INTERNET-DRAFT A. T. Cambell, S. Kim, J. Gomez, C-Y. Wan
Columbia University
Z. Turanyi, A. Valko
Ericsson
October 1999
<draft-gomez-cellularip-perf-00.txt>

Expires May 2000

Cellular IP Performance

Status of this Memo

Abstract

Cellular IP specifies a protocol that allows routing IP datagrams to a mobile host. The protocol is intended to provide local mobility and handoff support. It can interwork with Mobile IP [1] to provide wide area mobility support. This I-D provides performance results based on the implementation and analysis of Cellular IP as defined by draft draft-valko-cellularip-01.txt [2]. The evaluation is focused on handoff performance, optimum active-state-timeout choice and scalability issues. The Cellular IP source code used for the experimental results presented in this Internet-Draft is freely available from comet.columbia.edu/cellularip.

Table of Contents

1. Introduction

Cellular IP inherits cellular technology principles for mobility management, passive connectivity and handoff support, but implements these around the IP paradigm. The universal component of a Cellular IP network is the Cellular IP node that serves as wireless access point but at the same time routes IP packets and integrates cellular control functionality traditionally found in Mobile Switching Centers (MSC) and Base Station Controllers (BSC). The nodes are built on regular IP forwarding engine, but IP routing is replaced by Cellular IP routing and location management. The Cellular IP network is connected to the Internet via a gateway router. Mobile hosts attached to the network use the IP address of the gateway as their Mobile IP care-of address.

In Cellular IP, location management and handoff support are integrated with routing. To minimize control messaging, regular data packets transmitted by mobile hosts are used to establish host location information. Uplink packets are routed from a mobile host to a gateway on a hop-by-hop basis. The path taken by these packets is cached in intermediate nodes. To route downlink packets addressed to a mobile host the path used by recent packets transmitted by a mobile host is reversed. When the mobile host has no data to send then it transmits ICMP IP packets called route-updates to the gateway to maintain its downlink routing soft-state. Following the principle of passive connectivity commonly found in cellular systems mobile hosts that have not received packets for some period of time allow their downlink routes be cleared from the caches as dictated by soft-state timers. In order to route packets to idle hosts a Cellular IP mechanism called paging is used.

The source code for Cellular IP v1.0 [4] developed at Columbia

University is freely available (comet.columbia.edu/cellularip) for experimentation. This Internet-Draft provides an overview of the source code design and implementation. In addition, the experimental results presented in this memo are based on the public domain source code operating on a Cellular IP testbed built at Columbia University. For a full discussion of Cellular IP see [3] and for a full specification of the protocol see [2].

2. Implementation

Cellular IP is based on modular software design and is implemented on FreeBSD 3.2 software platforms in user space. Cellular IP is comprised two protocol modules: the node and mobile host modules. Both protocol modules rely on a common system module. The system module filters IP packets from the physical medium and moves them to user space and delivers packets processed in user space to the required network interface. The system module uses the Berkeley Packet Filter's Packet Capture library (PCAP). In what follows, we describe the node and mobile host protocol modules.

2.1. Node Module

As described previously, the Cellular IP node serves as wireless access point, router and location manager. In addition, our implementation allows a node to also implement the gateway functionality relying on the kernel's IP routing function. In our implementation, the uplink and downlink neighbors are configured by network management. This is in contrast to the gateway beacon approach discussed in [2]. Along with the routing and (optional) paging cache, the most important functions of the node include:

- a paging update function, which maintains the paging cache by updating it for each uplink packet and by clearing expired mappings;

- a classifier, which parses uplink packets and selects those that should update the routing cache (data, route-update and semisoft packets);

- a route update function, which maintains the routing cache by updating it for each packet selected by the classifier and clears expired mappings;

- a routing cache lookup function, which parses downlink packets and searches the routing cache for mapping(s) associated with the destination mobile host;

- a paging cache lookup function, which searches the paging cache for mappings if a routing cache mapping was not found;

- a forwarding engine, which forwards downlink packets to the interface selected by routing cache lookup in the first instance and by paging cache lookup if no route was found; and

- a delay device, which is temporarily inserted in the downlink route if a semisoft handoff is in progress.

