## A Simulation Study of the Impact of Mobility on TCP/IP

Pietro Manzoni\*
Politecnico di Milano
P.za L. Da Vinci, 32
Milan, Italy 20133
manzoni@elet.polimi.it

Dipak Ghosal
Bell Communications Research
331 Newman Springs Road
Red Bank, NJ 07701
ghosal@nyquist.bellcore.com

Giuseppe Serazzi Politecnico di Milano P.za L. Da Vinci, 32 Milan, Italy 20133 serazzi@elet.polimi.it

#### Abstract

This paper presents a simulation analysis of the impact of mobility on data transport protocols. We consider the TCP/IP protocol augmented with features to support host mobility in wide area networks. Our results show that the existing version of TCP can yield low throughput in highly mobile environments due to the fact that TCP cannot discriminate packets dropped due to hand-offs with those dropped due to congestion in one or more network resources. We investigate a proposal in which the transport layer explicitly receives information from the network layer of any ongoing mobility. We show that by effectively capitalizing this information, TCP can appropriately extend the slowstart phase in the recovery process and achieve higher throughput. Based on the simulation analysis we also show the robustness of this scheme in the presence of both host mobility and network congestion.

#### 1 Introduction

This paper presents a study to determine the impact of mobility on the transport layer performance of the TCP/IP protocol suite [1] augmented with features to support mobile wireless data-stations (referred to as mobile hosts (MH) in this paper) in a wide area network (WAN). The modifications to the protocol take into account the following features peculiar to mobile hosts. First, in order for a new connection to be established with a mobile host, the network must find its current location. This may require the mobile host and the network to participate in maintaining current location information in some a priori determined database. Second, since the mobile host can roam from one network to another during an active connection, signalling between network nodes is required to perform the hand-off so that the packets can be correctly routed to the mobile host. A number of studies have been reported in the literature dealing with various aspects required to support host mobility on the Internet [2], [3], [4], [5], [6], [7]. The similarities and differences of these proposals are clearly summarized in [8].

The issues related to the impact of mobility on data transport protocol have been recently addressed in [9].

This paper extends the above work in many respects. We develop a simulation model which incorporates all the details of the TCP as well as the underlying mo-bile Internet Protocol. The simulation tool allows us to do a detailed sensitivity analysis with respect to key parameters such as the round trip time, the delay involved in the hand-off process and the window size. Our results show that in highly mobile environments the overall TCP throughput can be low, since the standard version of TCP cannot discriminate between packets dropped due to mobility and packets dropped due to congestion in network resources. As a result, if packets are lost during hand-offs, TCP initiates a recovery process which consists of the slow-start phase followed by the congestion avoidance phase. The slow growth of the TCP window during the congestion avoidance phase results in throughput loss. The impact is exacerbated for TCP connections with large windows and/or "hard hand-offs" causing long pauses in communication.

We investigate a modification to the TCP recovery process which explicitly uses network layer information of any ongoing mobility. In this scheme, TCP extends the slow-start phase in the recovery process if there is explicit knowledge of ongoing mobility. Based on the simulation model, we quantitatively demonstrate the efficacy of the protocol modification by showing that the throughput of the modified protocol converges to the basic TCP protocol as the network packet loss increases.

The remainder of this paper is organized as follows. Section 2 briefly discusses the wireless network that interfaces the MH with the Internet. Section 3 describes the protocol features that have been added to IP to support host mobility. Section 4 describes the simulation model developed to carry out the analysis of the transport and network layer performance. Section 5 discusses impact of mobility on the transport layer performance. Finally, Section 6 gives the concluding remarks and some future research directions.

#### 2 The wireless network

A logical view of the underlying wireless network and the manner in which it interfaces with the Internet are shown in Figure 1. In a regional scale, the geographic area is assumed to be divided into location areas which are in turn divided into cells. Each

<sup>\*</sup>Work partially supported by MURST (40% and 60% projects)

cell has a base station which provides the radio interface with wireless mobile hosts. The base stations are connected to switching offices, referred to as Mobile Switching Centers (MSC), which are capable of separating the voice and the data parts in the recieved packets. The data part is processed by an intermediate system such as the Mobile Data Intermediate System defined in the CDPD protocol [13]. This intermediate system could be connected directly to an Internet gateway or via private/public data networks.

