

# AN EXPLICIT LOSS AND HANDOFF NOTIFICATION SCHEME IN TCP FOR CELLULAR MOBILE SYSTEM

Nayeema Islam and Mohammed Nasir Uddin

Department of Computer Science and Engineering, Jagannath University, Dhaka, Bangladesh

## **ABSTRACT**

*With the proliferation of mobile and wireless computing devices, the demand for continuous network connectivity exists for various wired-and-wireless integrated networks. Since Transmission Control Protocol (TCP) is the standard network protocol for communication on the Internet, any wireless network with Internet service need to be compatible with TCP. TCP is tuned to perform well in traditional wired networks, where packet losses occur mostly because of congestion. However cellular wireless network suffers from significant losses due to high bit errors and mobile handoff. TCP responds to all losses by invoking congestion control and avoidance algorithms, resulting in degraded end-to-end performance. This paper presents an improved Explicit Loss Notification algorithm to distinguish between packet loss due to congestion and packet loss due to wireless errors and handoffs. Simulation results show that the proposed protocol significantly improves the performance of TCP over cellular wireless network in terms of throughput and congestion window dynamics.*

## **KEYWORDS**

*TCP, Cellular Wireless Network, Handoff, High Bit Error, Explicit Loss and Handoff Notification*

## **1. INTRODUCTION**

TCP is the most commonly used protocol at the transport layer of the network stack in the Internet. It provides reliable data transfer for many popular Internet applications such as electronic mail, file transfer, web browsing and remote network access. In this context, any wireless network with Internet service needs to be compatible with the protocol used in the wired network, i.e. the TCP/IP protocol.

Regular TCP was not designed with mobile hosts in mind. Thus we cannot expect regular TCP to perform well in a mobile network. One reason for this is that wired links have low bit error rates (BER), as opposed to wireless links that suffer from high bit error rates and mobile handoff [1]. Reference [2] indicates that BER of wireless can reach  $10^{-5}$ . When a packet is lost, regular TCP assumes that it is due to congestion. If regular TCP is used on a mobile network, it will encounter packet losses that may be unrelated to congestion. Nonetheless, these losses will trigger congestion control procedure at the fixed host. These procedures will result in significant reductions in throughput and unacceptable inter-reactive delays for active connection, thus severely degrading performance.

Generally, hand-offs are expected to introduce three major impacts on the transport protocol [3]. First, the packets will experience a much higher delay during a hand-off due to the turn-over

delay and packet rerouting. Especially in the future mobile IP network environment the handoff delay is supposed to become significant. The second impact is multiple packet droppings. Packets still travelling on the previous path between server and host or the ones which are stored at the former wireless access point might either be dropped or re-routed. Since intersystem hand-offs are not expected to support packet forwarding, the dropping of multiple sequenced packets is most likely. Finally, if packet forwarding is supported, the transport protocol has to deal with massive packet re-ordering. Packets travelling on the new path will be mixed with forwarded packet coming from the wireless access point.

TCP performance has been extensively researched on wired networks that have high bandwidth and throughput, and low latency and delays [4]. As expected, TCP performs very well on wired networks. However if standard TCP is used without any modification in the mobile networks a serious drop in throughput of the connection is experienced [5]. This is because TCP has been tuned for traditional wired networks. It assumes congestion in the network to be the primary cause for packet losses and unusual delays [6]. TCP performs well over such networks by adapting to end-to-end delays and congestion losses. On the other hand, in cellular mobile environment losses are more often caused by high Bit Error Rate (BER), handoff and frequent disconnections [5]. If the losses are not due to congestion, then TCP unnecessarily reduces throughput leading to poor performance.

## **2. RELATED WORKS**

The degradation performance of TCP over wireless links is mostly due to mistaking wireless losses for congestion. Many approaches have been proposed to improve TCP throughput in the wireless cellular network.

Link layer schemes [7] [8] are the early approaches to improve the poor performance of TCP over wireless links. These schemes attempt to hide the unreliable wireless links from the wired hosts by using local retransmission or forward error correction (FEC) [7] over wireless links. Since the transport layer has its own retransmission scheme, the interaction between data link and transport layer would be complicated. Also link layer solutions cannot account for disconnections.

