Deployment of TCP University of Bridgeport (UB) to Control Law Enforcement Department over Wireless Mesh Network

Wafa Elmannai · Abdul Razaque · Khaled Elleithy

Received: 15 May, 2012/ Accepted: 02 November, 2012

Abstract Transmission Control Protocol (TCP) is the most reliable protocol. That was previously introduced for wired networks only. Lately the TCP started to act poorly for several factors due to the advent of wireless network. The represented factors which need to be addressed are recovery mechanism, backup mechanism, mobility, congestion window and maximum packet size. Based on existing variants, new were introduced. The most two debated variants are Vegas and Westwood which motivated many researches. However, the number of studies proved that Vegas could perform more efficiently than other variants during the congested network whereas Westwood could consume less bandwidth. Hence, in our previous publications, we already proved by integrating some features of Vegas and Westwood to get better and more stable variant even in high congested network. It is called TCP University of Bridgeport (TCP-UB).

Based on a previous simulation, TCP-UB proved a high efficiency, less bandwidth consumption and stability from static and mobility point of view. Therefore, in this paper we are using NS2 simulator to perform TCP-UB over wireless mesh network supported with manhattan mobility model. It provides robust and faster transmission service among several police stations in Connecticut, USA.

Furthermore, we are comparing TCP-UB with Vegas and Westwood from a static and mobility point of view based on bandwidth consumption and congestion window scenario. On the basis of findings, we validate that TCP-UB is also better performer even over wireless mesh network.

Keywords: TCP UB · wireless mesh network · stability · manhattan mobility model · Bandwidth consumption · congestion window. Abdul Razaque

1. Introduction

In order to offer seamless mobility and improve coverage over the wireless local network (WLAN), WLAN had to be extended by introducing the wireless mesh network (WMN). WMN was the promised supplement for current WLAN’s broadband entree infrastructures.

The important improvement in wireless mesh network was in the communication service by providing the radio links between all the nodes. Furthermore, for urbanite zone mobile, citywide observation systems and broadband access networks, WMS’s applications provide the backhaul [1].

However, TCP was originally presented for traditional wired network [13, 14]. The main point of TCP was to provide the reliability for both congestion control and data transfer services to support the upper application layer. But based on past researches, TCP could not provide the high quality of transport service in mobility point of view. Therefore, several numbers of variants were introduced to enhance the performance of TCP [2].

Karels and Jacobson introduced first variant which called TCP Tahoe in 1986s [3]. Later on, other variants started to be employed such as: Vegas, Westwood, Reno, and New Reno, Sack … etc.

We have implemented TCP-UB in our previous research in 2012 which was based on integrating the features of Vegas and Westwood in order to provide better performance over the wireless and MANET areas. However, TCP Vegas was implemented in 1994s by Brakmo et al. The new mechanism for congestion avoidance which was provided by Vegas makes Vegas the most powerful variant during high congested networks. Vegas used this mechanism in efficient and different way from what other variant do.

If we compare Vegas and Reno, Vegas could reduce the packet loss and enhance the performance of the throughput by detecting the congestion before it occurs. When, Reno determines the congestion only by indicating the packet loss in the network. So, there is no way to know if there is congestion or not before packet loss happens. In the starting of the communication, Vegas usually compute the available...
bandwidth on the base of the difference between expected and actual throughput to escape packet loss [4].

Other important variant which makes a difference in bandwidth consumption matter is TCP Westwood especially in loss links of wireless networks [2].

Westwood applies a high performance in sender side with its bandwidth estimation mechanism. Wireless’ bottleneck sharing the connection was defined in Westwood to synchronize the congestion window size (CWND) and slow start threshold (ssthresh). When the congestion occurs in the network, Westwood starts to reset the CWND and ssthresh basis on the measured acknowledgments (ACK) [5,6&11]. Both sender and receiver have an updated information table for all the received ACK. Based on that, the fairness of bandwidth estimation is met in Westwood.

Whenever, the duplicated ACK is arrived then a new estimation would start as we explained above. However, in this paper we are examining TCP-UB over the mesh network in order to provide the reliability, more security as well as faster communication between the police departments in Connecticut State, USA. The residual of this paper is prepared as: the related work will be discussed in section 2, the projected work in section 3, the experiment setup in section 4, the experiment results 5 and the conclusion in section 6.

