

# Performance Analysis of SINEMO: Seamless IP-diversity based Network Mobility

**Pulak K Chowdhury,**

**Abu S Reaz,**

**Mohammed Atiquzzaman**

School of Computer Science,

University of Oklahoma,

Norman, OK 73019-6151

Email: {pulak, sayeem.reaz,  
atiq}@ou.edu

**William Ivancic**

Satellite Networks & Architectures Branch,

NASA Glenn Research Center,

21000 Brookpark Rd. MS 54-8,

Cleveland, OH 44135

Email: wivancic@grc.nasa.gov

**Abstract**—IETF has proposed Mobile IPv6-based Network Mobility (NEMO) basic support protocol (BSP) to support network mobility. NEMO BSP inherits all the drawbacks of Mobile IPv6, such as inefficient routing path, single point of failure, high handover latency and packet loss, and high packet overhead. To address these drawbacks, we proposed an IP diversity-based network mobility management scheme called Seamless IP-diversity based Network MObility (SINEMO). In this paper, we develop analytical models to analyze and compare the performance of SINEMO and NEMO BSP. Our analysis shows that SINEMO enhances the performance of network mobility compared to NEMO BSP. We have implemented a testbed for SINEMO to support our claim of SINEMO's high performance.

## I. INTRODUCTION

A mobile network is a set of IP nodes that move collectively as a unit. Space satellites, trains, or ships with several IP enabled devices are examples of mobile networks. IETF has recently proposed Network MObility (NEMO) Basic Support Protocol (BSP) [1] to support the mobility of a network. NEMO BSP [1] is an extension of Mobile IPv6 and allows all nodes in the mobile network to continue ongoing connection while the network moves. In the NEMO BSP architecture, a Mobile Router (MR) takes care of all the nodes within the Mobile Network (MN). Mobile Router allows an entire network to roam; thus devices connected to the MR are not aware of mobility.

NEMO BSP is based on Mobile IPv6, and hence it inherits the following drawbacks of Mobile IPv6 - (1) in NEMO BSP, all packets are routed through the HA of the mobile router, giving rise to inefficient routing path, (2) packet overhead also increases for encapsulating packet twice, (3) during handover, the MR has to acquire its new care of address in the foreign network and register the new address with its HA which increases handover latency due to multiple level of indirection; incurring packet loss during handover period. A number of proposals to improve performance of NEMO BSP have been introduced in the literature. Petander et al. proposed a make-before-break handover scheme to improve the handover and routing performance of NEMO BSP [2]. Other papers ([3], [4]) proposed different route optimization techniques to improve the

routing of NEMO BSP. Ryu et al. [5] proposed a seamless handover technique for NEMO BSP. The above papers individually address various limitations of NEMO BSP. However, none of them provide a complete solution, taking into account all the limitations of NEMO BSP [6].

To address the aforementioned drawbacks of NEMO BSP, we have proposed an IP diversity-based network mobility management scheme called Seamless IP diversity based Network MObility (SINEMO) [7]. SINEMO is an extension of SIGMA (Seamless IP diversity based Generalized Mobility Architecture) [8], an IP-diversity based mobility management scheme developed through collaborative efforts of NASA and University of Oklahoma for handling the mobility of individual nodes in IP networks; and SINEMO supports the mobility of a whole network. SINEMO is an end-to-end solution instead of a network layer solution (like NEMO BSP) for network mobility, and has the following advantages over NEMO BSP. SINEMO - (1) exploits IP diversity [7] to achieve a seamless handover between adjacent access points, (2) is easier to deploy because of no required change in the Internet infrastructure, (3) co-operates with Internet security protocols, (4) is efficient in utilization of network bandwidth due to the absence of tunnelling, (5) utilizes wireless links efficiently (i.e. minimum signalling), and (6) has low latency and packet loss during handover [7].

The *objective* of this paper is to develop analytical models of the performance of SINEMO and NEMO BSP, and compare NEMO BSP and SINEMO based on those performance measures. We also present the performance of SINEMO using experimental results. Our results show that, in addition to removing the limitations and disadvantages of NEMO BSP, SINEMO performs better than NEMO BSP. Our *contributions* in this paper are - (i) developing analytical models of SINEMO and NEMO BSP to evaluate and compare their performance, and (ii) illustrating and analyzing the experimental results of SINEMO to show its performance.

The rest of the paper is organized as follows: Sec. II describes the architecture of SINEMO. Sec. III briefly discusses NEMO BSP architecture. In Sec. IV, we develop analytical models of the performance of SINEMO and NEMO BSP. Sec. V compares the performance of SINEMO with NEMO BSP. Sec. VI demonstrates and analyzes the experimental results of SINEMO. Finally, concluding remarks are included in Sec. VII.