The fast retransmission approach [9] addresses the issue of TCP performance when communications resumes after a handoff. The main drawback of this approach is that it deals only with handoffs and not with the error characteristics of the wireless link. Also, the receiver shrinks its window to the minimum and applies the slow-start algorithm [6].

Indirect –TCP (I-TCP) [10] is proposed to improve the TCP performance that involves splitting each TCP connections between a sender and a receiver into two separate connections. Later (Brown and Singh, 1997) proposed another similar protocol named TCP for mobile cellular networks (M-TCP) based on split-connection approach. Split schemes violate end-to-end TCP semantics, since acknowledgements may reach the sender before data packets reach their destination [11]. Also, every needs to go through the TCP protocol stack and incur the associated overhead four time, once at the sender, twice at the base station and once at the receiver [6].

Snoop TCP is [12][13] aimed to achieve the goal of improving TCP performance thereby retaining end-to-end semantics. An agent, called Snoop Agent is introduced to cache the TCP segments and retransmit the segments only on the wireless link. Snoop protocol is a good scheme to improve the performance of TCP in wireless networks. The snoop scheme does not cover the cases where the Mobile Terminal (MT) gets involved in handovers [13]. Also Snoop protocol retransmits the lost packet like other link layer solutions [14].

### 3. THE PROPOSED SYSTEM

The design goal of developing modified TCP for cellular mobile network can be outlined as follows:

- Improves the TCP performance by proper mechanism with respect to high BER, handoff and Congestion.
- Maintains end-to-end TCP semantics.
- Compatibility with standard TCP.

#### 3.1. Acknowledgement Packet Modification

A modified form of acknowledgement packet is used in the proposed explicit loss notification (ELN) system. For this purpose TCP acknowledgement packet has been extended with two more bits named ELN flag and ELN bit. ELN flag indicates whether the ACK packet contains an error notification or not. A '1' in ELN bit of ACK indicates that the packet is lost due to the wireless error and a '0' in ELN bit (elnb) of ACK is used to indicate the Handoff completion. Though the handoff of mobile host in cellular system is not an error, since this causes considerable packet losses the proposed system informs the sender about handoff completion. So that the sender can distinguish packet losses due to handover from other and thus can initiate proper steps to reduce the loss related to handoff.

#### 3.2. Handoff Notification

To handle handover the sender stores previous congestion window (*cwnd*) size and slow start threshold (*ssthresh*) whenever they are changed. The Mobile Node also stores the ACK for the last data segment it received.

The base stations send advertisements at a regular interval to help the mobile station to detect link changes. As soon as the mobile station receives the advertisement after hand-over, it sets the ELN flag of the ACK, sets ELN bit to '0' and sends ACK to the base station. This ACK also contains the SN same as the ACK stored by mobile host. When such a handoff takes place, the routing tables in both the mobile host and Base stations get appropriately updated [14]. This routing mechanism is not being considered in this research since routing task is performed at network layer and beyond the scope of this research.

#### 3.3. Software Modification at the Base Station in Forward Direction

In this scheme an agent is introduced at the base station. The function of this agent is to cache the SN of the packets transmitted from fixed host. Like ordinary wired network, some transmitted packets may be lost due to congestion. In this scheme a small extra cache will be used to record the SN of these discarded packets. The installation of such discard recording cache needs extra cost. However, since a few miss of loss notifications can seriously degrades the performance of TCP, the use of such small cache is assumed to be cost-effective.

The following algorithm shows the routine at the base station when the packets arrive from fixed host to base station.

- When BS receives a packet

```
Check the input queue length
If BS discards the packet
    Record SN of the packet at discard-recording Cache
Else
    Record SN of the packet at Cache
```

### 3.4. Software Modification at the Mobile Host

When mobile host (MH) is a receiver and doesn't receive a packet, it sets the ELN bit of the duplicate ACK as '1' and sends it to the base station. It also stores ACK for the last data segment it received.

