Measuring the Performance of Call Admission Control in Two Tier Wireless Cellular Communication Networks

Sumit kumar srivastava[1], Dharmendra Kumar Gupta[2]

M. Tech. (CSE) Student & Faculty of Computer Science & Engineering, D.B.I.T, Dehradun, Uttarakhand[1]

M. Tech. (CSE) Student & Faculty of Computer Science & Engineering, IET Alwar, Rajsthan[2]

ABSTRACT

A mobile ad hoc network (MANET) is a spontaneous network that can be established with no fixed infrastructure. This means that all its nodes behave as routers and take part in its discovery and maintenance of routes to other nodes in the network. Its routing protocol has to be able to cope with the new challenges that a MANET creates such as nodes mobility, security maintenance, quality of service, limited bandwidth and limited power supply. These challenges set new demands on MANET routing protocols. With the increasing interest in MANETs, there has been a greater focus on the subject of securing such networks. Out of the many discussions and research groups discussing the different security issues in the field of mobile ad hoc networks, many papers have been written describing different proposed secure routing protocols that defend against malicious nodes' attacks that MANETs face. However, the majority of these MANET secure routing protocols did not provide a complete solution for all the MANETs' attacks and assumed that any node participating in the MANET is not selfish and that it will cooperate to support different network functionalities. Recently, researchers started to study selfish nodes and their effects on mobile ad hoc networks. That resulted in creating a new thread of research in the MANET field. A number of research papers discussing different cooperation enforcement schemes for detecting and defending against selfish nodes and their disturbance to mobile ad hoc networks were published. Still none of these proposed cooperation enforcement schemes were based on any existing MANET secure routing protocols. All of these proposed schemes were based on routing protocols with no security measures at all. My research strategy is to choose one of the secure routing protocols according to its security-effectiveness, study it and analyze its functionality and performance. The authenticated routing for ad hoc networks (ARAN) secure routing protocol was chosen for analysis. Then, the different existing cooperation enforcement schemes were surveyed so that to come up with a reputation-based scheme to integrate with the ARAN protocol. The result of that integration is called: Reputed-ARAN. Consequently, Reputed-ARAN is capable of handling both selfish and malicious nodes' attacks. Also, the Glomosim simulation package was chosen to carry out the experimental part of this thesis work. The results of the experiments showed that in the presence of 30% selfish nodes and with node mobility of 10 m/s, the newly proposed Reputed-ARAN protocol improves network throughput to 63.1%, from 38.8% network throughput provided by normal ARAN. This improvement is obtained at the cost of a higher overhead percentage with minimal increase in the average number of hops. The main contribution in this thesis: Reputed-ARAN proves to be more efficient and more secure than normal ARAN secure routing protocol in defending against both malicious and authenticated selfish nodes.

Manuscript received January 5, 2013
Manuscript revised January 20, 2013

INTRODUCTION

The development of affordable palmtop devices with built in high-speed radio interfaces will have a major impact on the mobile communications industry. Large numbers of mobile users equipped with wireless Internet enabled communicators will require access to web based services anywhere anytime. The ubiquitous availability of wireless Internet access may supersede the popularity of cellular telephony and change the way we communicate. This environment places significant demand on existing and next generation mobility solutions. The recent years have seen a rapid development of mobile communications technology. The cellular principle allows for the efficient use of the scarce radio resources and helps to support large subscriber populations. Advances in microelectronics, on the other hand, have made cellular telephones a commodity. The growing number of cellular phone users suggests that mobility will soon become the norm in communications, rather than the exception. While state of the art cellular mobile systems are still optimized for voice communication, they support an increasing variety of data services [13], [61]. Recent initiatives to augment the Internet with mobility support indicate the increasing interest in mobile data services [15], [14]. Future technologies for the support of wireless Internet access should leverage experiences from both cellular telephone systems and Internet technology. Flexible and scalable solutions are required that can adapt to a wide range of environments. Users must be offered seamless mobility across possibly heterogeneous systems which need to interact and co-operate to provide the best service available. The efficient use of the wireless interface, which continues to be the bottleneck in mobile communications, will become increasingly important with the emergence of mobile multimedia services. In this dissertation we address some of the challenges imposed by the design and analysis of wireless mobile communication systems in this new environment.