The research reported in this paper was funded by NASA Grant NAG3-2922.

## II. ARCHITECTURE OF SINEMO

In this section, we provide a brief description of SINEMO architecture. Detailed description of SINEMO architecture can be found in [7]. Fig. 1 presents the architecture of SINEMO. A Mobile Network (MN) in SINEMO consists of multi-homed Mobile Routers (MR) (MR1 and MR2 in Fig. 1) which can be connected to two wireless networks through Access Points A and B. Hosts inside the MN can be Fixed Hosts or Mobile Hosts (MH). MHs can perform handover inside the MN; thus exhibiting nested mobility.

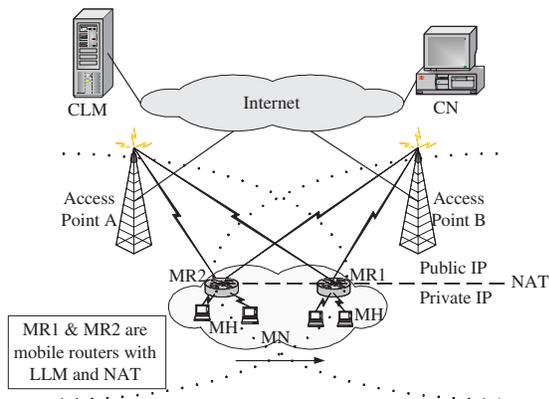


Fig. 1. Architecture of SINEMO.

MRs in the MN act as gateways between all the hosts in MN and the Internet. When MN moves into the coverage area of Access Point A, MR obtains its own public IP address and one or more public IP address prefixes for serving the MHs. MR provides each host (under its coverage) inside the MN with a private IP address, and also reserves a public IP address for the host. The hosts are not aware of their public IP addresses; they use only the private IP addresses for connectivity. The MR maintains *one to one mapping* of the public and private IP addresses of its MHs. When MN changes subnet due to handover, MR gets a new public IP address and prefixes from the new Access Point B (Fig. 1). After handover, only the public addresses are changed in the address mapping at MR, the private IP addresses of the MHs remain unchanged. MR thus hides mobility from the MHs. MR uses NAT (Network Address Translation) to maintain one to one IP address mapping, and to translate between the hosts' private and public IP addresses [7]. This mapping of public and private IP addresses of MHs has the advantage of reducing signaling across the air interface [6], as the MHs will not generate any dynamic updates or binding updates while the MN moves.

A Central Location Manager (CLM) maintains the IP addresses of MRs in an MN. A Local Location Manager (LLM), co-located with the MR inside the MN, is used to keep the public IP addresses of the hosts. When CN wants to send data to a host, it queries the CLM; the query is forwarded to the LLM. LLM responds with the public IP address of the MH directly to the CN. Data packets sent by CN to MH are intercepted by the MR and forwarded to the MH after address translation.

## III. ARCHITECTURE OF NEMO BSP

In NEMO BSP [1], the Mobile Router (MR) takes care of all the hosts in the MN by ensuring continuous connectivity even

as the MR moves and changes its point of attachment to the Internet. A Mobile Router (MR) has a unique IP address and has one or more prefixes that it advertises to the MHs attached to it. MR provides complete transparency of network mobility to the MHs. The MR establishes a bi-directional tunnel with its Home Agent (HA) to pass all the traffic between the MHs and the correspondent nodes.

When a MR moves away from its home network and changes its point of attachment, it acquires a new care-of-address from the visited network. It then sends a binding update to its HA which creates a cache entry binding MR's home address with its care-of-address, and creates a tunnel between HA and MR. When a correspondent node sends data to a MH, it is routed to the HA of MR. The HA looks at its cache entry and forwards the packet to the MR using the bidirectional tunnel. Finally, MR receives the packet, decapsulates it, and forwards it to the corresponding node in the mobile network.

## IV. PERFORMANCE MODEL AND ASSUMPTIONS

Performance of a mobility management scheme has three major components: *transmission delay*, *throughput*, and *number of binding updates*. *Transmission delay* is the time needed to send a data packet from CN to MH. *Throughput* is defined as the number of total useful bytes that are received by a MH during a time unit (granularity). *Number of binding updates* gives an estimate of signaling costs during handover in mobile networks. In this section, we develop analytical models to quantify the above three performance measures for NEMO BSP and SINEMO.