The following algorithm shows the routine at the MH

- When MH is a receiver

```
    If MH does not receive a packet
        Set ELN Flag of ACK
        Set ELN bit of ACK ==1 and send
    Else if MH completes handover
        Set ELN Flag of ACK
        Set ELN bit of ACK==0
        Send ACK
    Else
        Send ACK
```

### 3.5. Software Modification at the Base Station in Reverse direction

The base station checks the ACK received from mobile host. If the ELN bit of the ACK is '1', the SN of the ACK at the cache of BS is checked. If SN of the ACK exists at the cache of BS, the ACK is only passed toward the sender. If there is the ACK's SN at discard-recording cache, BS resets ELN flag.

The following algorithm shows the routine distinguishing transmission losses from packet losses.

- When BS receives a packet

```
    If ELN Flag is set
        If ELN bit==1
            If there is the SN of ACK at cache of BS
                Pass the ACK to the sender
            Else if the ACK's SN exists at discard-recording cache
                Reset ELN flag
                Pass the ACK to the sender
        Else if ELN bit==0
            Forward the ACK to the TCP sender
    Else
        Forward the ACK to the sender
```

### 3.6. Software Modification at TCP Sender

When the fixed host receives the ACK, it takes action with the information stored in the ELN bits. If the flag is set then it checks the ELN bit. If ELN bit is '1', the sender decides that the loss is caused by transmission error and only retransmits the packet. If ELN flag is not set and the sender meets timeout or three duplicate ACK's, sender does congestion control mechanism. The following routine shows the sender's action.

To handle handoff the sender stores previous congestion window (*cwnd*) size and slow start threshold (*ssthresh*) whenever they are changed. If the sender receives an ACK with 'Handover bit' (i.e. '0') it resets the retransmission timer, adjusts its send window in response to the sequence number of the ACK and starts transmission. The sender does not reduce its congestion window size since it can distinguish these hand-over losses from congestion losses if the retransmission timer is not expired.

- When sender receives an ACK

```

If ELN Flag is set
  If ELN bit== '1'
    Only retransmit the loss packet without window reduction
  Else If ELN bit== '0'
    If retransmission timer is not expired
      Reset transmission timer
      Adjust send window in response to SN of ACK
      Restart transmission without window reduction
    Else
      Restore cwnd and ssthresh to previous value stored by sender
Else (If the sender meets timeout or 3 duplicate ACK)
  Perform Congestion Control mechanism

```

## 4. SIMULATION MODEL

The simulation model (network topology) of the proposed system is shown in figure 1. As shown in figure, there are four nodes: a fixed host (FH), two base stations (BS), and a mobile host (MH). Each base station includes a finite buffer drop-tail gateway. In this research we are only concerned with bulk data transfer from a fixed host to a mobile host. Therefore a TCP source is embedded in the fixed host and the TCP sink is embedded in the mobile host.

Although the proposed scheme can be applied to various network topologies, this topology is used since similar topology has been adopted in many experiments of TCP performance over wireless mobile links, for example [16] and [17]. So that it will be easy to compare the performance of the proposed system with that of others.

