PERFORMANCE EVALUATION OF TCP ALGORITHMS ON HYBRID WIRED/WIRELESS LAN TEST-BED

Apostolos Kotsiolis, Christos Antonopoulos and Stavros Koubias
Applied Electronics Laboratory, Department of Electrical & Computer Engineering
University of Patras, Rio Campus, Patras, Greece

Keywords: Transport Layer, TCP Algorithms Evaluation, WAN Test-bed, Linux, Congestion and Admission Control, Wireless/Wireline Networks.

Abstract: This paper aims to provide comparative performance evaluation of various available TCP algorithms over a hybrid local area network comprising by, both wired and wireless sections. Although TCP originally was designed aiming to control internet traffic, LAN networks increasingly depend on new TCP protocol versions to provide congestion control, fairness and optimum resource utilization. Additionally wireless LANs' popularity increase exponentially. Based on the aforementioned observations, the objective of this paper is two-fold. Firstly, to present an adequate test-bed enabling, through a significant number of experiments, an accurate performance evaluation of five different TCP versions available by a typical Linux distribution. Secondly based on the conclusions extracted, propose efficient, yet low complexity modifications, able to improve network performance of the considered scenarios.

1 INTRODUCTION

Transmission Control Protocol (TCP) was designed to provide a reliable, connection-oriented, transport service over the unreliable service provided by the Internet Protocol (IP). The effect of the TCP/IP protocol combination was so profound that formed the backbone of the existing wired computer networks with the Internet being the most characteristic example.

During the last decade there has been a rapid increase on the number of Local Area Networks (LANs) implementations covering a large range of end user demands. Furthermore, the most significant advance of networking during the last years is expressed by wireless connectivity providing a whole new area of networks that lead to an even bigger expansion of LANs' size and popularity (Mohaparta, 2005). However, on the other hand the nature of wireless connection has been the main reason for TCP congestion control suboptimal behaviour in WLANs, as TCP is not well suited to error prone links (Mascolo, 2001).

These reasons lead to the evolution of TCP protocol family which had to maintain its significant role as the robust and reliable Transport Layer Protocol in addition to providing solutions to the congestion control problems arisen by the growing complexity of LANs and the Internet.(Mascolo, 2001), (Grieco, 2004).

The main objective of this paper, is comparatively evaluating five of the most popular TCP congestion control algorithms that are provided by the Linux kernel, (BIC, New Reno, H-TCP, Vegas and Westwood+) on a typical network environment. A critical difference of the presented approach, as opposed to many already presented research efforts focusing on simulation based evaluation, is that the experiments were carried out on a real local wired/wireless network. Additionally measurements were extracted directly from the Linux kernel as well as by using well known network analyzing applications, for verification as well as system level measurements, as opposed to relying solely to application level measurements. Our goal is to evaluate the performance and behaviour of these algorithms that are designed aiming to control network traffic on a number of large data transferring scenarios taking place on a hybrid wired/wireless LAN test-bed.

Additionally, based on these measurements, modifications are proposed concerning a specific algorithm evaluated. Through these measurements,

Kotsiolis A., Antonopoulos C. and Koubias S. (2010).
PERFORMANCE EVALUATION OF TCP ALGORITHMS ON HYBRID WIRED/WIRELESS LAN TEST-BED.
In *Proceedings of the International Conference on Data Communication Networking and Optical Communication Systems*, pages 73-79
DOI: 10.5220/0002981900730079

potentially significant benefits are identified and presented.

The rest of this paper is structured as follows: Section 2 presents the TCP versions under evaluation. In section 3 the test-bed setup is described followed by the performance analysis in section 4. Section 5 includes the presentation and evaluation analysis of the proposed modifications while in section 6 the main conclusions are summarized.

2 PROMINENT TCP PROTOCOLS

Five congestion control algorithms are evaluated through numerous performance measurements including, BIC, New Reno, H-TCP, Vegas and Westwood+. The following brief description aims to reveal each one's philosophy and specialized mechanisms.

Binary Increase Control TCP (BIC) (Xur, 2004) is designed focusing on fast long-distance networks and aims to satisfy three main performance criteria: RTT fairness, TCP friendliness and scalability.

New Reno TCP (Floyd, 1999) is a modification of the classic Tahoe/Reno algorithm that improves retransmission during the fast recovery phase of TCP Reno.