RELATED WORK

To give more prospective about the performance of the Two Tier Wireless Cellular Communication Networks, this section discusses the results obtained from other resources.
Variance reduction techniques improve computational efficiency by using statistical methods to obtain more accurate performance measures, as in [60], [44], [43], [36], [47] and [48]. I have found that finding a good probability transform at various abstraction levels and time scales can be difficult. Even though these methods offer a considerable increase of simulation speed without requiring more processing capacity so far their applicability has only been shown for relatively simple examples and their extension for more realistic problems needs further research. For an overview of these and other special simulation techniques including hybrid and hierarchical simulation see [49] and [50].
Co-simulation techniques aim at loosely interconnecting two or more independently running simulators of different abstraction levels by allowing them to exchange messages. This approach, though attractive, often suffers from problems caused by timing and causability constraints [41]. The challenge of efficient communication between the various levels in multiple time scale simulations is addressed in e.g., [42], but the solution proposed there is not directly applicable to communication networks. Our approach is in fact a one directional co-simulation technique, also importing ideas from the hybrid approach. The main benefit of these changes is that the higher level simulator never needs to await results from the lower level counterpart. Instead, when needed, the higher level simulator uses predictions.

PERFORMANCE EVALUATION OF CELLULAR IP NETWORKS

The system consists of simple peer nodes that can be interconnected in an arbitrary topology to automatically form a cellular access network. In accordance with IP principles, Cellular IP takes a simplistic approach to location management, routing and handoffs. These properties make Cellular IP an ideal candidate to build simple, cheap cellular networks for the provision of ubiquitous wireless Internet access. In this chapter we provide a performance evaluation of Cellular IP networks based on a combination of analytical, simulation based and experimental techniques.

PROBLEM STATEMENT

We assumed that wireless access networks provide mobility and handoff support in “local" areas of various scales and character and that a global mobility protocol supports roaming between access networks. This vision motivated the design of Cellular IP, a wireless access technology that provides a cheap and flexible solution for wireless IP access networks ranging from small indoor systems to large area networks. In this environment a property of outmost interest is the solution's ability to adapt to a wide range of mobility and traffic conditions. Based on a combination of analytical, simulation and experimental studies we will analyze Cellular IP from four key aspects related to performance and adaptability. A fundamental design objective of Cellular IP was implementation and functional simplicity. To reduce complexity, we omitted explicit location registrations and replaced them by implicit in-band signaling. As a result, nodes of the access network need not be aware of the network topology or of the mobility of hosts in the service area. This design choice deliberately trades off performance for simplicity, potentially letting packets to be lost at handoff rather than explicitly buffering and redirecting packets as the mobile host moves.

- Route-update time;
- Paging-update time;
- Route-timeout;
- Paging-timeout; and
- Active-state-timeout.

MODEL

I limit our attention to the performance of a Cellular IP network in isolation. Interworking with global mobility protocols, particularly with Mobile IP is the subject of future phases of this work. We assume a Cellular IP system where cells partially overlap allowing the mobile host to immediately connect to a new base station after leaving the old one. In systems where this is not the case, the time it takes for the mobile host to move from the old cell to the new one adds to the handoff delay calculated and measured in this chapter. In order to focus on the networking aspects of Cellular IP, we ignore possible errors over the wireless link. Radio errors are not modeled in the simulator and are not generated in the test-bed. (Due to the proximity of the base stations to mobile hosts in the test-bed, transmission errors are rare.) In accordance with real systems, the wireless link represents, however, a throughput bottleneck in our experiments. This phenomenon is particularly interesting when the performance of TCP during handoffs is studied. The analysis is focused but not limited to best effort environments. Our examples are taken from basic IP applications, but packet loss is measured and the impact of handoff delays on performance is also analyzed. Though the current version of the protocol is optimized for best effort services, we believe that this analysis can serve as a springboard for augmenting Cellular IP with service quality provisioning techniques.

METHODOLOGY

The analysis of complex systems calls for a combination of analytical, simulation and experimental methods. Since an analytical approach gives the deepest insight into a system's behavior, in this chapter we use analytical tools whenever a meaningful and still tractable model can be established. Analytical methods prove to be particularly powerful in the analysis of mobility management cost. Issues beyond the limits of analytical methods are studied using a combination of measurement and simulation techniques. I use measurements in most cases where the Cellular IP prototype system allows for a meaningful experiment. The advantage of measurement techniques compared to both analytical and simulation methods are the lack of a modeling phase that inevitably results in the loss of some details. However, due to the limited size of the currently available Cellular IP test-bed, some aspects of protocol performance can not be measured. In addition, measurements are time consuming and inefficient in cases where major system configuration parameters need to be varied. In these cases we rely on simulations. In the following two sections we describe the simulation environment and the experimental setting used for the analysis.