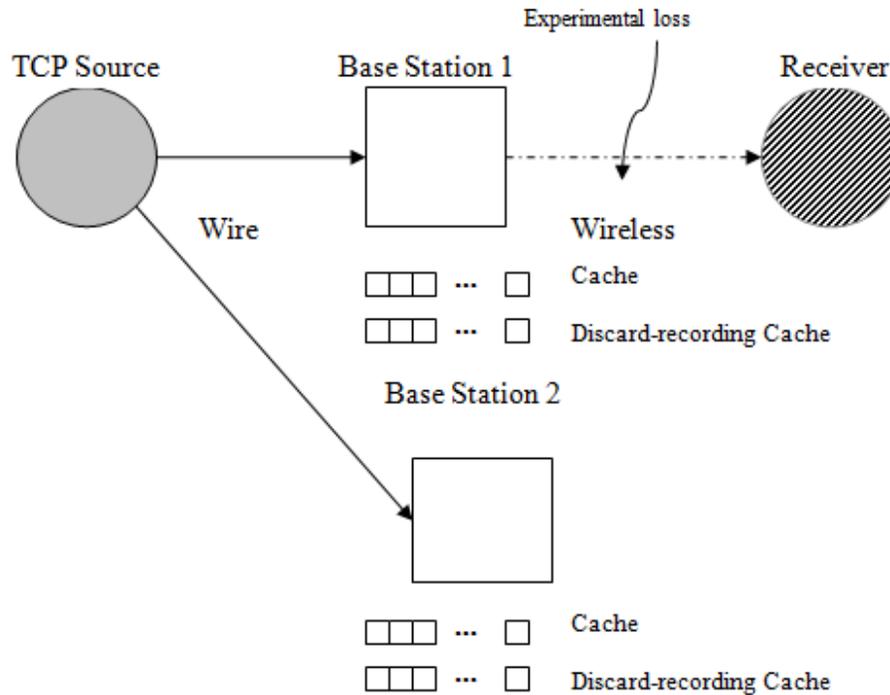


Figure 1: Network topology for simulation

The network simulator (NS2) from Lawrence Berkeley [18] is used to simulate the TCP packet transmission in wired / wireless segments of the network. Table 1 shows the system parameters that have been used in this simulation.

Table 1: Simulation Parameters

Wired link capacity	10 Mbps
Wired propagation delay	20 ms
Wireless link capacity	2 Mbps
Wireless delay	2 ms
TCP segment / packet size	1024 , NS2 default [18]
ACK size	40 byte NS2 default [18]
Wireless router buffer size	50 packets
Wireless packet loss rate	0.0001 to 0.1 per packet

## 5. SIMULATION RESULTS AND DISCUSSION

Simulation has been carried out to evaluate the performance of the proposed scheme i.e., Explicit Error and Handoff Notification Scheme under high BER condition of wireless link and during mobile handoff. This Scheme is then compared with TCP Tahoe [19], Reno [20][21], New Reno [22], Sack [23] and Snoop Protocol [12].

The performance metrics of the interest in this study is throughput: This is the measure of how soon an end user is able to receive data. It is determined as the ratio of the total data received by the end user and the connection time. A higher throughput will directly impact the user's perception of the quality of service.

### 5.1. Performance of Explicit Error and handoff Notification during High BER

Figure 2 shows the throughput performance of TCP Tahoe, Reno, New Reno, Sack, TCP Snoop and proposed Explicit Loss and handoff Notification Scheme.

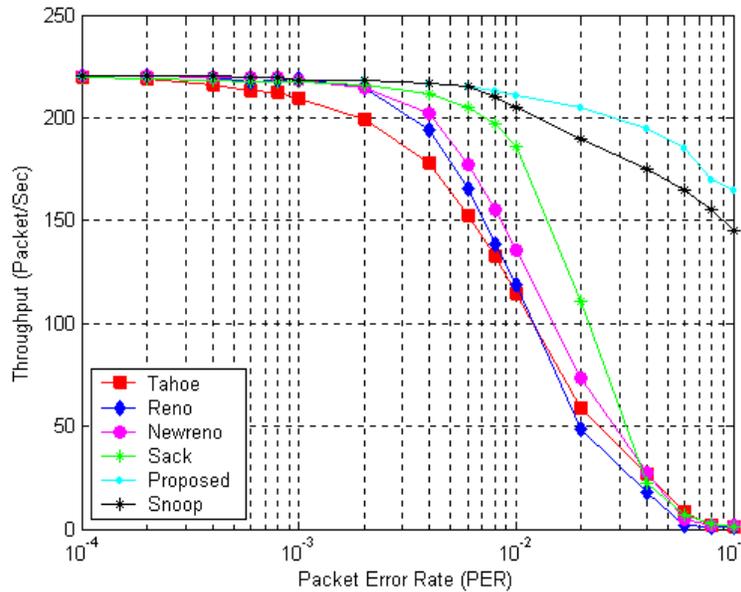


Figure 2: Throughput Performance of TCP Tahoe, Reno, New Reno, Sack, Snoop and Proposed system (Explicit loss notification)