H-TCP (Leith, 2004) is a congestion control algorithm presented by the Hamilton Institute suitable for deployment in high-speed and long-distance networks.

Vegas TCP (Brakmo, 1995) was the first attempt to depart from the loss-driven paradigm of the TCP by introducing a mechanism of congestion detection before packet losses.

Westwood+ TCP algorithm (Mascolo, 2001), (Grieco, 2004) is based on end-to-end estimation of the bandwidth availability along the TCP connection path.

Due to the specific features and characteristics of the aforementioned TCP protocols are considered in the following evaluation process.

3 TEST-BED SETUP & CONFIGURATION

As depicted in "Fig. 1" two different topologies comprise the evaluation test-bed. The goal of the wired topology is to evaluate network performance in an optimal case were server-client connection is interrupted only by a switch and therefore comprises by two wired sections supporting 100Mbps connections. Contrary, the objective of the hybrid topology is to stress the TCP operation by inserting a wireless section theoretically providing 54Mbps communication. To implement this topology, data from the switch are now forwarded to a router with wireless capabilities and then to the final destination (and vice versa of course). Additionally between the wireless router and the computer with wireless interface, walls intervene in order to emulate a suboptimal wireless connection.

In all cases communicating computers are running a 2.6.14.3 based Linux modified, recompiled kernel. The kernel, depending on the experiment scenario, was configured to support each TCP version presented in section 2. There was an FTP server daemon running on the "Server" station, while a specific 800Mbyte test file used is transferred during all experiments. The selected file assures sufficient transferring period in order to expose the behavioural and performance characteristics of the TCP algorithm under evaluation.

Measurements are taken using tcpdump, tcptrace and xplot together with the kernel extracted measurements using the `printk` function. In order to comparatively evaluate download (server-to-client) and upload (client-to-server) directions, five types of transferring scenarios were considered, including one or two simultaneous data flows,

- One download/ upload
- Two parallel downloads/ uploads
- A two data flows crossed download/upload.

From a hardware point of view all components supported adequate communication rates, so as not to impose bottlenecks in the experiments. Finally it is noted that all measurements have been taken from the "Client" station.

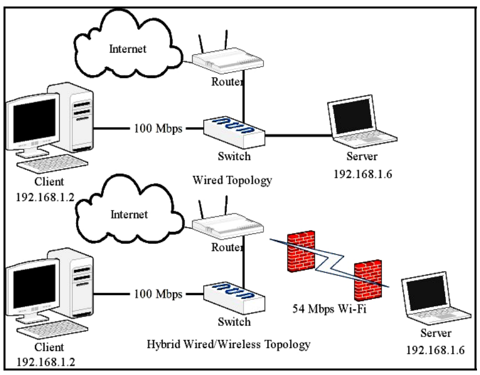

Figure 1: Test-Bed.

4 COMPARATIVE PERFORMACE EVALUATION

In order to clearly and objectively investigate the influence of each different factor under evaluation, three different parameters will be used as means of comparison. Each one of them focuses on a different area of influence such as, network topology, data transfer scenario and congestion control algorithm.

4.1 The Effect of Network Topology

As indicated by respective measurements, the differences of the two network topologies have an immediate effect on the behaviour of *cwnd* and *RTT*. However it is observed that for the same topology the TCP algorithm version it is not a critical factor. Thus indicative measurements concerning the H-TCP algorithm will be presented since analogous conclusions are drawn from all cases regarding different algorithm.

Thus, in “Fig. 3” it is depicted that in wired scenario the range of *cwnd* value’s fluctuation is quite limited indicating a steady behaviour with an average value of 18140 bytes. On the contrary for wireless scenario in “Fig. 4” a rather unstable behaviour is depicted with abrupt and frequent fluctuations reaching about 50% of the actual window value, while the mean value is significantly higher reaching about 32162 bytes. Round trip time (*RTT)* measurements also provide interesting insights. In “Fig. 2” the grey graph clearly indicate the effect of packet loss of the wireless link in hybrid topology which triggers the end of congestion avoidance phase and results the decrease of *cwnd*. As observed in the hybrid topology the experiment required almost twice the time period due to an *RTT* overhead of 20-30msec. In addition, as shown, there are short periods of time (spikes) where *RTT* in hybrid topology exceeded even 100msec severally degrading network performance.