SIMULATION ENVIRONMENT

The Cellular IP simulator is an extension of the ns-2 network simulator. Ns-2 is a public domain simulator written in C++ and Tcl programming languages and is widely used to analyze IP networks, in particular the TCP protocol. The Cellular IP extension is written in Tcl language. The simulator supports Cellular IP networks of arbitrary topology but the model contains a few limitations compared to real systems. The most important limitations are the following:

An “ideal wireless interface" is used. Packets transmitted over the wireless interface encounter no delay, bit error or loss. Congestion over the air interface can not be modeled.

The beacon messages transmitted by a Cellular IP gateway are not modeled. The network is configured when the simulation session is initiated and the topology remains constant during simulation.

Wireless cells are assumed to overlap and mobile hosts move from one cell to another in zero time. (We point out that this does not limit the simulator's ability of studying packet loss at handoff because that is caused by misrouted packets rather than by the mobile host's being hidden during handoff.)

EXPERIMENTAL SETTING

The Cellular IP network consists of three nodes, all multihued 300 MHz Pentium PCs. All three nodes implement the Cellular IP protocol as defined in [D1]. The protocol software is written in C programming language and resides in user space. One of the nodes also serves as gateway router. This node also implements the Cellular IP node software. In addition, it relies on the operating systems built in IP routing function to interface a regular IP network. The gateway is connected to a 100 Mbps Ethernet Local Area Network (LAN). The gateway node is connected to two other nodes through 100 Mbps full duplex links. Nodes are equipped with Wave-LAN 2.4 GHz radio interfaces. These devices implement the IEEE 802.11 protocol that is optimized for wireless packet data services, particularly wireless LANs [25]. Wave-LAN radio interfaces appear to the operating system as regular network interfaces (e.g., Ethernet), which makes it attractive for LANs and for Cellular IP. The mobile host (MH) is a 300 MHz Pentium PC notebook. The Cellular IP mobile host functions, including the state machine are implemented as a daemon running in user space. The mobile host is also equipped with a Wave-LAN 2.4 GHz radio interface. Unlike the nodes that operate at statically assigned frequencies, the mobile host can dynamically select one out of the eight frequencies supported by this product. At any time, the mobile host is tuned to be able to communicate with exactly one of the three nodes. The Wave-LAN radio devices operate over distances up to 50 meters. The test-bed's nodes are close to one another and throughout the experiments the mobile host is in the overlapping region of the three cells. This gives us full control over handoffs. We extended the mobile host's implementation with a utility that can periodically trigger handoffs regardless of the signal strength. A handoff initiated by this utility is identical to a handoff triggered by signal strength measurements.

RESULT ANALYSIS

HANDOFF PERFORMANCE

Handoffs are central to the performance of a cellular access network especially in systems with small wireless cells and fast moving hosts. Cellular IP is designed to operate efficiently even at very high handoff frequencies. In accordance with the design goals, a lightweight handoff algorithm is used that avoids explicit signaling messages (used for example in cellular telephony systems and in Mobile IP) and buffering or forwarding of packets (proposed in [21] and [24]). By this design decision, however, performance is traded for simplicity. Explicit registrations, and packet buffering or forwarding reduce or eliminate the disturbance handoff means to active data sessions. In Cellular IP, packets can be lost at handoffs and these losses must be dealt with at higher protocol layers (e.g., TCP). In this section we analyze the performance of Cellular IP handoff to determine the performance penalty we pay for a simpler implementation and operation.

HANDOFF DELAY

The disturbance that handoff means to ongoing sessions is commonly characterized by the handoff delay. Handoff delay is usually defined as the time it takes to resume normal traffic flow after the host performs a handoff. Though this does not fully determine the performance seen by the applications, it is a good indication of handoff quality. In [19] handoff delay is further decomposed to rendezvous time and protocol time. Rendezvous time refers to the time it takes for a mobile host to attach to a new base station after it leaves the old one. This time is related to wireless link characteristics, particularly to the inter-arrival time of beacons transmitted by base stations. Protocol time refers to the time it takes for the traffic flow to be restored once the mobile host has received the beacon from the new base station. In the present analysis we assume that the rendezvous time is small and handoff performance is determined by the protocol time. Instead of adopting the notations proposed in [19], we therefore define handoff delay as the time it takes for a mobile host to receive the first packet through the new base station after it moved from the old to the new base station | which we assume to take zero time. In Cellular IP, handoff delay and packet loss are consequences of the time it takes for the distributed routing state to follow host mobility. Immediately after handoff, mobile hosts transmit a route-update packet to reduce this time to a minimum. The route-update packet travels from the new base station to the gateway and configures a new downlink route to the mobile host. The old and new downlink routes both originate in the gateway but while the former routes packets to the old base station, the latter leads to the base station the host has just moved to. The node where the old and new routes join is referred to as the cross-over node. The new downlink route becomes operational when the first route-update packet transmitted through the new base station reaches the cross-over node. The time period while the mobile host is not receiving packets after handoff is therefore the time it takes for the route-update packet to reach the cross-over node plus the time it takes for the first downlink packet to travel from the cross-over node to the base station. Handoff delay, as defined previously, is therefore equal to the round-trip time between the new base station and the cross-over node.