It can be shown from simulation result that TCP Explicit Loss Notification is able to maintain high throughput over the wireless link for a range of error rates because it does not reduce its sending rate when the losses are due to wireless errors. The difference between TCP ELN and Standard TCP is illustrated in Figure 2. In case of standard TCP the lost packets or acknowledgment results in a timeout or retransmissions. In case of timeouts, the standard TCP shrink their congestion window thus resulting in low throughput. In case of TCP ELN, it distinguishes wireless losses from congestion related losses and then decide whether to shrink congestion window. The difference in performance between standard TCP and TCP ELN can be explained by the fact that the TCP invokes congestion control frequently during the experiment because of loss packet or duplicate acknowledgements and uses slow start to increase its window. Figure 3 shows the congestion window dynamics of Snoop TCP, TCP Sack and TCP ELN.

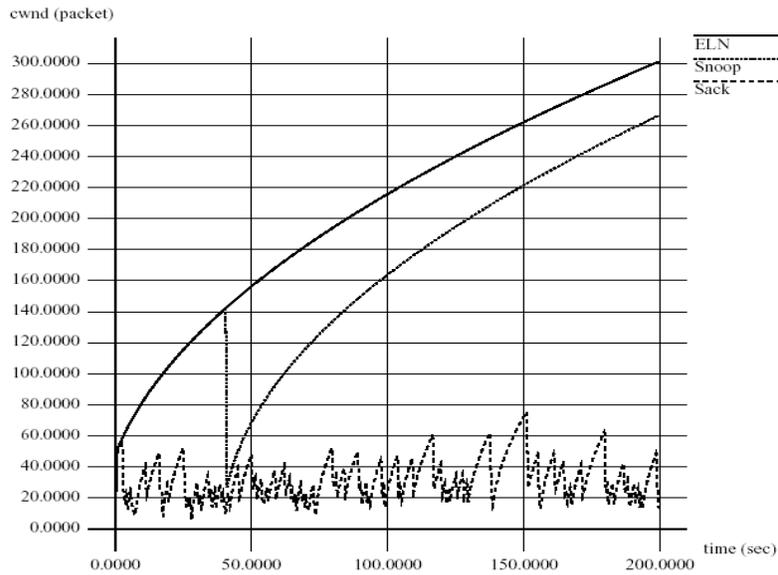


Figure 3: Comparison of Congestion window dynamics between ELN, Snoop TCP and TCP Sack over wireless network at PER 0.002 for a simulation of 200 seconds

In case of unmodified TCP, the lost packets or ACKs result in timeouts and retransmissions. In case of timeouts, the unmodified TCP shrinks its congestion window thus resulting in low throughput. Snoop TCP decreases the rate of window fluctuation significantly, through Snoop agent. However, it cannot completely avoid shrinking its congestion window. On the other hand, the congestion window of TCP ELN never really shrinks.

### 5.2. Performance during Mobile Handoff

Figure 4 compares the congestion window dynamics of modified TCP and compares it with unmodified TCP. It can be shown that when a handoff occurs unmodified TCP reduces its congestion window size, however the window size continues to increase in case of modified TCP.

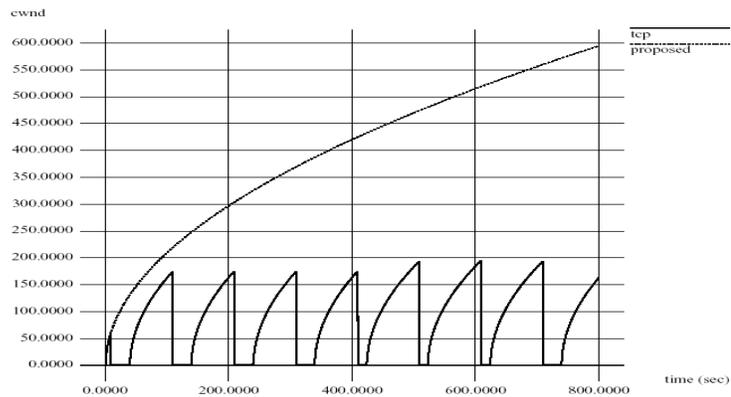


Figure 4: Congestion window dynamics of Standard TCP i.e., TCP Sack and Modified TCP for wireless network for mobile host speed 40 km/hour in presence of mobile handoff

Figure 5 plots the sequence number that mobile host receives versus time. From the figure it is shown that since ELN does not invoke congestion control procedure during timeout it is able to receive more packets for a given time.