The result of this performance degradation is clearly shown on the respective throughput graphs in “Fig. 5”. The combination of a higher average RTT of 50msec and the periods of 100+msec together with the wider range of *cwnd* in “Fig. 4” lead to a 45% higher achieved throughput for wired topology and consequently to an almost 50% delay increase for the hybrid topology.

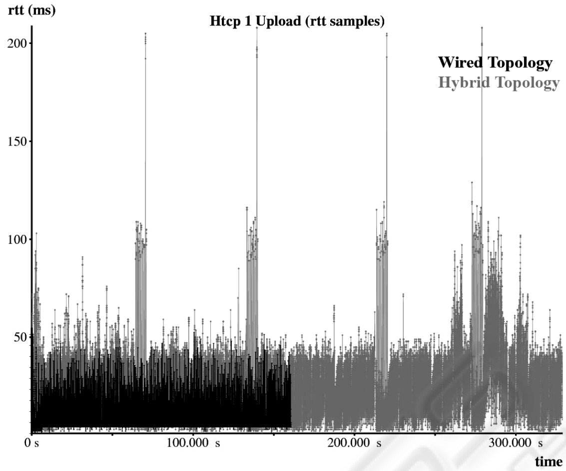

Figure 2: *RTT* of “H-TCP 1 Upload Scenario” on both topologies.

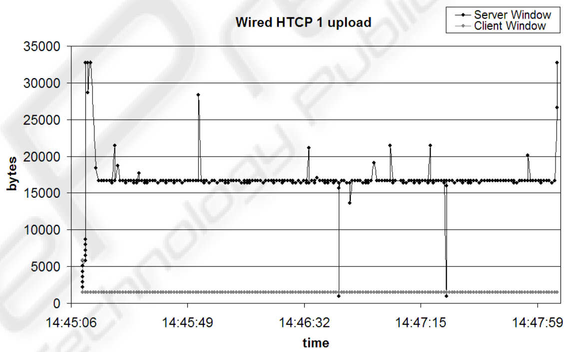

Figure 3: cwnd of “H-TCP 1 Upload” Scenario on the Wired topology.

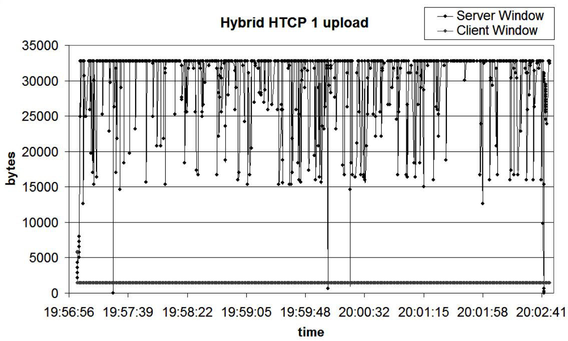

Figure 4: *cwnd* of “H-TCP 1 Upload” Scenario on the Hybrid topology.

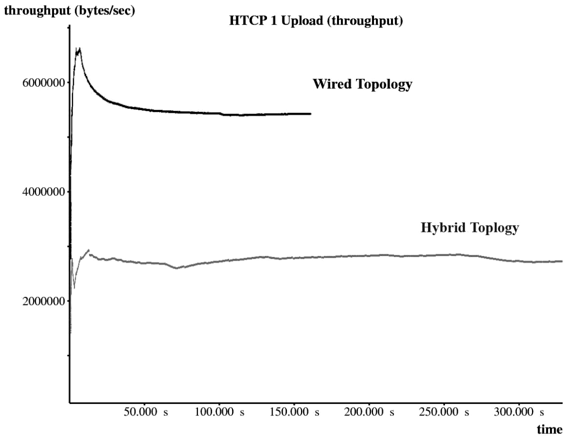

Figure 5: Throughput of "H-TCP 1 Upload" Scenario on both topologies.