2. Related Work

In our previous research, we have introduced and implemented a new algorithm called TCP-UB using the NS2 simulator. The implementation was based on merging the features of TCP Vegas and TCP Westwood regarding their high efficient mechanisms. We compared and tested TCP-UB with Vegas and Westwood over the Hybrid network basis on using realistic system. According to the experiment results, TCP-UB could provide a higher performance, less bandwidth consumption, and more stability as compared to Vegas and Westwood in a high congested network from the mobility and static point of view [6].

Based on the badly performance of TCP in Wireless multi-hop broadcast areas, TCP was examined over the mesh network based on two important factors of channels interference which are hidden and exposed terminals [1]. The multichannel solution of this matter was introduced in this paper.

The solution could eradicate the hidden terminal issue and fulfill the altitudinal channel reprocess property by creating a wireless mesh network. The experiment results could show a high efficiency of the suggested assignment for developing the performance of TCP over wireless multi-hop environments.

In order to provide a combined improvement to multipath Packet Reordering, broadcast error, multi-hop influences, and congestion; those are the greatest challenges for TCP over WMSs. Therefore, the Congestion Coherence enhancement protocol was introduced in [7]. This proposed protocol could distinct from the native operations of multipath routing and wireless retransmission, the congestion control and end to end services. The paper result shows the improvement of the proposed protocol in WMNs over other developed wireless TCP variants. Hence, the projected work eliminated needless congestion window which drops and decrease forged retransmissions.

Paper [8] focuses on enhancing the performance of TCP over the WMNs using link-aware reliable transport protocol which can be considered as state full transport protocol in order to save the state between the middle nodes. On the other hand, adaptive and receptive transport protocol was proposed over the WMNs in [9]. The proposed protocol made the fairness of resources allocation over many surveys. That could improve the performance of TCP over the WMNs.

Based on the impact of TCP performance over the WMNs, the coding of the network was examined in [10], which permits the node either to chain compound packets into one or several. The results proved the network coding could help the TCP over WMNs by reducing packet loss rate and the wireless channel’s contentions.

3. Proposed Work

A. Description of Police Departments System over Wireless Mesh network using TCP-UB:

![Fig.1: Communication among different police stations through wireless Mesh topology using TCP-UB](image)

The benefit of mesh topology for being a fully connected network, we have designed our mesh network which contains a large number of police stations over a number of cities in Connecticut State, USA. Each station would be considered as mesh node using TCP-UB protocol in order to provide faster and reliable communication service as well as more security. All the nodes are connected to each other as shown in figure1.
Our aim in this paper is to provide a faster data transfer using routing technique over the police department system. Each station has to perform in two stages; one to capture its own data, when no other station is allowed to get this information or it should relay to other nodes.

Since we are using a routing technique, the packet will be sent through a path from node to node until it reaches its destination. The router will be responsible to choose an appropriate path for propagating the packet.

With use of self-healing algorithms which allows the router to determine if there is any broken or congested path, we are using zone Routing Protocol.

ZRP works as zone around each node which really can be helpful for mesh topology in order to decrease the overhead of the control of practical routing protocols. Not only can it be helpful for mesh topology but ZRP also reduces the routing latency which is caused by interactive routing protocols.

In case the router determines a broken path, the router can assign other path since we are using mesh network and all the stations are connected to each other. So, each node will have more than one path to be accessed over.

TCP-UB proves a high efficiency and stable connection over this network since no disconnection would occur. Furthermore, there is no loss of data can be expected with TCP-UB since its communication control mechanism is used between the nodes.

However, according to the above information the delivering percentage can get higher than other protocols which used with WMNs.

Also, less packet loss leads to increasing the efficiency of the throughput. In summary, this network can benefit the police department applications which are used between their stations and need to send and receive data with providing higher security and more privacy.

Since the TCP-UB is controlling the communication by sending the packet over a committed line, then we can guarantee all above promises. Also, with the use of mesh topology we can ensure two important things:

First, if there is broken or an effected link inside the network, then the network will continue work efficiently. Only that link will be exaggerated. Second, the configuration from point to point can identify the errors and conforms the node’s identification.

B. TCP-UB algorithm Description:

Actually, TCP-UB has been designed based on three congestion phase partitions (Alpha (α), Beta (β) and Gama (δ)) as shown in figure 2.

In this paper we are going to describe only the behavior of TCP-UB and on what we made our assumption to design TCP-UB. However, we are not going to describe the TCP-algorithm steps since we had done that in previous works [6 and 11].