PACKET LOSS AT HANDOFF

Application level quality, however, is more related to the number of packets lost at handoff than to the handoff delay. To determine handoff packet loss, let us assume that a periodic stream of packets is being transmitted from the Internet to a mobile host. Before handoff, the packets are routed along the old route. In the following calculation, we will assume that the cross-over node knows in advance which of the stream of packets will be the last one to reach the mobile host at its old location. Let us assume that the cross-over node marks this packet. Upon receiving the marked packet, the mobile host performs a handoff and immediately transmits a route-update packet through the new base station. Downlink packets routed by the cross-over node after the marked packet but before the arrival of the route-update packet are routed to the old base station and get lost. This time interval is equal to the sum of the time it takes for the marked packet to propagate from the cross-over node to the mobile terminal and the time it takes for the route-update packet to reach the cross-over node. The loss of packets at handoff is therefore related to the “handoff loop time" defined as the transmission time from the cross-over node to the mobile host's old location plus the transmission time from the mobile host's new location to the cross-over node. Specifically, the number of lost packets at handoff loss is equal to the number of packets arriving to the cross-over node during the handoff loop time TL, that is

$$\mathbf{n_{loss} = {}_{w}T_{L}}$$

Where w is the rate of downlink packets. Since the average handoff loop time is equal to the average handoff delay, the expected number of packets lost at handoff can equally be calculated using the handoff delay (which is easier to measure) and in what follows we do not differentiate between these two values. During these measurements the mobile host received 100 byte UDP packets at rates of 25 and 50 packets per second (pps) while performing handoffs every 5 seconds. Each point on the graph was obtained by averaging loss measurements over 50 consecutive handoffs. To vary the round-trip time between the mobile host and the cross-over node, I emulated an increasing load which results in increasing buffering of downlink packets. Under these experimental conditions hard handoff results in at least 1packet loss for small mobile to gateway round-trip delays and up to 4 packet losses for delays of 80 ms. This result is comparable to handoff packet loss results reported about the Caceres protocol (3.7 packets lost per handoff) and about the multicasting approach (2 to 4 lost packets if buffering is not used). This comparison does not reveal, however, the fundamental differences between Cellular IP handoff and the handoff schemes proposed in the cited approaches. In Cellular IP handoff does not differ from normal operation. Neither mobile hosts nor base stations have special states associated with handoff. In exchange for this simplicity, however, handoff performance is dependent upon the traffic conditions. In a highly loaded network the handoff delay will be higher and more packets will get lost. Real time Internet applications, for example voice over IP, are sensitive to packet delay and can not retransmit lost packets. For these applications, the number of lost packets fully characterizes handoff performance. Other applications, however, use end-to-end flow control to respond to network and traffic conditions and retransmit packets and/or reduce transmission rate if errors occur. In what follows, we will focus on TCP performance in the presence of handoffs. TCP is selected because it represents the most typical traffic type over today's Internet which carries World Wide Web, file transfer, remote login and other applications. Investigating TCP performance is also important because its flow control has been shown to operate sub-optimally in a wireless environment.