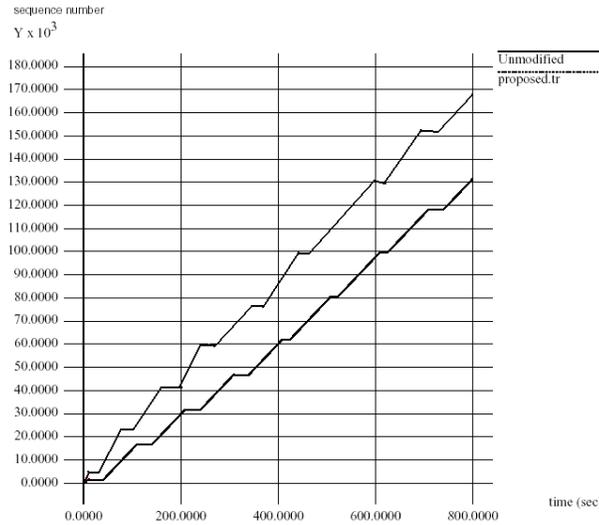


Figure 5: Sequence number vs. time graph of Standard TCP i.e., TCP Sack and Modified TCP for wireless network for mobile host speed 40 km/ hour in presence of mobile handoff

Figure 6 shows the throughput performance of TCP Explicit loss Notification compared to unmodified TCP. Because of the advantage of the proposed scheme in enabling the TCP sender to distinguish the packet losses due to handover it does not invoke unnecessary congestion control procedure during mobile host handoff. Thereby, it maintains comparatively high throughput under handoff condition. However the performance of the modified TCP decreases at higher speed.

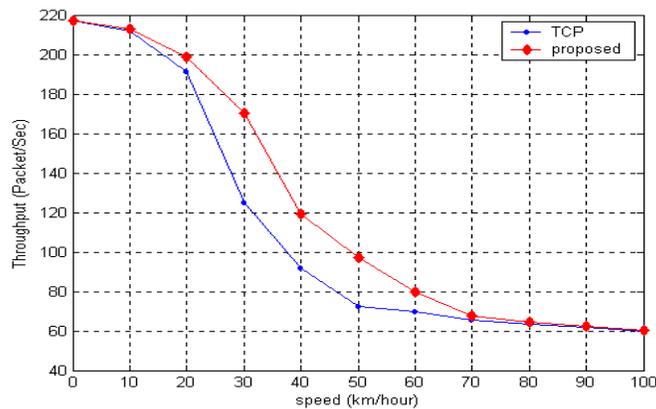


Figure 6: Speed of mobile host versus throughput graph to compare the performance of Standard TCP and Modified TCP (Explicit Loss Notification)

## 6. CONCLUSIONS

In this research an explicit loss notification scheme is presented to improve the throughput performance of TCP/IP over wireless networks. It is shown through simulation that the Explicit Error Notification scheme is able to maintain high throughput over the wireless link for a range of error rates because it does not reduce its sending rate when wireless losses or handoffs are detected. According to the simulation results the algorithm achieves significant throughput improvement.

This scheme maintains the end-to-end semantic of TCP and correctly informs the sender the reason of packet loss. In this scheme extra cache is used at Base station to store sequence number of packets. This cache at BS needs extra cost. However, because a few of miss loss notification can seriously degrade performance, the use of a small extra cache is cost-effective. This protocol improves the performance of TCP in case of high BER as well as in case of handoffs and does not interfere with TCP's congestion control behaviour.

This mechanism successfully differentiates congestion loss from corruption loss (high BER). However, this scheme relies on intermediate nodes to provide sufficient information for distinguishing corruption loss from congestion loss. Such mechanism is not very useful when the IP traffic is encrypted or when incorporating TCP-awareness in intermediate node is not feasible. Further research can be carried out to solve this problem.