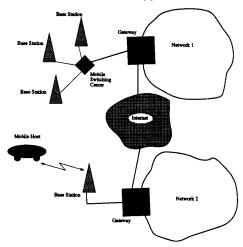


Figure 1: Interconnection between the Mobile Host (MH) and the Internet.

A mobile host can be connected to any network and is allowed to roam into other networks without requiring any address change. The main difference between a mobile host and a fixed host is that while a fixed host requires just one address to be accessed, the mobile host requires two addresses - 1) the mobile host address and 2) the address of the currently controlling gateway. When a gateway receives a packet for a mobile host which it currently controls, it simply forwards the packet to the appropriate base station through which the mobile host can be reached. In general, a gateway can be associated with one or more MSCs and this would depend on the type of cellular technology used [10], [13]. In the following discussion we will associate a set of port numbers with each gateway through which mobile hosts can be reached. Sending a packet to a mobile host will be done by writing the packet into the appropriate port. Aspects of communication related to the data link and the physical layers are not addressed in this paper.

#### 3 Mobile Internet Protocol

We consider a simple protocol which is generic in nature and essentially encompasses most of the key features of the different protocols discussed in the literature and in the various mailing lists. The differences that do exist are unlikely to make any significant impact to the conclusions that are drawn from this performance study.

The protocol described below is designed based on three underlying suppositions. First, determining the gateway which controls the mobile host by broadcasting queries to all the gateways may be too expensive in a wide area network. As a result, maintaining location information tables for mobile hosts in a fixed location is likely to be more appropriate from a performance perspective. Second, migration of the mobile host should be supported primarily at the network layer, i.e., IP. Although this view is widely accepted [3] [4], the results in the paper show that some interaction between the transport layer and the network layer is needed to alleviate the impact of mobility on the transport layer throughput. Finally, the likelihood of a mobile host to have a complete memory loss failure is much higher than that of a gateway. Thus, the burden of maintaining information is placed on the gateway rather than on the mobile host.

There are two main parts in this protocol. First part deals with the procedure by which a mobile host maintains connection with the controlling devices and the way in which information about the current position of the mobile host is updated in the relevant databases. The second part describes the scheme by which any host (fixed or mobile) retrieves information about the mobile host to which it wants to send data and the manner in which the connection is maintained as the mobile host roams across network boundaries.

## 3.1 Terminology

We introduce the following terminology:

- Home Gateway (HG): the gateway controlling
  the network to which a mobile host is assumed to
  be registered. It is likely for a network to have
  more than one gateway, in which case it will be
  necessary to statically designate one of the gateways as the home gateway. The home network
  may be the network in which the mobile host remains connected most of the time.
- Currently Controlling Gateway (CCG): the gateway which is currently controlling the mobile host. Datagram destined to a mobile host has to pass through the CCG.
- Previously Controlling Gateway (PCG): a gateway which controlled the mobile host before being controlled by the CCG. As will be shown in the subsequent discussion, maintaining MH information in the PCG for some time may significantly reduce the traffic through the home gateway.

With respect to a gateway, there are three types of hosts, namely, fixed hosts, local mobile hosts, and visitor mobile hosts. Fixed hosts are those that do not roam and hence do not change their point of attachment to the network, local mobile hosts are those for which the gateway is the HG, and visitor mobile hosts are those for which the gateway is a CCG or a PCG.

#### 3.2 Mobile host table structure

Each gateway maintains mobile hosts' information in a table called *Mobile Host Table* (MHT) which has an entry for each local host and an entry for each visitor host that it currently controls. entry for visitor hosts. Each entry in the MHT consists of the following fields:

- Mobile Host Address: the address of the corresponding mobile host.
- CCG Address: the address of the CCG.
- Port Address: the address of the radio equipment, i.e., the base station through which the MH can be reached.
- Time To Live (TTL): this field holds different values depending on whether the MH is a local or a visitor. If the MH is a local then TTL is set to a infinite value implying that the entry can never be deleted. If the MH is a visitor, then TTL is set to a "time-out" value. As soon as the visitor MH goes out of the area controlled by the gateway, this field is counted down. When TTL becomes 0, the entry is deleted from the MHT. This "time-out" determines how long PCGs maintain information about MHs.

