ad hoc networking and computing MobiHoc ’05) 2005. ACM. p. .

[Xu ()] *TCP-friendly CBR-like Rate Control. Network Protocols*, J F L Xu . 10.1109/ICNP.2008.4697036. 2008. 2008. 2008. p. .

[The network simulator -ns-2 ()] *The network simulator -ns-2*, http://www.isi.edu/nsnam/ns/ 2001.

[Postel ()] ‘Transmission Control Protocol’. J Postel . URLhttp://www.rfc-editor.org/rfc/rfc793.txt *RFC* 1981. 793.

[Rangwala et al. ()] ‘Understanding congestion control in multi-hop wireless mesh networks’. S Rangwala , A Jindal , K Y Jang , K Psounis , R Govindan . *Proceedings of the 14th ACM international conference on Mobile computing and networking, MobiCom ’08*, (the 14th ACM international conference on Mobile computing and networking, MobiCom ’08) 2008. ACM. p. .

[What do packet dispersion techniques measure? ()] *What do packet dispersion techniques measure?*, 2001. IEEE INFOCOM.

[Dukkipati and Mckeown ()] *Why flow-completion time is the right metric for congestion control*, N Dukkipati , N Mckeown . 2006. ACM SIGCOMM.

[Akyildiz et al. ()] 'Wireless mesh networks: a survey'. I F Akyildiz , X Wang , W Wang . *Computer Networks* 2005. 47 (4) p. .

[Barreto and Sargento ()] *XCP-winf and RCP-winf: A new approach for wireless congestion control*, L Barreto , S Sargento . URLhttp://plataformas.esce.ipvc.pt/lbarreto/XCPwinf.pd 2014. Submitted to Springer Journal of Network and Systems Management

[Barreto and Sargento ()] 'XCP-winf and RCP-winf: Congestion control techniques for wireless mesh networks'. L Barreto , S Sargento . *IEEE International Conference on Communications*, (Kyoto, Japan) 2011. 2011.