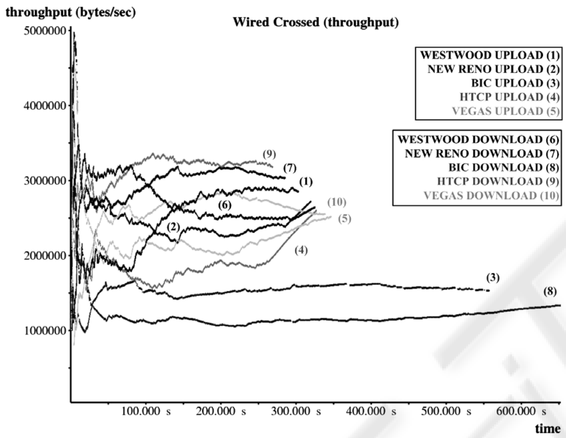

Figure 6: Throughput of "Crossed" Scenario on Wired topology.

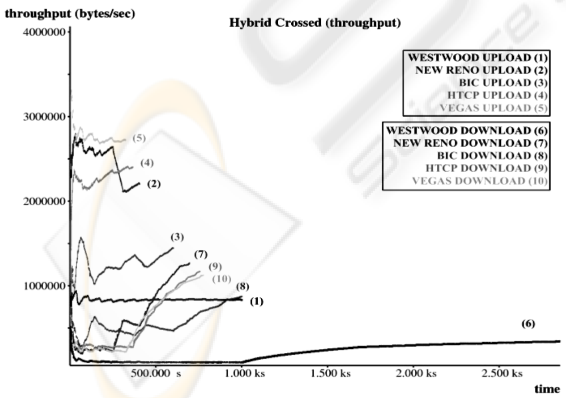

Figure 7: Throughput of "Crossed" Scenario on Hybrid topology.

4.2 The Effect of Data Flow Scenario

The number and direction of the coexisting data flows (either from the server towards the client and vice versa) in the channel clearly affect the efficiency of TCP congestion control mechanism as well. This observation is clearly depicted in Table 1 presenting the time durations required for the file transfer completion on the hybrid topology scenarios. There is a notable difference between the upload and download scenarios since the upload flows complete their transfer sooner. Furthermore the significance of the direction is also evident through the bandwidth distribution measurements since, the upload data flows capture a range of 60% up to 80% of the available bandwidth while the download flows only from 20% up to 40%.

Analysing the measurements in more detail it is observed that in the one flow scenario, upload is approximately 30% faster than download where on the crossed scenario the upload flow is 40% faster than the download flow. The same observation can be extracted from "Fig 7" concerning hybrid topological scenarios. Therefore, during an FTP file transfer where two crossed flows coexist, TCP tends to favour the client-to-server flow at the expense of the server-to-client flow leading to faster file transfers for the upload direction. Furthermore, throughput graphs of scenarios concerning two crossed flows comparative analysis between hybrid and wired topology offer additional valuable insights as shown in "Fig. 6" and "Fig. 7". Focusing on the wired topology the two crossed flows converge rapidly to a fair distribution of the channel's bandwidth. On the other hand such a convergence was not observed in the hybrid topology; significantly degrading download direction performance since the respective throughput graph exhibits increase tendencies only after the upload session is concluded.

In the two parallel flow scenarios (either download or upload) TCP is able to guarantee an equal 50% share of the channel's bandwidth to the two competing flows. This behaviour was confirmed for all the tested algorithms and both network topologies.

4.3 The Effect of Configured TCP Algorithm Version

Using the measured transfer time durations from Table 1 together with the throughput graphs of "Fig. 7", we will try to compare the five algorithms

Table 1: Hybrid Topology Transfer Time (minutes).

Algorithm	1 Download	1 Upload	Crossed		2 Downloads		2 Uploads	
			Down	Up	Down#1	Down#2	Up#1	Up#2
BIC	12.15	6.0	17.0	16.0	10.0	10.0	11.0	10.30
New Reno	7.0	6.40	12.0	7.0	11.0	10.0	10.0	10.0
Vegas	10.0	5.10	13.10	5.20	13.30	14.10	10.0	9.30
Westwood+	11.30	12.10	47.0	16.0	67.30	68.0	15.10	14.50
H-TCP	8.30	5.45	13.0	6.10	10.0	9.30	10.10	10.30

focusing on the bandwidth percentage they manage to exploit.

Therefore, it is evident that, regarding the data transferring scenario, in the crossed data flow scenario Westwood+ algorithm exhibited the poorest bandwidth utilization as both flows captured only approximately 1Mbyte/sec and completed the transfer on 47 min and 16 min for download and upload respectively. On the other hand Vegas and Reno delivered a considerably higher 3Mbyte/sec aggregated throughout, followed by HTCP 2,5Mbyte/sec and BIC recording 1,2Mbyte/sec.