#### 3.3 Datagram modification

MHs are identified by two addresses: the MH address and the CCG address. In order to support this new requirement, the existing IP datagram structure is modified as shown in Figure 2. As the figure shows, the "IP option" field was utilized and two new fields were added. The "IP option" field is now divided into three parts:

0	4	0	8	16	19	31	
VERS	HLEN	VERS	SERVICE TYPE		TOTAL LENGTH		)
	IDENTIFICATION			FLAGS	FRAG. OFFSET		mmodified
TIME '	TIME TO LIVE PROTOCOL			HEADER CHECKSUM		≻ IP datagram	
			fields				
			)				
ОРТЮ	TYPE	OPTION	OPTION LENGTH		DATAGRAM TYPE		modified fields
			new field				
	NEW ADDRESS						new field
	deta						
data							
F							
i						ŀ	
			new field				

Figure 2: The modified datagram to support mobile hosts in TCP/IP.

- Option Type: A one-byte long field with a suggested value of 33. This field identifies the new datagram scheme required to support host mobility.
- Option Length: A one-byte long field which is used to discriminate whether we are using both "auxiliary address" and the "new address" (option length equal to 10) or we are just using the "auxiliary address" (option length equal to 6).

• Datagram Type: This field uses the remaining 16 bits to define the following 11 types of messages whose functions are explained in the following subsection.

ack\_euw, ack\_pcg, careof, data, itishere, mynameis, stillhere, tryhome, trythere, umustfwd, welcome.

The two new fields are the Auxiliary Address, which is primarily used to pass the permanent address of the destination MH, and the New Address, which is mainly used to carry the address of the new CCG of the MH.

#### 3.4 Protocol messages

The sequence of messages required to perform the hand-off is shown in Figure 3. It includes three phases: 1) registering the MH with the new CCG, 2) updating the MHT in the PCG, and 3) updating the MHT in the HG. The interaction is initiated from the MH, which sends a mynameis message to the CCG, which acknowledges with a welcome message:

MH	broadcast	MH-HG	-	mynameis
MH-CCG	MH	-	-	welcome

If the receiving gateway determines that it is already the CCG of that MH, the operation terminates. If the gateway has no record of the mobile host in its MHT, it is considered as a new visitor host and a new entry is created in the MHT. The CCG then informs the MH's HG and PCG to update their MHTs by sending two itishere messages:

## MH-CCG | MH-HG | MH | - | itishere

The order in which this is done will determine the delay in performing the hand-off. In this paper we consider the situation in which the PCG is updated first. This reduces the delay in performing the hand-off for on-going communications.

off for on-going communications.

When the PCG receives the itishere message, it replies with an ack\_pcg message. The CCG then sends the second itishere message to the home gateway which replies with an ack\_euw message indicating the termination of the update operation:

MH-HG	MH-CCG	MH	MH-PCG	ack_pcg
MH-PCG	MH-CCG	MH	-	ack_euw

If the previously controlling gateway was also the home gateway, it responds with the ack\_euw message. As shown in Figure 3, the CCG periodically sends a stillhere message to the home gateway. This message, whose format and content is similar to that of the itishere message, is required for protocol robustness against gateway failures.

IP-routing uses the destination address field in the datagram to decide the next hop. Routing packets to the mobile hosts requires a combination of two addresses: 1) the permanent address of the MH and 2) the address of its CCG. First, plain IP-routing scheme is used to deliver packets to the CCG and then the permanent address is used to send the packet to the appropriate MH.

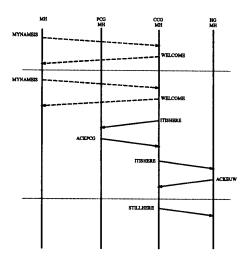


Figure 3: Signalling required for updating mobile host tables.

The sequence of messages required for setting up and maintaining a conversation [11] is shown in Figure 4. In the following discussion we consider conversations in which both the source and the destination hosts are mobile hosts; these will be referred to as MH-S and MH-D, respectively.