This research mainly focused on quantifying improvements achievable using loss notifying method and was therefore limited to long-lived TCP flows. Further research can be carried out to isolate the effects of loss during the slow start phase and quantify the benefits of Explicit Loss Notification Scheme for short-lived flows.

This scheme assumes that acknowledgement packets never get corrupted or lost due to high BER characteristic of wireless medium. Further study can consider the situation when data packet as well as acknowledgement packet get corrupted or lost in wireless medium.

The mechanism can be evaluated using real network topologies including other workloads, for example Hypertext Transfer Protocol (HTTP) transaction. The effectiveness of this scheme can also be evaluated under other error models such as channel fades and Gilbert model [24]. Further research can be carried out to extend or modify this algorithm so that this scheme can be able to differentiate random packet loss from congestion loss.

## REFERENCES

- [1] Vaidya, V. and Jeyakumar, A., (2016) "A Review of Use of TCP over Wireless Cellular Networks", International Journal of Advanced Research in Computer and Communication Engineering", Vol 5, Issue 7, pp 44-48, ISSN (Online): 2278-1021, ISSN (Print): 2319-5940, July 2016.
- [2] Bidwe, R. V., (2013), "Different Issues and Survey of Proposed Solutions in TCP over Wireless Environments", International Journal of Future Computer and Communication, Vol. 2, No. 6, pp 556-561, December 2013.
- [3] Paul, L. C., (2013), "Handoff/Handover Mechanism for Mobility Improvement in Wireless Communication", Volume 13, Issue 16, Publisher: Global Journals Inc (USA), Online ISSN: 2249-4596 & Print ISSN: 0975-5861, 2013.