The main idea of the three partitions of TCP-UB congestion window is to prevent our network of the congestion and change the state based on the available bandwidth in network over different times. So, always TCP-UB starts controlling the network based on initial computation on basis of CWND and the Base RTT. This process can be called an expected rate where the expected rate is data rate which is calculated based on previous network condition.

Also, we are calculating the actual rate which is the current data rate that is being sent over the network. Then the difference between both the actual and the expected rate need to be determined

\[ \text{Difference} = \frac{(\text{Expected Rate} - \text{Actual Rate})}{\text{Base RTT}} \]

BaseRTT is the minimum RTT that calculated by original TCP for each congestion window. However, the CWND has to be updated within each party (α, β, δ) based on this difference.

We consider (α) as the minimum threshold because in that time we don’t expect any congestion over the network. In that particular time we compare the calculated difference with alpha, if it is less than alpha then we increase our CWND by one and we start sending the packets. But if not then we need to compare the difference again with the middle threshold (gama), if they are equal or not.

In other work did we reach this middle threshold of our network or not; this comparison can be used to estimate the congestion possibility over network. If they are equal, then
we make both slow start threshold (ssthresh) value and bandwidth estimation value be equal which is the sharing of the bottleneck that used by the linking of the network, multiplying the segment size. The point of using ssthresh is to check whether TCP-UB is in congestion phase or slow start phase.

\[ ssthresh = \frac{(\text{BWE} \times \text{BaseRTT})}{\text{seg_size}} \]

Then we need to check CWND if it get greater than ssthresh or not. If yes then we congestion can occur, reset the CWND and make it equal to ssthresh. If the previous condition does not meet then there is a big possibility of congestion. So, we need to check the time out before we can check if there is available bandwidth or not.

As known the time out is the limited time of transmitting the packet and receiving the acknowledgment within this giving time. If the time is expired then we need to use retransmission mechanism and restart the slow start phase by setting the CWND to value 1. After re-estimating the bandwidth, we reset also the ssthresh.

\[ ssthresh = \frac{(\text{BWE} \times \text{BaseRTT})}{\text{seg_size}} \]

After restarting the transmission process again we need to check ssthresh weather is less the value 2 or not; if so then reset it to 2.

Last comparison need be done for our protocol is to compare the difference and maximum threshold (Beta). This step is very important to know whether our network can be in risk point or not. The value should be less than beta always to avoid the congestion and losing the data in that particular time.

If not then decrees CWND by one value and recheck the bandwidth. If we have enough bandwidth we can continue sending, if not we need to reduce the number of packet based on the bandwidth availability. Using these previous steps we can avoid the congestion to occur in the used network.

Final, if all above comparisons did not meet, then CWND need to be fixed. By that we can make sure that the CWND is under our protocol’s control.

C. Bandwidth Estimation Algorithm in TCP-UB:

TCP-UB uses its own point of view to estimate the bandwidth over the network. This estimation is used to calculate the bandwidth availability over the whole network for sending all the data and receiving the ACKs.

Our aim is to measure the bandwidth based on transmitted packet. We have used Algorithm1 in earlier work which was explained and stated in both [6 and 11].

Then determine the bandwidth’s value. We used algorithm1 to measure the transmitted packets and how many ACKs we have.

Basis on this calculation we can estimate the bandwidth in next algorithm we named algorithm2. We checked if the three duplicated ACK are received then we set the ssthresh = Bandwidth Estimation * RTT_min.

Also, TCP-UB reset the CWND and makes it equal to ssthresh if they are not equal.

In the above explanation, the estimation was in case the time out is not expired but the congestion occurs.

However, in case the time out is expired even there is no congestion then we need to reset the ssthresh and make the CWND equal to one.

4. Mobility Model Over View and Simulation Set Up

In this section we demonstrate the mobility model that we are using which is Manhattan Mobility Model. It is used to help TCP-UB over wireless mesh network to control the nodes’ mobility.

A. Overview of Manhattan Mobility Model:

This model can be described as geographic model when all the nodes are distributed randomly. Each node decides one path each time. All the nodes have the same probability of choosing which path that node needs to be moved to from its initial location.

After the nodes reach their destinations, the subsequence paths will be decided probabilistically. Whether the node can be able to move in same path or it should move to other.

Based on the probabilistic rules, the node has half the probability of moving in same path and other half divided between other two choices (south/north) and (west/east).