If the source machine (MH-S) does not have any information about the CCG of the MH-D, the first message of the conversation will be sent to the MH's home network:

When the HG receives this message it starts a search for MH-D in the MHT. Three situations can occur:

- No related entry is found, in which case the HG will use the standard ARP protocol to search for a fixed host in the local network.
- The CCG field in the MHT's entry holds the address of the HG. In this case the message is forwarded to the base station via the port address obtained from the same entry in the MHT.
- The CCG field in the MHT's entry holds a CCG address. The HG will send back an umsutfwd message to MH-S indicating in the new\_address field the address of MH-D's CCG address:

## MH-D | MH-S-CCG | MH-S | MH-D-CCG | umustfwd

When the MH-S receives the above message it sends it again but now to the CCG of the MH-D, and as a careof message:

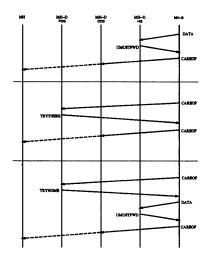


Figure 4: Signalling required for setting up and maintaining a conversation.

A careof message received by a MH-D-D's CCG is interpreted as an indication that the message is for a mobile host which it currently controls. The message is routed by the CCG to the MH-D through the appropriate port address obtained from the corresponding entry in the MHT.

Since the MH can roam from one network to another, it is likely that packets may reach a gateway which has already become a PCG. As mentioned before, in our protocol a gateway which has become a PCG keeps information about the visitor host for a certain period, whose duration is determined by the TTL counter in the MHT. This counter is decremented each time an access to the entry is made after the entry has been marked as related to a previously controlled MH. The entry is deleted when the counter reaches zero. This ensures that the size of the MHT does not grow indefinitely.

If the PCG still has the entry corresponding to the MH-D, it will send a trythere message to the MH-S:

## MH-D | MH-S-CCG | MH-S | MH-D-CCG | trythere

On receiving this message, the MH-S will send the packet, still tagged as careof, to the CCG in the "new address field" in the trythere message. If in the meantime this gateway also becomes a PCG, then the above operation is repeated and another trythere message is sent back to the MH-S. If the MHT in gateway has no entry corresponding to MH-D, it sends a tryhome message to the MH-S, indicating to the source to request the home gateway for MH-D's CCG:

## MH-D MH-S-CCG MH-S - tryhome

The time for which a PCG maintains information about the MH is an important parameter. This value

would depend on the characteristics of the mobility, namely, the size of the geographic area, the mobility rate, and the concentration of mobile hosts among other performance and fault-tolerant considerations.

#### 4 Simulation model

To carry out a detailed analysis of the impact of host mobility on the transport and network layer performance, a queueing network simulation model was developed. The model is defined in terms of stations (one or several servers and one queue) through which "customers" (i.e., packets) can circulate according to some given routing rules. The customers are distributed into several classes characterized by different routing rules and different service requests in the stations. The processing in each station can be described by a simple delay drawn from a given probability distribution or by a complex algorithm which may include synchronization operations. Asynchronous events such as timers and exception handling were needed to model the signalling algorithm.

The simulation model developed for this study has the following important features. First, it accurately models the TCP protocol, including the dynamic window flow control algorithm, retransmission of timedout packets with appropriate time-out value based on round-trip time estimates and the Van Jacobson's congestion control algorithm [12]. Second, it accurately models the mobile Internet Protocol described in Section 3. Finally, it incorporates a reasonably accurate model of the network delays which was obtained by fitting probability distribution function to actual network delay measurements made using the Unix ping command.

The queuing network model is shown in Figure 5. It

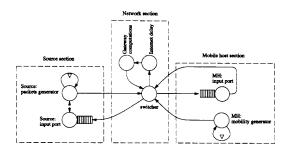


Figure 5: The simulation model.

consists of three distinct sections: 1) the mobile host section which models the behavior of the destination mobile host, 2) the network section which models the Internet, and 3) the source section which models the fixed source host.

### 4.1 Mobile host section

The mobile host section performs two activities: sending acknowledgments for packets received from the source and generating packet which model the mobility of the mobile host. In this paper the intermobility time, denoted by *imt*, is assumed to be constant. The model consists of a pure delay element with a single token which is held in the station until the inter-mobility delay expires at which time a mynameis packet is sent to the *switcher* station and the token is returned to the delay center. The actions pertaining to the acknowledgments are carried out in a dedicated station which maintains a table of received packets and sends acknowledgments using the cumulative acknowledgment technique described in [1].

#### 4.2 Source section

The source section models the data packet source which is assumed to be a fixed host. It consists of two stations: 1) the packet generator and 2) the input port. The packet generator has a single token which goes in and out of this station every inter-packet time which is the delay between two consecutive packets. When the token arrives, the station checks whether it can send a packet by checking if the difference between the number of the last packet sent and the number of the last packet sent and the window size.