Valuable information can be extracted observing that after the completion of each upload, on the crossed scenario, the corresponding download flow probes the channel in order to occupy higher percentage of the available bandwidth. However, Westwood+ was unable to take advantage of the absence of the competing flow as it managed to increase its throughput by only 200Kbyte/sec in the remaining 31 min whereas H-TCP and Vegas improved their throughput by 800Kbyte/sec in 7min and 8min till session termination respectively, Reno by 600Kbyte/sec in 5min and BIC by 200Kbyte/sec in 1min.

The clear performance degradation of the network when using Westwood+ algorithm, as opposed to the rest of the considered TCP algorithms when operating on the hybrid topology, is represented by high required transfer periods and low measured throughput for all the transport scenarios. This fact raised some concerns about the effectiveness of its bandwidth estimation mechanism. Especially on the crossed scenario where data and ACKs flow in the same direction, the result was very high *RTT* values leading to even lower bandwidth estimation. Therefore, based on understanding the theoretical approach of Westwood+ TCP algorithm and presented measurements, in the following section modifications in key parameters and respective performance evaluation are presented towards improving Westwood+ performance in the specific scenarios.

5 PROPOSED MODIFICATIONS AND EVALUATION

The proposed Westwood+ algorithm modifications focus on the respective bandwidth estimation mechanism. Two parameters of the Linux implementation are going to be modified each one toward a different goal. The main objectives are, tackling poor bandwidth estimation due to high measured RTT values, and slow response of the algorithm following the absence of the competing data flow. Both parameters are modified in the “tcp_westwood.c” code file of the Linux kernel.

The first modification concerns the value of the constant parameter `TCP_WESTWOOD_RTT_MIN`. The initial value of this parameter is 50msec while the value considered in experiments presented in this section is 40msec. The observation triggering this modification was the comparative analysis of the average measured value of *RTT* for Westwood+ as opposed to the rest algorithms on the hybrid topology “Fig. 7”. This parameter is used by the function `westwood_update_window` in order to decide whether the `westwood_filter` function that evaluates the bandwidth samples should be called. Our goal is to reduce the values of the measured *RTT* samples so as the algorithm to achieve better utilization of the channel’s bandwidth.

The second modification aims to affect the sensitivity of the Low pass filter (Grieco, 2004), implemented by the function `westwood_do_filter` using constant coefficients. We are going to present experiments configured to two different values for the factor $C=7$ that is used in the function. The objective of these changes is to make the algorithm more aggressive when it comes to probing for available bandwidth.

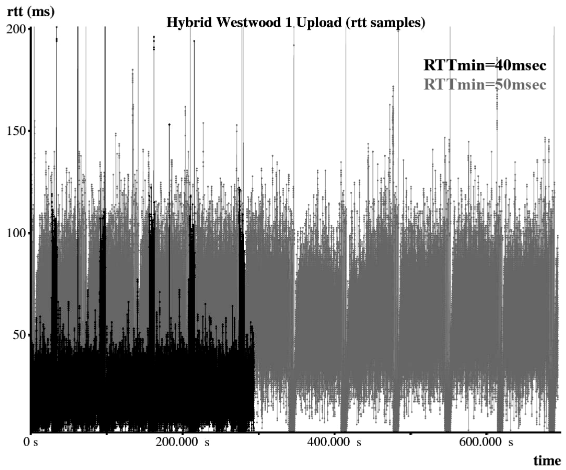

Figure 8: *RTT* of “1 Upload Scenario” on hybrid topology, *RTTmin* modified.

“Fig. 8” shows the *RTT* measurements for the two values of the `TCP_WESTWOOD_RTT_MIN` parameter. The grey graph represents the initial 50msec value and the black graph the new 40msec value. As it is depicted, there has been a significant reduce of 50% in the average *RTT* value from 100msec to 50msec, which caused a 250% raise on the used throughput from 1,2Mbyte/sec to 3Mbyte/sec compared to the original Westwood+ algorithm.