TCP BEHAVIOR AT HANDOFF

I will first use simulation to look at the behavior of a TCP session at handoff. The TCP packet size is 1000 bytes and a mobile user has up to 5 Mbps downlink bandwidth, that is, the downlink packet rate w is 625/sec. Packet transmission time between nodes in the simulated configuration is 2 ms, resulting in a handoff delay of 4 ms. After the handoff delay packets continue to arrive at the mobile host. These packets, however, are out of sequence and cause the receiver to generate duplicate acknowledgments as indicated by the horizontal line of acknowledgment sequence numbers. The duplicate acknowledgments inform the TCP transmitter about the losses and cause it to retransmit the lost packets. The first retransmitted packet arrives approximately 20 ms after the handoff. Using Tahoe flow control, the transmitter remains silent until this packet is acknowledged and increases its transmission window size as further acknowledgments arrive. (A detailed description of TCP flow control is presented in [78].) Full speed is not regained until approximately 4.07 sec simulated time. We conclude that a Cellular IP handoff is interpreted by a transmitter in the wired IP network as congestion and causes it to reduce transmission rate. Using Tahoe flow control the handoff triggers a slow start which increases the performance impact of handoff packet loss. In the studied circumstances, normal operation is resumed approximately 70 ms after handoff. I will show the impact of this disturbance on TCP throughput in a series of experiments later in this section. In the next simulation session TCP is used to carry data from the mobile host. In this case handoff packet loss aspects acknowledgments instead of data packets. Before handoff the TCP sender (in the mobile host) uses its maximum window size of 20 which is reflected in the difference between data packet and acknowledgment sequence numbers. At 4 sec simulated time the mobile host performs a handoff and stops receiving acknowledgments for a period of approximately 4 ms, which is the handoff delay. During the handoff delay the sender does not transmit any packets since its window size is used up and it needs incoming acknowledgments to advance its transmission window. After the handoff delay, these acknowledgments are routed to the mobile host's new location. Due to the cumulative nature of TCP acknowledgments, the first acknowledgment arriving to the mobile host after handoff informs the sender that all its transmitted packets have arrived to the receiver (up to the sequence number shown in the acknowledgment). This causes it to advance its transmission window and continue transmitting at the maximum available data rate. In the simulation example this rate is slightly higher than the rate dictated by TCP flow control (which is the long term average capacity) so the curve of data packet sequence numbers is somewhat steeper after handoff delay than outside the handoff area. Normal operation is resumed quickly and handoff represents little disturbance to the data session. The behavior is different, however, if handoff occurs when the TCP session is in its initial, slow start phase and acknowledgments are not regularly arriving to the mobile host. In this case the new downlink route is established after the handoff delay but no acknowledgments arrive to the sender. If at this point the sender has used its entire transmission window and is waiting for acknowledgments then TCP can suffer a delay equal to the sender's retransmission timer. Mechanisms to avoid this problem are for further study.

TCP THROUGHPUT

Next, we study the impact of handoff performance on TCP Reno throughput in the experimental test-bed. The mobile host performs handoffs between B2 and B3 at fixed time intervals. I measure TCP throughput using TCP by downloading 16 M Bytes of data from the correspondent host to the mobile host. The throughput measured at zero handoff frequency (i.e., no handoffs) is lower that the 1.6 Mbps we could achieve using standard IP routing in the same configuration. This difference between IP and Cellular IP forwarding is attributed to the fact that IP is implemented in the kernel and Cellular IP in user space. In addition, Cellular IP uses PCAP to forward packets which are optimized for monitoring rather than IP forwarding. We observe that the performance of TCP degrades as the handoff frequency increases due to packet loss. The shape of the throughput curve changes around 5 handoffs per minute. This is attributed to the fact that as the handoff rate increases TCP has less time to recover from losses and at this point it starts operating continuously below its optimal point. Further increasing the handoff frequency results in a significant drop in performance approaching 550 kbps as the handoff rate moves toward one per second.

SCALABILITY

The design of Cellular IP was motivated by the vision of ubiquitous wireless Internet access where the same protocol may be used in small indoor systems up to metropolitan area wireless ISPs. This role can only be fulfilled by a protocol that adapts to a wide range of network sizes and shapes. In this section we study Cellular IP scalability through the scalability of our experimental implementation. Nodes are the universal building blocks of a Cellular IP network determining its performance and scalability. We will therefore start by looking at our node implementation and determining the performance limits of a Cellular IP node using off-the-shelf hardware. This is important because Cellular IP design promotes the use of commodity hardware to support network elements such as the node. Based on the performance measured in the node implementation we will next study three mobile networking scenarios that provide insight into the ability of Cellular IP to be customized to a wide range of mobility, application and network conditions.