The input port receives incoming customers which can either be control packets or acknowledgments from the destination. Control packets are trythere messages that are received when the destination mobile host roams across two networks. These messages are used to update the cache entries in the source host. Acknowledgments are used to update the value of a status variable that keeps track of the last acknowledged packet.

Besides these two tasks, this station also maintains a single timer to detect packet loss. For this purpose the source records the transmit time of each packet that is sent. When an acknowledgment is received, the timer is restarted with time-out value which is equal to the time at which the first unacknowledged packet was sent plus the new round-trip time estimate multiplied by  $\beta$  (in this study  $\beta=2$  [1]). The expiry of the timer is viewed as a packet loss and it invokes the Van Jacobson's congestion control algorithm discussed in the following section.

#### 4.3 Network section

The network section is made up of three stations: 1) the Internet delay station, 2) the switcher station, and 3) the gateway computation station. These stations correspond to the three main effects that a WAN has on IP packets namely, the switching effects, the delays due to network congestion, and the packet processing at the gateways.

The Internet delay models the delay in the network. Packets sent over the Internet go from the initial point to the final point via many gateways. Since we do not model each intermediate gateway with a different station, the network delay was modeled using a unique station. This element behaves as a pure delay station. The probability density function of the delay is sum of a constant and an Erlang distribution with 2 stages and mean  $\mu=0.01$ . This was obtained based on a

statistical analysis of round-trip time measurements made using the Unix ping command.

The switcher station models the actual routing of packets. When the source send a data packet, the switcher routes it to the delay station and then it is passed through the gateway computation station which routes the packet to the MH station. If a packet has to be sent from one gateway to another, e.g., when the CCG sends an itishere packet to the PCG, it is always done using the same gateway computation station but the packet is first passed through the switcher and the delay stations. The network congestion is also modeled by the switcher, which drops one or more packets at a predefined rate.

The gateway computations station performs all functions that are associated with packet handling. All tasks pertaining to control packets defined for the mobile Internet Protocol in Section 3 are performed in this station.

# 5 Impact of mobility on the transport layer performance

Although there are a number of different implementations of TCP [1], in this study we consider the version which incorporates the Van Jacobson's congestion control scheme [12]; this version is known as TCP-Tahoe. Below we highlight the key aspects of TCP which have been implemented in the simulation model. The details of TCP can be found in [12], [1].

- We consider data transfer in one direction, from source to destination. We assume that the source is heavily loaded implying that it always has data to send.
- We use W<sub>max</sub> to denote the maximum window size that is negotiated between the source and the destination during the initiation of the TCP connection. Furthermore, we also assume that the source transmits data in equal size packets of length 512 bytes.
- The source has a single timer and uses Karn's algorithm [1] to set timeout values for retransmitted packets.
- For the purpose of the congestion control scheme, the source maintains two variables, W which denotes the current window size and Wthresh which denotes the window threshold. Upon detecting a packet loss (expiry of the timer), the window threshold is set to half the window size and the window size is set to one. The recovery process consists of two phases; first the slow-start phase followed by the congestion avoidance phase. In the slow-start phase the window size is incremented by 1 for every acknowledgment received by the source. The slow-start phase continues until the window size is equal to Wthresh after which the congestion avoidance phase takes over. In this phase the window size is incremented by one for every window size number of acknowledgments.

The destination host uses cumulative acknowledgment scheme described in [1] for acknowledging the received packets.

There are a number of important parameters of interests, namely, the maximum window size,  $W_{max}$ , the mean round trip time, rtt, the delay in performing the hand-off, denoted by  $\gamma$ , and the inter-mobility time, denoted by imt. Furthermore, a TCP connection can be characterized as being either host limited or network limited. In a host limited connection the network is assumed to have infinite bandwidth and there is no packet loss in the network. The maximum throughput in this case depends on the round trip time and the window size that is negotiated between the source and the destination hosts. On the other hand, in a network-limited TCP connection, the network losses packets due to congestion in one or more network resources. In this study, a network limited connection is characterized by a network packet loss rate.