- [4] Ahn, J., Danzig, P., Liu, Z. and Yan, L., (1995) "Evaluation with TCP Vegas: Emulation and experiment," ACM SIGCOMM Computer Communication Review, Vol. 25, pp. 185-195, Cambridge, Massachusetts, USA, August 28 – September 1, 1995.
- [5] Brown, K. and Singh, S., (1997), "M-TCP: TCP for Mobile Cellular Networks", ACM Computer Communications Review, Vol. 27, No. 5, pp. 19-43, July 1997.
- [6] Chawla, R. and Nandi, S., (2002), "TCP FECN: a unified solution for wireless networks", Proceedings of The 8<sup>th</sup> International Conference on Communication Systems (ICCS 2002), Vol. 2, pp. 815 -819, Singapore, November 25-28, 2002.
- [7] Nanda, S., Ejzak, R. and Doshi, B., (1994), "A Retransmission Scheme for Circuit-Mode Data on Wireless Links", IEEE Journal on Selected Areas in Communications, Vol.12, No. 8, pp. 1338-1352, October 1994.
- [8] Ayanoglu, E., Paul, S., LaPorta, T., Sabnani, K. and Gitlin, R., (1995), "AIRMAIL: A link-layer protocol for wireless networks", ACM/Baltzer Wireless Networks Journal, Vol. 1, No.1, pp. 47-59, February 1995.
- [9] Caceres, R. and Iftode, L., (1995), "Improving the performance of reliable transport protocols in mobile computing environments", IEEE Journal on Selected Areas in Communications, Vol. 13, No. 5, pp. 850-857, June 1995.
- [10] Bakre, A. and Badrinath, B., (1995), "I-TCP: Indirect TCP for Mobile Hosts", Proceedings of 15<sup>th</sup> IEEE International Conference on Distributed Computing Systems, pp. 136-143, Vancouver, British Columbia, Canada, May 30 – June 2, 1995.
- [11] Xylomenos, G., Polyzos, G., Mahonen, P. and Saaranen, M., (2001), "TCP Performance Issues over Wireless Links", IEEE Communications Magazine, Vol. 39, No. 4, pp. 52-58, April 2001.
- [12] Balakrishnan, H., Seshan, S. and Katz, R., (1995), "Improving reliable transport and handoff performance in cellular wireless networks", ACM Journal of Wireless Networks, Vol. 1, No. 4, pp. 469-481, December 1995.
- [13] Hadjiefthymiades, S., Papayiannis, S. and Merakos, L., (2001), "TCP Performance Enhancement in Wireless / Mobile Communications", Proceedings of 26<sup>th</sup> Annual IEEE Conference on Local Computer Networks, pp. 604-610, Tampa, Florida, November 14-16, 2001.
- [14] Hadjiefthymiades, S., Papayiannis, S. and Merakos, L., (2001), "TCP Performance Enhancement Wenqing, D and Jamalipour, A. (2001), "Delay performance of the new explicit loss notification TCP technique for wireless networks", Global Telecommunications Conference, GLOBECOM '01. IEEE, Vol. 6, pp. 3483-3487, San Antonio, TX, November 25-29, 2001.
- [15] Perkins, C. (1996), "IP Mobility Support", Network Working Group, Editor Request for Comments: 2002, October 1996.
- [16] Elaarag, H., (2002), "Improving TCP Performance over Mobile Networks", Journal of ACM Computing Surveys, Vol. 34, No. 3, pp. 357-374, September 2002.
- [17] Balakrishnan, H., Seshan, S. and Katz, R., (1998), "Explicit loss notification and wireless web performance", Proceedings of IEEE Globecom Internet Mini-Conference, Sydney, Australia, November 8-12, 1998.
- [18] NS, The network simulator – ns-2. <URL:<http://www.isi.edu/nsnan/ns>>
- [19] Jacobson, V., (1988), "Congestion avoidance and control", Proceedings of the ACM Symposium on Communications Architectures and Protocols, Vol. 18, No. 4, pp. 314-329, Stanford, CA, USA, August 16-18, 1988.
- [20] Stevens, W., (1997), "TCP Slow Start, Congestion Avoidance, Fast Retransmission and Fast Recovery Algorithms", Internet Engineering Task Force (IETF), RFC-2001, January 1997.
- [21] Jacobson, V., (1990), "Modified TCP Congestion Avoidance Algorithm", end2end-interest mailing list, April 30, 1990. <URL: <ftp://ftp.isi.edu/end2end/end2end-interest-1990.mail>>
- [22] Hoe, J. C., (1996), "Improving the Start-behavior of a Congestion Control Scheme for TCP", Annual conference of the Association for Computing Machinery's Special Interest Group on Data Communication (ACM SIGCOMM '96), pp.270-280, California, USA, August 26-27, 1996.
- [23] Mathis, M., Mahdavi, J., Floyd, S. and Romanow, A., (1996), "TCP Selective acknowledgement options", IETF, RFC 2018 (Status Proposed Standard), 1996. <URL: [www.rfc-editor.org/rfc/rfc2018.txt](http://www.rfc-editor.org/rfc/rfc2018.txt)>
- [24] Gilbert, N., (1960), "Capacity of a burst-noise channel," Bell Systems Technical Journal, Vol. 39, pp. 1253--1265, September 1960

## Authors

**Nayeema Islam** received her M.Sc. in Telecommunication Engineering from Asian Institute of Technology, Thailand in 2005. Also she received her M.Sc. and B.Sc. in Computer Science and Technology from Rajshahi University, Bangladesh in 1998 and 1997 respectively. She is presently working as a lecturer in the department of Computer Science and Engineering in Jagannath University, Dhaka, Bangladesh. Her research fields of interest include Cellular Mobile System, Ad-hoc and Sensor Networks, Network Protocols, and QoS Routing.



**Mohammed Nasir Uddin** was born in 1974 in Bangladesh. He received his Ph.D. degree from Moscow Power Engineering Institute, Russia in 2004. He also received his M.Sc. and B.Sc. in Computer Engineering from State University of Lvivska Polytechnica, Ukraine. He is presently working as an associate professor in the department of Computer Science and Engineering in Jagannath University, Dhaka, Bangladesh. His research fields of interest include Cellular Mobile System, Ad-hoc and Sensor Networks, Cryptography and Network Security.