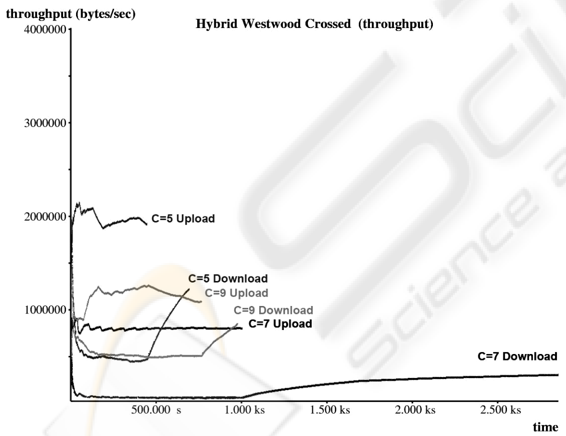

Figure 9: Throughput of “Crossed Scenario” on hybrid topology, factor *C* modified.

In “Fig. 9” the crossed scenario throughput graphs for three different values of the constant factor C are presented. For *C=9* the modified Westwood+ algorithm managed to increase its throughput by 400Kbyte/sec in 3,30min while for *C=5* by 700Kbyte/sec in 4,10min. Due to the linear pattern of the graphs it is possible to quantitatively estimate the algorithm’s aggressiveness increase effect through the respective measurements’ time duration decrease down to 1 min. Thus, compared to the 12.5Kbyte/sec/min of the initial *C=7* (estimating that the download flow reached its maximum value in 16^{th} minute of the experiment) respective results were increased 9 times for *C=9* and 13 times for *C=5*. Thus the modified algorithm appears significantly more aggressive in grabbing the available bandwidth after the absence of the opposite data flow and indicating that the default value leads to a rather passive behaviour.

6 CONCLUSIONS

This paper presents a comparative performance evaluation of five different congestion control algorithms using a local area network test-bed setup comprising by both wired and wireless segments. The evaluated algorithms have been tested in a series of experimental, data transferring scenarios taking considering both wired and hybrid wired/wireless topologies.

Additionally experiments carried out indicated possible directions towards improving the performance of specific TCP algorithms. In this context it is observed that in hybrid topologies upload data flows tend to be more aggressive in capturing the available bandwidth. In the two parallel flow scenarios TCP manages to evenly distribute the channel’s available bandwidth to the two competing flows, but is unable to do so in the two crossed flow scenarios. On the wired topology, the two crossed flows converge rapidly to a fair distribution of the channel’s bandwidth but such a convergence was not observed in the hybrid topology.

Additionally, in respective experiments Westwood+ algorithm recorded the poorest utilization of the available bandwidth in all data transfer scenarios assuming wired-wireless topology. Based on this observation, two modifications are advocated concerning key parameters of the implementation code in order to improve the performance of the bandwidth estimation mechanism. Respective measurements using the modified TCP implementation have shown significant improvement in the algorithm’s sensitivity as well as bandwidth utilization.

REFERENCES

P. Mohaparta, S. Krishnamurthy, "AD HOC NETWORKS Technologies and Protocols", 2005 *Springer Science* + Business Media, Inc. eBook ISBN 0-387-22690-7.

Lisong Xu, Khaled Harfoush, and Injong Rhee, “Binary Increase Congestion Control for Fast, Long Distance Networks” In: *IEEE Infocom* 2004.

S. Mascolo, et. al “TCP Westwood: end-to-end bandwidth estimation for efficient transport over wired and wireless networks” In: *Proceedings of the ACM Mobicom 2001*, Rome, Italy, July 2001.

Luigi A. Grieco and Saverio Mascolo, “Performance Evaluation and Comparison of Westwood+, New Reno, and Vegas TCP Congestion Control”, ACM SIGCOMM Computer Communication Review, 2004.

V. Jacobson, “Congestion avoidance and control” In: *Proceedings of the ACM Sigcomm’88*, Stanford, CA, August 1988, pp. 314–329.

Floyd, S., Henderson, T. “New Reno Modification to TCP's Fast Recovery”, RFC 2582, April 1999.

Brakmo, L. S., O’Malley, S.W., and Peterson, L. “TCP Vegas: End-to end congestion avoidance on a global Internet”. *IEEE Journal on Selected Areas in Communications,* 1995.

D. Leith, R. Shorten, “H-TCP: TCP for high-speed and long-distance networks”, Hamilton Institute, NUI Maynooth, 2004.