NODE PERFORMANCE

In the first experiment we investigate the performance of our node implementation using a multihued 300 MHz Pentium PC. The fact that the throughput curve is hardly decreasing with increasing routing cache size suggests that in the scenarios that we have studied the performance bottleneck was not the cache lookup time. To verify this assumption we also measured the throughput that a single traffic stream achieved using standard IP forwarding through the same multihued PC. The Cellular IP node throughput is somewhat below the standard IP throughput due to the additional packet processing involved with PCAP and additional packet copies across kernel and user space domains. In our implementation, the routing cache is stored in a binary tree to achieve fast lookups. We also measured the performance of the search algorithm and found that the maximum packet rate can be well approximated as

$$\frac{2,500,000}{1 + 2\log(m)}$$

Where m is the number of mappings in the cache. This explains that the performance bottleneck was found to be the network interface throughput rather than the search time over the range measured. For significantly larger user populations the cache lookup would likely become a bottleneck too. I did not, however, verify this thesis due to memory size constraints and unavailability of network interfaces that operated in excess of 100 Mbps. To illustrate this phenomenon we also experimented, however, with linear search instead of the binary tree.

SERVICE QUALITY PROVISIONING

Cellular IP deliberately trades handoff performance in exchange for implementation simplicity motivated by the desire to provide a cheap and robust solution primarily for best effort service. I studied handoff performance and showed the performance penalty of handoff simplicity. As a final step of my evaluation methodology I will now depart from the original design decision of minimum complexity and investigate Cellular IP's capability to extend toward the support of services beyond best effort. Cellular IP can be considered as a modification of IP networking where routing state is dynamically modified in response to user mobility. The packet forwarding unit present in Cellular IP nodes is fundamentally identical to an IP router's packet forwarding engine. This similarity suggests that recent efforts to extend regular IP networks with service quality provisioning schemes (e.g., the differentiated service concept [27]) will likely be adaptable to a Cellular IP network. The added difficulty of supporting service quality in Cellular IP stems from the handoffs. In the remainder of this section we will therefore focus on improving handoff performance.

ADVANCE BINDING

The Cellular IP base protocol's handoff algorithm is founded on a simplistic approach that allows some packet loss in exchange for minimizing handoff messaging instead of trying to guarantee zero loss. In Cellular IP the routing information associated with a mobile host's old location is not cleared at handoff, rather, it times out as the associated timers expire. Before the timeout a period exists when both the old and new downlink routes are valid and packets are delivered through both base stations. This feature is used in the advance binding procedure that significantly improves handoff performance but still fits in the lightweight nature of the base protocol. Advance binding provides a probabilistic improvement instead of fully eliminating packet loss. An important feature of the advance binding handoff is that it improves handoff performance without any modification in Cellular IP nodes. The necessary changes are limited to the mobile host state machine where a single (temporary) state must be added to the base protocol's state machine. Nodes remain fundamentally stateless and unaware of handoff. The purpose of the advance binding algorithm is to reduce handoff latency. To this avail, the routing cache mappings associated with the new base station must be created before the actual handoff takes place. When the mobile host decides handoff, it sends an advance binding packet, which is technically a route-update packet, to the new base station but immediately returns to listening to the old base station. While the host is still in connection with the old base station, the advance binding packet configures routing cache mappings associated with the new base station. After an advance binding delay (Ta), the host can perform a regular handoff. In the case of advance binding handoff, the handoff delay period is started at the time when the mobile hosts transmit the advance binding packet. If the mobile host performs the actual handoff later than the expiration of the handoff delay, the routing state associated with the new base station will have been established and packets will continue to arrive to the mobile host immediately after handoff. The advance binding handoff is attractive because of its simplicity. It requires only minor modification in the mobile host state machine and no modification at all in the nodes. That advance binding does not always eliminate handoff packet loss. Investigating the difference between simulation and experimental results we have found that advance binding fails to provide seamless handoff if transmission delays from the cross-over node to the old and new base stations are very different. In such cases the data session is disturbed at handoff despite the elimination of handoff delay because the packet streams transmitted through the two base stations will not be synchronized. The mobile host continues to receive packets immediately after handoff, but does not necessarily receive them in the correct order. If the new base station is \behind" the old one, the mobile host will receive duplicate packets at handoff. IP based applications tolerate sporadic duplicate packets but can be disturbed by multiple duplicates of the same packet. Multiple duplicates can be generated if multiple handoffs occur within a short time period. To avoid such handoff ping-ponging, signal strength based handoff control can involve hysteresis. If, however, the new base station is \ahead" compared to the old one, packet loss can occur at handoff. These losses can not be compensated for because Cellular IP base stations do not buffer packets. The decline of TCP throughput with increasing handoff frequency. I conclude that advance binding decreases handoff packet loss in exchange for very little added complexity but do not eliminate it, especially when handoff occurs between base stations that operate in different traffic conditions (resulting in different delays).