Our implementation uses 2 Mbps 2.4 GHz WaveLAN radio devices, but the protocol module can transparently interwork with other radio interfaces. Note that the source code also works with a variety of higher speed radios (e.g., Aironet's 11 Mbps and WaveLAN IEEE 802.11

10/6 Mbps cards). A Cellular IP node may have multiple air interfaces or no air interface at all if it serves as a concentrator node only. In addition to the functions described above, the node contains a beacon generator for each wireless interface. In Cellular IP, the content of the beacon is similar to the standard WaveLAN beacon but is extended with the gateway's IP address.

2.2. Mobile host module

The mobile host module is implemented as a deamon that in our experimental testbed runs in user space. The standard IP protocol stack is not touched by the Cellular IP deamon and applications are unaware of mobility. The main elements of the deamon are as follows:

- a handoff controller, which keeps statistics of measured beacon strengths and decides and performs handoff. Handoff mainly involves the setting of radio frequency and changing the IP default route to a new node's address. In the case of the semi-soft handoff mode, a semi-soft packet is first transmitted to the new base station has part of the mobile initiated handoff procedure and after a constant time delay the handoff is completed to the new base station.

- a protocol state machine, which has two states: active and idle state. In idle state, any incoming packet triggers a transition to active state. At the same time, a timer is initiated that is reset by each incoming packet. The expiration of the timer triggers the transition to idle state.

- a control packet generator, which periodically transmits route-update or paging-update packets as required by the state machine. The packet generator also monitors outgoing packet to stop transmitting control packets when data is being transmitted.

3. Evaluation

To evaluate Cellular IP performance we have built a testbed and designed a set of experiments to analyze the Cellular IP algorithms discussed in [2]. In what follows we describe our Cellular IP testbed configuration and experimental results.

3.1 The Experiments

The goal of the experiments is to evaluate the performance and scalability of the Cellular IP protocol. An important object of the first experiment is to analyze the performance of hard and semi-soft handoffs and to investigate the impact of handoff on TCP performance. The second experiment investigates the cost of setting the active-state-timeout. In the third and last experiment we investigate the scalability limits of a node based on off the shelf multi-homed PC hardware. This experiment provides some insight into what would be achievable for a Cellular IP node built from commodity hardware, operating systems and air-interfaces. The experimental results of the Cellular IP node implementation are therefore conditioned more by the technology used than the protocol itself.

```
                                                |
                   _______                   ___|___
                  |       |                 |       |
                  | Host  |                 | Router|
                  |_______|      LAN        |_______|
       _________|_______________________________|________
                               ___|___
                              |       | Gateway
                              |  B1   |
                              |_______|
                     _________|     |_________
                  ___|___                ___|___
                 |       |              |       |
                 |  B2   |              |  B3   |
                 |_______|              |_______|


                               ____
                              |    |
                              | MH |
                              |____|
```

Figure 1. A Cellular IP Testbed

Figure 1. Cellular IP Testbed
URL:http://comet.columbia.edu/cellularip/publications/draft-gomez-cellularip-perf-00.pdf

All of the experiments described below were conducted using the configuration of shown in Figure 1. The Cellular IP network consists of three nodes, all multi-homed 300 MHz Pentium PCs. One of the nodes also serves as gateway router. 100 Mbps full duplex links interconnects Cellular IP nodes and the gateway is connected to a 100 Mbps Ethernet LAN. In the experiments, the correspondent host is a Sun SPARCstation 5 with Solaris operating system. The mobile host is a 300 MHz Pentium PC notebook. The mobile host and the nodes are equipped with WaveLAN 2.4 GHz radio interfaces. These devices can operate at eight different frequencies to avoid interference between adjacent cells. In the testbed the nodes stations are statically assigned frequencies while the mobile can dynamically change frequency to perform a handoff. Throughout the experiments the mobile host is in the overlapping region of the cells which gave us full control over handoffs. We have extended the mobile host's implementation with a utility that can periodically trigger handoffs regardless of the signal strength observed at the mobile host under experiment. A handoff initiated by this utility tool is identical to a handoff triggered by signal strength measurements at the mobile host.