Long transmission delay negatively affects the throughput and user response time. Binding updates result in signaling traffic, and is another important concern during handover of the mobile network. In NEMO BSP, CNs and HAs have to be updated by new address bindings after handover. Similarly, in SINEMO, CNs and the location manager have to be updated with the new address. Frequent binding updates can cause waste of wireless bandwidth [9], as these binding updates have to be transferred over the wireless link. If the binding update rate is high, the wireless link will be inefficient, i.e., majority of the bandwidth will be dedicated to signaling. This is in contrast with the design characteristic of network mobility protocols. We therefore consider number of binding updates for performance comparison of NEMO BSP and SINEMO. We now develop the analytical models for the three main performance metrics of interest.

### A. Variables for NEMO BSP and SINEMO

Here, we declare all the variables that are necessary for developing analytical models of the performance of both NEMO BSP and SINEMO.

#### Variables common to NEMO BSP and SINEMO:

$h_{mn}$  = Average number of hops from first MR to AR of visiting network

$h_{in}$  = Average number of hops in the Internet

$h_p$  = Average number of hops from Internet to arbitrary HA (in NEMO) = Average number of hops from Internet to CN (If we consider Internet is connected directly to CNs and HAs)

$N_{mr}$  = Number of MRs in a nested level

$N_{mh}$  = Number of MHs attached with lowest level's MR  
 $N_{cn}$  = Number of CNs communicating with the MHs in the mobile network  
 $L_p$  = Latency, including processing at network and DL layer  
 $R_{bw}$  = Bandwidth of a network link  
 $L$  = Number of nested levels in the mobile network  
 $\theta$  = Maximum data segment size  
 $p$  = Packet loss probability, defined as the probability of a packet being lost in the link  
 $C$  = Constant of proportionality, adopted from [10].

**Variables for NEMO BSP only:**

$P_{ed}$  = Packet encapsulation/decapsulation time  
 $h_{ha}$  = Average number of hops between inter domain HAs  
 $h_{cn}$  = Average number of hops from the HA of MH to CN  
 $\delta_{TD}^{bsp}$  = Transmission delay from MH to CN for NEMO BSP  
 $\Delta_{RTT}^{bsp}$  = Round Trip Time (RTT) from MH to CN for NEMO BSP

**BSP**

$\Gamma_{TH}^{bsp}$  = Throughput for NEMO BSP  
 $\mu_{BU}^{bsp}$  = Number of binding updates for NEMO BSP

**Variables for SINEMO only:**

$P_t$  = Time needed for NAT translation  
 $\delta_{TD}^{sinemo}$  = Transmission delay from MH to CN for SINEMO  
 $\Delta_{RTT}^{sinemo}$  = Round Trip Time (RTT) from MH to CN for SINEMO  
 $\Gamma_{TH}^{sinemo}$  = Throughput for SINEMO  
 $\mu_{BU}^{sinemo}$  = Number of binding updates for SINEMO

**B. Performance Model of NEMO BSP**

NEMO BSP uses bidirectional tunnelling without route optimization for transferring data from MH to CN. Data packets are routed through the tunnel between HA and MR. We consider bidirectional tunnelling as this is the baseline for all improvements of NEMO BSP. NEMO BSP supports nested mobility. Let us consider that MH exists at the nested level  $L$  (in Fig. 2,  $L = 3$ ).

Lim et al. [9] have quantified transmission latency, and number of binding updates in NEMO BSP. In the following, we modify those equations to find transmission latency, throughput and number of binding updates of NEMO BSP so that they fit with our model for SINEMO performance.

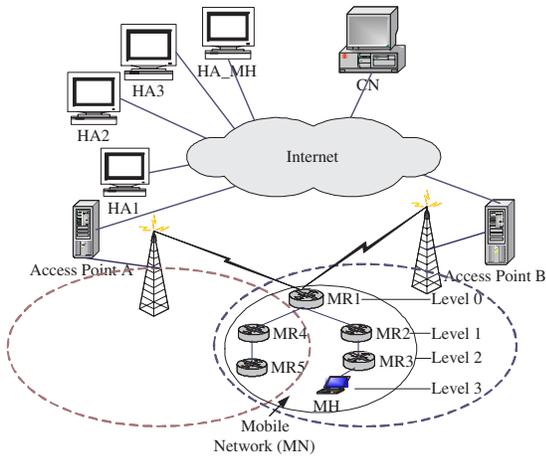


Fig. 2. Nested mobility in NEMO BSP.

1) **Transmission Delay:** In the bidirectional tunnelling approach for NEMO BSP, MH sends data to its own HA via upper levels' MRs' HA through tunnelling, followed by HA forwarding the data to the CN. Data from CN also follow the same route. Thus, the delay time per hop is the sum of processing delay time ( $L_p$ ), and the transmission delay time ( $\theta/R_{bw}$ ) at the relevant link.