SEMISOFT HANDOFF

While advance binding ensures that the mobile host continues to receive packets immediately after handoff, experimental results. It alone does not always provide smooth handoff. This observation motivates the design of Cellular IP semisoft handoff. Semisoft handoff has two components. First, similar to advance binding handoff, it involves establishing the downlink route toward the new base station before handoff actually takes place. Unlike in advance binding handoff, however, semisoft handoff uses a special packet called semisoft packet for this purpose. A semisoft packet is transmitted through the new base station before handoff and it propagates through the network to create Routing Cache mappings in nodes on the way. The second component of the semisoft handoff procedure is based on the observation that perfect synchronization of the data streams through the old and new base stations is needless. The condition resulting in packet loss at advance binding handoffs can be eliminated by temporarily introducing into the new path a constant delay sufficient to compensate for the time difference between the two streams. The delay will ensure that the new base station is behind, rather than ahead of the old one. This will result in duplicate packets instead of packet loss. Converting packet loss into duplicate packets is advantageous because these duplicate packets can be eliminated at the mobile host which is naturally aware of a semisoft handoff being in progress. Unlike advance binding handoff, semisoft handoff requires support of Cellular IP nodes in introducing the said temporary delay. Semisoft handoff does still not require, however, any interaction between nodes, base stations and the mobile host. The delay is introduced in the new data stream by the cross-over node. This node knows that a semisoft handoff is in progress from the fact that a semisoft packet arrives from a mobile host that has mapping to another interface. The mapping created by the semisoft packet in the cross-over node has a flag to indicate that downlink packets routed by this mapping must pass a delay element before transmission. I recall that similar to other mappings, this information is stored as soft state and is cleared after the route-timeout. In normal conditions it is actually cleared before the expiration of the soft-state timer because mobile hosts send a route-update packet immediately after handoff. This packet will update the mapping created by the semisoft packet and clear the flag explicitly. Clearing the flag causes all packets eventually stored in the delay device to be forwarded to the mobile host. We point out that base stations only need a small pool of delay buffers since very few mobile hosts will simultaneously be in semisoft position. The experimental conditions for hard and semisoft handoff were identical. The mobile host received 100 byte UDP packets at rates of 25 and 50 packets per second (pps). Each point on the graph was obtained by averaging loss measurements over 50 handoffs. In these experiments, the new downlink packet stream at semisoft handoffs was delayed in the cross-over node by a buffer holding each packet until the arrival of the next downlink packet. When the semisoft handoff is completed, the last packet is cleared from the buffer and is sent to the mobile host. Note that buffering a single packet in the delay element was sufficient to eliminate loss even in the case of a large round-trip time when hard handoff resulted in the loss of up to four packets. This is because the semisoft buffering is only to compensate for the difference between the transmission times along the old and new paths and not for the entire round-trip time between the mobile and the cross-over point. The next experiment investigated the improvement in TCP throughput gained using semisoft handoff. The experimental conditions for the semisoft, advance binding and hard handoff measurements were identical. The semisoft delay element in this case was an 8-packet circular buffer. I observe that using semisoft handoff, packet loss is entirely eliminated and a slight disturbance only remains due to the transmission delay variations encountered at handoff. We point out that even for one handoff per second; the throughput is almost identical to that observed for a static host.

CONCLUSIONS

In this dissertation I have addressed some of the challenges that existing and future cellular mobile systems will meet in an environment of ubiquitous wireless Internet availability. I have argued that the development of small palmtop computers and of web based services demands the development of efficient and flexible cellular wireless access technologies. These technologies will build on cellular telephony technology on one hand and will incorporate IP concepts on the other hand. Cellular access networks in this new environment will face new challenges in terms of network throughput requirements and service quality constraints. In order to plan and manage cellular networks that serve a variety of applications providing the expected service to each of them, new performance analysis methods are required. The increasing user population and data rates call for increasing cellular system throughputs. Cellular network operators can increase system throughput by decreasing the cell size which, however, results in increased handoff frequency. To avoid the frequent dropping of connections at handoff, the operator must then reserve an increasing amount of capacity in wireless cells for future handoff attempts. I have analyzed the efficiency of cellular systems where resources are reserved for future handoff attempts in a deterministic way, and have calculated the decrease of efficiency in response to increasing user speed, decreasing cell size and increasing application heterogeneity. We have studied systems with statistical resource reservation policies and have calculated the upper bound of system efficiency as a function of the tolerated handoff blocking probability, regardless of what admission control and reservation policy are used. These results are also reported in [C1] and [J4]. Third generation cellular mobile systems and Internet host mobility proposals both address the issue of wireless IP host mobility, but do it from two different directions. I have established requirements for a cellular access networking solution and have derived network design principles. Based on these principles, we have designed Cellular IP, a new architecture and protocol that allows for providing Internet access to wireless mobile users in a simple, flexible and scalable manner. Cellular IP is described in details in [J2], in [C2] and in [D1].