3.2. Handoff Performance

In this experiment we measured packet loss for hard and semi-soft handoff. During these measurements the mobile host receives 100 byte

UDP packets at rates of 25 and 50 packets per second (pps) while continually making handoffs between nodes B2 and B3 every 5 seconds. The measurement results are shown in Table 1. Each point in the table was obtained by averaging loss measurements over 50 consecutive handoffs. To vary the round-trip time between the mobile host and B1, we emulate an increasing load that results in an increased buffering of downlink packets. Table 1 shows that hard handoff causes packet losses proportional to the round-trip time and to the downlink packet rate. Under these experimental conditions hard handoff results in at least 1 packet loss for small mobile to gateway round-trip delays and up to 4 packet losses for delays of 80 ms.

Packet Loss per Handoff

		Hard(25 pps)	Hard(50 pps)	Semi-soft(25 and 50 pps)
Mobile-GW	3	0.2	0.68	0
roud-trip	43	1.22	2.64	0
time [ms]	83	2.21	4.50	0

Table 1. Downlink Packet Loss during Handoff

Figure 2. Downlink Packet Loss during Handoff
URL:http://comet.columbia.edu/cellularip/publications/draft-gomez-cellularip-perf-00.pdf

Table 1 also shows packet loss that results from using Cellular IP semi-soft handoff [2]. The experimental conditions for semi-soft and hard handoff were identical. In these experiments, the new downlink packet stream was delayed by one buffer holding for each packet until the arrival of the next downlink packet. See [3] for full details. When the semi-soft handoff is complete, the last packet is cleared from the delay device [3] buffer and is forwarded to the mobile host.

Table 1 illustrates that semi-soft handoff eliminated packet loss entirely. Note that buffering a single packet in the delay element is 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 semi-soft buffering is only used 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.

In the next experiment, we studied the impact of handoff performance on TCP Reno throughput. The mobile host performed continuous handoff between B2 and B3 at fixed time intervals. We measured TCP throughput using ttcp by downloading 16 MBytes of data from the correspondent host to the mobile host. Each data point represents an average of 6 independent measurements. TCP throughput to a mobile host performing hard handoff is shown in Table 2. The throughput measured at zero handoff frequency (i.e., under no handoff conditions) is slightly less than the 1.6 Mbps achieved using standard IP forwarding 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 is not optimized for IP forwarding. We observe that the performance of TCP degrades as the handoff frequency increases due to packet loss. As the handoff rate increases TCP has less time to recover from loss forcing it to operate below its optimal point and resulting in a significant drop in performance as the handoff rate moves toward one per second.

The next experiment investigated the improvement gained using the semi-soft handoff. The experimental condition for the semi-soft and hard handoff measurements were identical. Table 2 also shows TCP throughput to a mobile host that performs semi-soft handoffs at increasing rates. Semi-soft handoff reduced packet loss and significantly improved the throughput in relation to hard handoff.