However, if there are only two options available to the node which is one of the bounded paths of the network, then the percentage will be half and half. But if the node has only one choice which is one of network’s corners, then the percentage will be 100% for that path [12].

B. Simulation Set Up:

To simulate our network, we have used NS2.28 on Linux Red Hat-9. In order to control the mobility of the nodes, we have used Manhattan Mobility Model.

We use Object-Oriented extension of TCL language (OTCL) to implement TCP-UB as shown in figure3.

OTCL is programming language which is supported by NS2 simulator.
We also have implemented Westwood and Vegas to be compared with TCP-UB. The transmission and propagation range is 250 meter. We set 100 nodes over mesh network. The packet length is 1040 bytes within square of 1000*1000 meter. This is the limitation of node for transmission range [1]. The speed of mobile node is set as (0m/sec for minimum speed and 35m/sec for maximum speed). Each node is allowed to send 8 packets/sec. The pause time is 5 seconds per 50 seconds. So, each 50sec the process is stopped for 5 sec. We set 140 seconds for the simulation time. The queue size is 80 packets. Finally, we have used Omni directional as antenna type.

5. Simulation Results and Analysis

In this section we are discussing our results based on the comparison that we have made between (TCP-UB, Vegas and Westwood) variants over the wireless mesh network. We have compared those variants from the static and mobility point of view based on two scenarios which are bandwidth consumption and congestion window.

A. The Congestion Window Scenario:

This section is based on the congestion window algorithm for TCP-UB, Westwood and Vegas. We have examined the congestion algorithm for the three variants over the mesh network. Figure 4 show the performance of the three variants of static point of view for congestion window algorithm. However, we have studied several researches on TCP’s congestion control algorithm; we concluded the congestion window algorithm can be deployed but the work that had done is still not sufficient.

Our aim of this experiment is to exam our new protocol with recognized protocols over the wireless mesh network and show its performance in case if congestion occurs. Our investigate gives proof about suitability of proposed and existing variants. In both figures 4 and 5 for static and mobility points of view, we could control the congestion in scenarios of busy traffic. As seen in our results which determine there is no congestion control algorithm can be stable all the time over the congested network. In same time TCP-UB show better performance in both cases even when the nodes are moving.

We can clearly see during the simulation time, the congestion window shrinks in the case of TCP Westwood and Vegas whereas congestion window of TCP UB shrinks little if we compared with other two. In this case we can pass more traffic over TCP UB than other two variants.

![Fig. 3: TCP-UB implementation over mesh network](image1)

![Fig. 4: Congestion Window Scenario for Static Point of View](image2)

![Fig. 5: Congestion Window Scenario for mobility Point of View](image3)
The same thing is shown in figure 5, determining the mobility scenario when the nodes are moving. TCP-UB performs better and tries to be stable in same range from 145 to 119 when Vegas and westwood are shrinking to 63.

The reason of Vegas and Westwood’s badly performance is due to their bandwidth and mobility issues. Vegas perform as worst one over both scenarios due to its lower sharing bandwidth utilization issue. Vegas always have issue with sharing the bandwidth with other variants. Based on that, TCP-Vegas get lower percentage of bandwidth. So, based on our result we can claim that TCP-UB can offer better transmission service than other compared variants over the congested network.

B. Bandwidth Consumption Scenario:

In this section we have examined TCP-UB, Vegas and Westwood based on bandwidth consumption scenario from static and mobility point of view.

Each variant has its own bandwidth estimation algorithm. Vegas’s bandwidth estimation algorithm proved a high performance and consumes less bandwidth. That only if Vegas apply separately over the network. Vegas’s algorithm is based on controlling the CWND; that is basis on the difference between actual and expected data rate. However, in case of connection sharing Vegas start to perform aggressively and consume higher bandwidth as shown in figure 6.

Hence, Westwood provides a good bandwidth estimation algorithm based on its recovery mechanism. Its algorithm applies based on the measured acknowledgments. Every time the congestion occurs, Westwood estimate the bandwidth and reset both CWND and ssthresh based on the result. Then it starts recovering the packets from last ACK even if this packet is received by the receiver but the ACK is lost. That makes the westwood consume more bandwidth than it should do.

Basis on figure 6 in static scenario TCP-UB consumes 0.5Mbps as average when Vegas consume around 0.7Mbps and Westwood consumes around 0.8 Mbps.