Finally, in Chapter 4, I have analyzed Cellular IP from the aspects of handoff quality, mobility management cost, scalability and service quality provisioning. For the analysis, we have used a combination of analytical, simulation and experimental methods. I have found that Cellular IP performs well even in environments of very high mobility and have quantified the performance penalty of the simplistic approach taken in Cellular IP design. We have evaluated the impact of network planning decisions such as protocol parameter settings and paging cache distribution and have derived suggestions on setting these in an optimal way. We have also identified a few shortcomings of the Cellular IP protocol, primarily in the area of service quality provisioning.

ACKNOWLEDGMENTS

The authors would like to thank anonymous reviewers for their valuable comments and suggestions that improve the presentation of this paper.

REFERENCES

[1] K.S. Shanmugan, P. Balaban, \A modi_ed Monte Carlo simulation technique for the evaluation of error rate in digital communication systems,\ *IEEE Trans. Commun.*, Vol. COM-28, 1980.

[2] A.M. Law, W.D. Kelton, \Simulation, Modeling and Analysis," *McGraw-Hill*, New York, 1991.

[3] M. Vill_en-Altamirano, J. Vill_en-Altamirano, \RESTART: A Method for Accelerating Rare Event Simulation," in *Proc. 13th ITC, Queueing, Performance and Control in ATM*, pp.71-76, 1991.

[4] G.S. Fishman, \Accelerated Accuracy in the Simulation of Markov Chains," *Operat. Res.*, Vol. 31., 1983.

[5] R.Y. Rubinstein, R. Marcus, \E_ciency of Multivariate Control Variates in Monte Carlo Simulation," *Operat. Res.*, Vol. 33, 1985.

[6] S.S. Lavenberg, T.L. Moeller, P.D. Welch, \Statistical Results on Control Variables with Application to Queueing Network Simulation," *Operat. Res.*, Vol. 30, 1982.

[7] V.S. Frost, W.W. Larue, K.S. Shanmugam, \E_cient Techniques for the Simulation of Computer Communication Networks," *IEEE J. Select. Areas Commun.*, Vol. 6, No. 1, pp. 146-157, 1988.

[8] Performance evaluation and design of multiservice networks, *COST 224*, 1992.
[9] M. Coppola, \A Cosimulation Environment with Opnet and VHDL," *13th UK Workshop on Performance Engineering of Computer and Telecom. Systems (UKPEW '97)*, Ilkley, UK, 1997.
[10] K. Hines, G. Borriello, \Dynamic Communication Models in Embedded System CoSimulation," *34th ACM Design Automation Conference*, Anaheim, CA, 1997.
[11] G. Brasche, B. Walke, \Concepts, Services and Protocols of the New GSM Phase 2, General Packet Radio Service," *IEEE Commun. Mag.*, pp. 94-104, August 1997.
[12] Charles Perkins, editor, \IP Mobility Support," *Internet RFC* 2002, October 1996.
[13] Pravin Bhagwat, Charles Perkins, Satish Tripathi, \Network Layer Mobility: an Architecture and Survey," *IEEE Personal Commun.*, Vol. 3, No. 3, pp. 54-64, June 1996.
[14] Brian G. Marchent, \Third Generation Mobile Systems for the Support of Mobile Multimedia based on ATM Transport," tutorial presentation, *5th IFIP Workshop*, Ilkley, July, 1997.

Sumit kumar srivastava is a M. Tech. student of Computer Science & Engineering at Teerthanker Mahaveer University, Moradabad. Presently he is working as Assistant professor. at Dev Bhoomi institution of Engineering & Technology Dehradun, Uttarakhand

Dharmendra K. Gupta is a M. Tech. student of Computer Science & Engineering at Teerthanker Mahaveer University, Moradabad. He Received a M.C.A. in Computer Application from Punjab Technical University, Jalandhar in 2009. Presently he is working as lecturer at Institute of Engineering & Technology, Alwar since august – 2010. He has research interests are in the areas of data & network security and programming language.