Downlink TCP throughput [kbps]

		Hard	Semi-soft (1 buffer)	Semi-soft (8 buffer)
	0	1500	1510	1560
	2	1423	1426	1530
Number of	3	1193	1411	1520
handoffs	5	1120	1350	1500
per minute	10	1082	1345	1490
	20	966	1300	1480
	30	891	1232	1470
	60	519	1036	1430

Table 2. Throughput of TCP Donwload

Figure 3. Throughput of TCP Download
URL:http://comet.columbia.edu/cellularip/publications/draft-gomez-cellularip-perf-00.pdf

However, unlike in the UDP traffic experiment, packet loss is not entirely eliminated which is reflected in the decline of throughput at increasing handoff frequency. We attribute this to the fact that the delay we inserted in the downlink packet stream by buffering packets is tied to the packet inter-arrival time which is both shorter and more irregular in a TCP flow than in our UDP example. To introduce sufficient delay even if a few packets arrive back-to-back, we changed the semi-soft delay element to an 8-packet circular buffer. Table 2 shows the performance results using this delay element. Here, packet loss at handoff 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. This result looks very promising.

3.3. Active-State-Timeout

Now we analyze the tradeoff involved in the setting of the active-state-timeout. In Table 3, we show paging traffic rate measured in the testbed using a set of typical Internet applications. It must be noted that these values depend heavily on the user behavior and this data is presented for illustrative purposes only.

Our measurements involved 4 minute sessions where the route-update time is 100 ms and a recently created route has a validity time of 300 ms. In the telnet and WWW sessions the mobile host is the client. We collected data from two sets of measurements for two different active-state-timeout values. As discussed in [2], this parameter determines the time a mobile host maintains its routing cache mappings after receiving a packet. In other words, the active-state-timeout reflects the expectation that one downlink packet may with high probability be soon followed by another and that it is worth keeping up-to-date routing information for some time, despite the cost associated with transmitting route-update packets. Indeed, as Table 3 shows, taking a larger active-state-timeout decreases the paging traffic for all applications. The difference between applications is also intuitive. An interactive application (e.g., telnet) sends one or more packets to the server triggering some action on the server side. This in turn results in new packet(s) being sent back. In the case of a local server, paging only occurs when the time to process the request exceeds the active-state-timeout. This is rare, hence the low paging rate for local telnet sessions. The packet round-trip time adds to the server processing time for remote sessions. Table 3 shows that in some cases the total response time exceeded 1 sec in this example.

Rate of Paging Traffic to Mobile [bps]

		telnet local	telnet remote	WWW local	WWW remote	VIC to mobile	VIC from mobile
Active	100 ms	79	391	118	1507	800	39
state timeout	1 sec	2	94	47	438	61	3

Table 3. Paging Traffic Rate Generate by Some Applications

Figure 4. Paging Traffic Rate Generated by Some Applications
URL:http://comet.columbia.edu/cellularip/publications/draft-gomez-cellularip-perf-00.pdf

A similar difference can be observed between the local and remote WWW sessions. In this case paging occurs when the response time exceeded the active-state-timeout. We observe that this is rare when accessing a local server but frequent in the case of remote communications. In addition, when large amounts of data are downloaded TCP sometimes stalls which also resulted in paging being triggered for any pauses that exceeded the active-state-timeout. Our final example application is VIC. With the mobile host being the receiver of the video stream, we note that the state machine at the mobile host rarely moves to idle state since each incoming packet resets the active-state-timeout. Because of the relatively large data rate (500 kbps) in our experiment the paging rate is significant, however. We observed that when the mobile host transmits packet video then the downlink packets only carry quality of service information and paging rate is negligible.

3.4. Scalability

The main concern in terms of scalability is the use of per-host routes. We believe that there has to be per-host state in the access network to support high performance handoff. In Cellular IP scalability is achieved despite per host routes by separating the location management of idle from active mobile hosts. Data packets are normally routed by route cache mappings. The size of the route cache is determined by the number of mobile hosts in active state at any one time. This way the Cellular IP network can accommodate very large number of mobile hosts without having to search large data bases for each packet and This separation principle is fundamental to scalability as per-host state is to handoff performance in a Cellular IP network.