Figure 7 represents the mobility scenario for bandwidth consuming algorithm of TCP-UB, Vegas and Westwood. Also, here TCP-UB shows better performance from bandwidth consuming view. TCP-UB consumes only less than 0.7Mbps. When Vegas consume higher than 0.7 Mbps and Westwood ingests around 0.8 Mbps.

The reason of good performance of TCP-UB is its own algorithm which was built based on integrating the features of Westwood and Vegas. Also, division of the three portions over the congestion phase could control the congestion of the network for TCP-UB. So, if the congestion happened, then we can control the CWND by decrease it by one and reduce the number of the packets that needs to be sent based on bandwidth availability. That can help TCP-UB in consuming less bandwidth and avoid the congestion in same time. The other feature that TCP-UB provides is to retransmit only the lost packet based on the information table that assigned between the sender and the receiver.

6. Conclusion

In this paper we have examined TCP-UB over the wireless mesh network and compare it to Vegas and Westwood variant. We have simulated two scenarios which are congestion window and bandwidth estimation scenario for our comparison based on both static and mobility point of view.

Our results show that TCP-UB provides better performance in term of mobility effect and congestion window by allowing more traffic than Vegas and westwood. Also, TCP-UB consumes less bandwidth than the compared variants. All proofs show that TCP-UB is highly promising variant over wireless mesh network.
In summary, we have deployed TCP-UB for police stations connections but it can be used in hospital departments for faster and reliable communication especially for Ambulance helicopter. Finally, based on these results we can conclude this paper by saying: there are more challenges that still need to be addressed regarding to TCP over wireless mesh network.

References

1. Li-Ping Tung and Wei-Kuan Shih, Te-Chung Cho and Yeali S. Sun, Meng Chang Chen, “TCP Throughput Enhancement over Wireless Mesh Networks” IEEE, 2007.


Authors:

Mrs. Wafa Elmannai is a PhD student in the School of Engineering and Computer Science at the University of Bridgeport. She received her master degree from University Of Bridgeport, USA.

She has finished her Bachelor’s degree in Fall2005 at Ben Ashore College for Computer Science. She graduated with highest honors, ranking 4th in her Bachelor degree and awarded merit certificate. She also was recognized by both Phi Kappa Phi and Upsilon Pi Epsilon. She received a scholarship in 2012 based on her academic achievement from Upsilon Pi Epsilon.

Mrs. Elmannai worked as a Research Assistant at University of Bridgeport since 2010-2011. She worked in management department as assistant administrator in education department since 2005-2009. She worked also as teacher in Algamaheria School since 2004-2005.

Mrs. Elmannai is interested in network area, mobile communication, and some software applications. She has published several papers in international conferences. She mad presentation of her research in conferences.

Mr. Abdul Razaque is PhD student of computer science and Engineering department in University of Bridgeport. His current research interests include the design and development of learning environment to support the learning heterogamous domain, collaborative discovery learning and the development of mobile applications to support mobile collaborative learning (MCL), the congestion mechanism of transmission of control protocol including various existing variants, delivery of multimedia applications. He has published over 30 research contributions in refereed conferences, international journals and books. He has also presented his work more than 10 countries.

During the last two years he has been working as a program committee member in IEEE, IET, ICCAIE, ICOS, ISIEA and Mosharka International conference. Abdul Razaque is member of the IEEE, ACM and Springer Abdul Razaque served as Assistant Professor at federal Directorate of Education, Islamabad. He completed his Bachelor and
Master degree in computer science from university of Sindh in 2002.

He obtained another Master degree with specialization of multimedia and communication (MC) from Mohammed Ali Jinnah University, Pakistan in 2008. Abdul Razaque has been directly involved in design and development of mobile applications to support learning environments to meet pedagogical needs of schools, colleges, universities and various organizations.

Dr. Elleithy is the Associate Dean for Graduate Studies in the School of Engineering at the University of Bridgeport. He has research interests are in the areas of network security, mobile communications, and formal approaches for design and verification. He has published more than one hundred fifty research papers in international journals and conferences in his areas of expertise. Dr. Elleithy is the co-chair of the International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CISSE).

CISSE is the first Engineering/Computing and Systems Research E-Conference in the world to be completely conducted online in real-time via the internet and was successfully running for four years. Dr. Elleithy is the editor or co-editor of 10 books published by Springer for advances on Innovations and Advanced Techniques in Systems, Computing Sciences and Software.