Figure 6 shows the throughput trace for an host limited TCP connection when the mobile host moves deterministically every 120 seconds. The other parameters are given in the figure caption. In our simulation model, we assume that gateways do not buffer any packets. When a gateway receives a packet for a valid mobile host, it simply hands the packet to an appropriate link layer processor which simply drops the packet if there is no link to the mobile host. Subsequent to a hand-off, it takes  $\gamma$  time units for the gateway to know the address of the new controlling gateway and all packets received during this interval are lost. From the trace it is clear that despite the loss of a number of packets, the  $W_{thresh}$  decreases to at most half the maximum window size.

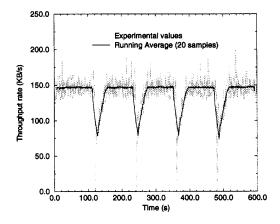


Figure 6: Throughput trace of a host-limited TCP connection  $[W_{max} = 64KB, rtt = 300ms, \gamma = 0.5$  and imt = 120 s].

The impact of window size is shown in Figure 7. The results are shown for an host-limited TCP connection for two different mobility rates. From the figure it is clear that as the window size increases, the throughput decreases.

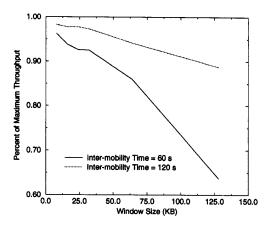


Figure 7: Impact of window size on the throughput of a host-limited TCP connection [rtt = 300ms,  $\gamma = 0.5$ ].

The larger the window size the longer it takes for the source host to recover to the nominal throughput levels and also as expected the decrease in the throughput is higher for higher mobility rates. The above result raises an important issue. For TCP connections with high bandwidth-delay products, it will be necessary to have large windows implying that impact of mobility will be worse. In order to alleviate this problem it will be necessary to determine approaches by which TCP can discriminate between packets dropped due to network congestion and packets dropped due to mobility.

The impact of the hand-off delay is shown in Figure 8. The result show that the exponential timer back-off scheme plays an important role in stabilizing the impact of large hand-off delay.

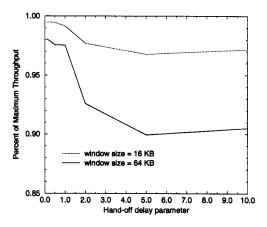


Figure 8: Impact of hand-off delay ( $\gamma$ ) on the throughput of a host-limited TCP connection [rtt = 300ms, imt = 120s].

### 5.1 A protocol modification

We study the following simple protocol modification which explicitly uses network layer information of any ongoing mobility to extend the slow-start phase in the recovery process. In this scheme the host maintains two registers: 1) paddress which contains the address provided in the trythere message and 2) mcregister, (the mobility-congestion register) which keeps status information and can assume values one of 1, 0, and -1. Each time a trythere is received, if the address in the new-address field is different from that in paddress and mcregister is not equal to one, the paddress is updated and/or the mcregister set to one. On each timeout, mcregister is decremented by one if it is greater than -1. When the window reaches the maximum window size the counter is set to 0.

During the recovery process, the mcregister value is used as follows: if the value is greater than or equal to 0, then the recovery process consists of only the slowstart phase, whereas if the value is negative then the recovery process is the normal TCP recovery process, namely, the slow-start phase followed by the congestion avoidance phase. The key idea here is that the mcregister maintains the state of the connection. If the connection is network limited, then the timeouts due to packet dropped in the network will maintain the value of the register to -1 most of the time. In this case, the recovery process will follow the normal TCP recovery process. On the other hand, if the connection is host limited than the value of the mcregister is likely to be 0 most of the time and will become 1 when trythere messages are received. Recovery process in this case should quickly increment the window to the maximum widow size by simply extending the slow-start phase. Note that, including state 0 of the mcregister in the extended slow-start recovery process is an "optimistic" approach. The decision process can be made more "pessimistic" by choosing the normal TCP recovery process when the mcregister is in state

The comparison of the above protocol with the basic TCP protocol for a host-limited connection is shown in Figure 9.

From the result is clear that using network layer information to extend the slow-start phase can result in significant throughput improvement particularly for large window sizes.