In following experiment we investigate the performance of our node implementation. Table 4 shows the node throughput for linear and binary-tree search algorithms measured on a multi-homed 300 MHz Pentium PC using ttcp and 1500 byte packets. In these measurements we substituted a 100 Mbps Ethernet connection for the radio interface and created routing cache mappings for random IP addresses to emulate a large active user population. The fact that the throughput curve is hardly decreasing with increasing routing cache size (binary-tree search) suggests that in the studied scenarios 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 multi-homed PC. As shown in Table 4, 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

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

where n is the number of mappings in the cache. This explains that the performance bottleneck was found to be the network interface throughput rather then the search time over the range measured. For significantly larger user populations the cache lookup would likely become a bottleneck too. We 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 with linear search instead of the binary tree. The throughput measured for the linear case is presented in Table 4 and shows that when the user population exceeds 100 the cache lookup becomes the bottleneck. When the user population reaches 5000 the maximum throughput obtained is drastically diminished.

Throughput [Mbps]

		Linear Search	Binary Search
	1	65	63
	11	64	62
	101	63	62
Number of	301	55	62
entries in	1001	44	61
routing cache	4001	22	61
	6001	14	61
	100001	-	60
	1000001	-	59

IP forwarding throughput = 73 Mbps.

Table 4. Node Throughput.

Figure 5. Node Throughput
URL:http://comet.columbia.edu/cellularip/publications/draft-gomez-cellularip-perf-00.pdf

The advantage of using paging is now clear from the results. If for example, a network serves n mobile hosts at a time, 0.1*n active at a time, then we can scale up the results obtained in Table 4 by a factor of 10 for example. In addition, with state-of-the-art L2 hardware Cellular IP will out perform these results in terms of the number of users that can be supported. This means that scalability limits are promising, but no doubt there are limits. However this is the price paid for the simplicity of the protocol design and smooth handoff support. This confirms the strength of Cellular IP as an access technology that can interwork with Mobile IP to provide global mobility.

Another way to leverage the scalability problem created by using per-host routes is to shorten the validity times of active entries in the routing cache and/or reducing the duration that a mobile host actively keeps its route updated as in the case of active state. These two procedures further limit the number of valid entries in a routing cache thus reducing route searching times to a minimum for incoming packets. Shortening validity times of routing entries can only be achieved at the expense of increasing the transmission rate of route-update packets in mobile hosts. On the other hand, by reducing the active-state-timeout value the mobile host can limit the period of time it keeps its route updated (e.g., transmit route-update packets) after it received the last packet. Therefore a compromise between the size of the routing cache and the load of control packets in the access network is important.

For full details associated with the evaluation of Cellular IP draft-valko-cellularip-01.txt see [3]

References

[1] "IP Mobility Support," C. Perkins, ed., IETF RFC 2002, October 1996.

[2] "Cellular IP," A. Valko, Z. Turanyi, A.T. Campbell, J. Gomez, Internet Draft, draft-valko-cellularip-01.txt, October 1999, Work in Progress.

[3] "Performance of a Cellular IP", A. T. Campbell, J. Gomez, S. Kim, A. Valko, Z. Turanyi and C.-Y. Wan October 1999 (submitted for publication).

[4] Cellular IP Source Code Distribution, comet.columbia.edu/cellularip, November, 1999.

Authors' Addresses

Andrew T. Campbell, Sanghyo Kim, Javier Gomez, Chieh-Yih Wan
Department of Electrical Engineering, Columbia University
Rm. 801 Schapiro Research Building
530 W. 120th Street, New York, N.Y. 10027
phone: (212) 854 3109
fax : (212) 316 9068
email: [javierg,campbell,shkim2,wan]@comet.columbia.edu

Zoltan R. Turanyi, Andras G. Valko
Ericsson Traffic Analysis and Network Performance Laboratory
H-1300 Bp.3.P.O.Box 197, Hungary
phone: +36 1 437 7774
fax : +36 1 437 7219
email: andras.valko,@lt.eth.ericsson.se,
Zoltan.Turanyi@eth.ericsson.se