We also addressed the robustness of the above protocol when there is both network congestion as well mobility. Figure 10 shows the comparison of the percent of the maximum throughput for the basic TCP protocol and the modified protocol for different packet loss rates. The result shows the behavior that is desired. At very low packet loss rates, the modified protocol has a higher throughput than the basic TCP protocol. As the packet loss rate increases the throughput of the modified protocol converges to that of the basic TCP protocol. For very high packet loss rates the connection is network limited and source should adopt the congestion avoidance phase of the basic TCP in order to avoid network collapse.

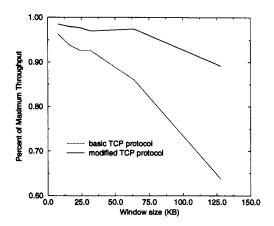


Figure 9: Impact of protocol modification on the throughput of a host-limited TCP connection  $[rtt = 300ms, \gamma = 0.5, imt = 120s]$ .

## 6 Concluding remarks

In this paper we have presented a detailed simulation analysis of the impact of mobility on the network and transport layer performance of the TCP/IP protocol suite augmented with network layer features to support host mobility. We showed that packet loss during hand-offs can significantly reduce the mean throughput particularly in highly mobile environments and connections with large window sizes. The loss of throughput is due to the fact that existing versions of TCP cannot discriminate between packet loss due to mobility and those due to congestion in one or more network resources.

We considered a simple modification to TCP which explicitly uses network layer information about mobility to extend the slow-start algorithm during the recovery process. We show that this leads to significant improvement in the throughput for host limited TCP connections. We also demonstrated the robustness for network limited TCP connections by showing that as the packet loss rate increases the modified protocol converges to the basic TCP protocol.

Further research is needed to investigate other approaches to help TCP discriminate between host mobility and network congestion. Although most schemes would yield improvement in throughput, the key factor will be the ease with which the modification can be incorporated in the existing software.

## References

- Douglas E. Comer, Internetworking with TCP/IP, Volume 1, Principle, Protocols, and Architectures, Prentice-Hall, Englewoods Cliffs, New Jersey 07362.
- [2] Charles Perkins and Yakov Rekhter, "Short-cut routing for mobile hosts," IBM T J Watson Research Center, Working Draft, 17 July 1992.
- [3] Fumio Teraoka and Yasuhiko Yokote and Mario Tokoro, "A Network Architecture Providing Hosts

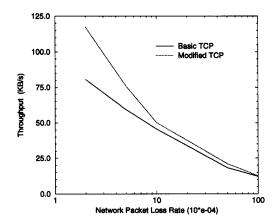


Figure 10: Robustness of protocol modification for a network-limited connection with different degrees of network congestion [rtt = 300ms,  $\gamma = 0.5$ , imt = 120s].

- Migration Transparency," ACM SIGCOMM 1991, pp. 209-220.
- [4] John Ioannidis and Dan Duchamp and Gerald Q. Maquire, "IP Based Protocols for Mobile Internetworking," ACM SIGCOMM 1991, pp. 235-245.
- [5] Hiromi Wada and Tatsuya Ohnishi and Brian Marsh, "Packet Forwarding for Mobile Hosts," Working Draft November 1992.
- [6] C. Sunshine and J. Postel, "Addressing mobile hosts in the ARPA Internet environment," IEN 135, March 1980.
- [7] D. Ghosal and P. Manzoni, "Performance of a Network Layer Protocol Supporting Host Mobility in a Wide Area Network," IEEE Workshop on High-Performance Communications Subsystems, Williamsburg, VA, September 1993.
- [8] Andrew Myles and David Skellern, "Comparison of Mobile Host Protocols for IP," Technical Report, Macquarie University, April 1993.
- [9] Ramon Caceres and Liviu Iftode, "The Effects of Mobility on Reliable Transport Protocols," MITL Technical Report, MITL-TR-73-93, November 1993.
- [10] K. S. Meier-Hellstern, E. Alonso and D. R. Neil, " The Use of SS7 and GSM to Support High Density Personal Communications," Winlab Workshop, April 1992.
- [11] Ramon Caceres and Peter B. Danzig and Sugih Jamin and Danny Mitzel, "Characterization of Wide-Area TCP/IP Conversations," ACM SIGCOMM 1991, pp. 101-112.
- [12] Van Jacobson, "Congestion Avoidance and Control," ACM SIGCOMM Proceedings, 1988, pp. 314-329.
- [13] Cellular Digital Packet Data System Specification, Preliminary Release 0.9, April 1993.