We can count the number of hops in the path from MH to CN by following the transmission path. In Fig. 2, the transmission path from MH to CN is  $MH \rightarrow MR3 \rightarrow MR2 \rightarrow MR1 \rightarrow AR \rightarrow Internet \rightarrow HA1 \rightarrow HA2 \rightarrow HA3 \rightarrow HA_{MH} \rightarrow CN$ . Thus, number of hops in the transmission path from MH to CN will be the sum of nesting level, average hop count from the MH to the top (0) level MR ( $h_{mn}$ ), average hop count from top level MR to access router (AR) (usually 1), average hop count in the Internet ( $h_{in}$ ), average hop count from Internet to top level MR's HA ( $h_p$ ), average hop count ( $h_{ha}$ ) between  $(L + 1)$  HAs and average hop count from MH's HA to CN ( $h_{cn}$ ), i.e.,  $(L + h_{mn} + h_{in} + h_p + (L + 1) * h_{ha} + h_{cn})$ .

Each MR and HA also requires a processing time of  $P_{ed}$  for tunnelling. There will be  $2(L + 1)$  such processing time as there are total  $(L + 1)$  tunnels, and tunnel processing (encapsulation/decapsulation) is done at the tunnel entry and the opposite end of the tunnel.

Therefore, the transmission delay in NEMO BSP is,

$$\delta_{TD}^{bsp} = (L_p + \theta/R_{bw}) * (L + h_{mn} + h_{in} + h_p + (L + 1) * h_{ha} + h_{cn}) + 2(L + 1) * P_{ed} \quad (1)$$

2) **Throughput:** Without loss of generality, we consider that Round Trip Time (RTT) is twice one way transmission delay between MH and CN. Thus, RTT in NEMO BSP is,  $\Delta_{RTT}^{bsp} = 2 * \delta_{TD}^{bsp}$ . According to Mathis et. al [10], throughput ( $\Gamma_{TH}$ ) is proportional to maximum segment size ( $\theta$ ) and inversely proportional to RTT ( $\Delta_{RTT}$ ). Thus, throughput in NEMO BSP is given by,

$$\Gamma_{TH}^{bsp} = (\theta/\Delta_{RTT}^{bsp}) * (C/p) \quad (2)$$

3) **Number of Binding Updates:** In NEMO BSP, with bidirectional tunnelling, the visiting mobile objects (MH and MR) update bindings through control message to their own HA (i.e.,  $(N_{mr} + N_{mh})$ ) and each CN (i.e.,  $N_{cn}(N_{mr} + N_{mh})$ ) [9]. Therefore, the number of binding updates is given by,

$$\mu_{BU}^{bsp} = (N_{cn} + 1) * (N_{mr} + N_{mh}) \quad (3)$$

**C. Performance Model of SINEMO**

In SINEMO, the connection between MH and CN is direct, i.e., not via any other agent (such as HA in NEMO BSP). Data packets are routed through the best (network chosen) path as determined by IP layer's well proven routing algorithms. This results in significantly lower transmission delay and higher throughput in SINEMO when compared to NEMO BSP.

1) **Transmission Delay:** As in NEMO BSP, the delay time per hop is the sum of network and data link layer processing times ( $L_p$ ), and the transmission delay time ( $\theta/R_{bw}$ ) at the relevant hop. The number of hops in the transmission path from MH to CN can be counted by following the transmission path. In Fig. 3, the transmission path from MH to CN is  $MH \rightarrow MR3 \rightarrow MR2 \rightarrow MR1 \rightarrow AR \rightarrow Internet \rightarrow CN$ . Thus,

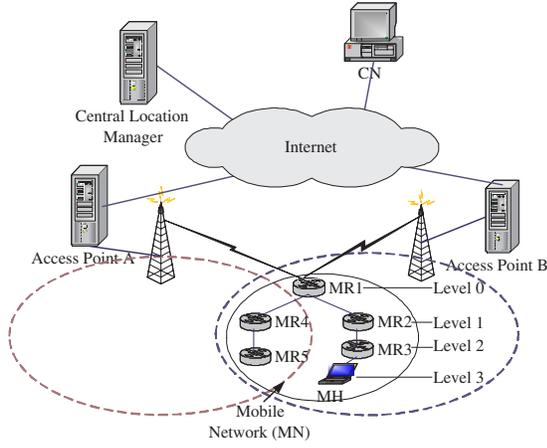


Fig. 3. Nested mobility in SINEMO.

number of hops in the transmission path from MH to CN is the sum of nesting level, average hop count from the MH to the top (0) level MR ( $h_{mn}$ ), average hop count from top level MR to access router (AR) (usually 1), average hop count in the Internet ( $h_{in}$ ), i.e.,  $(L + h_{mn} + h_{in} + h_p)$ .

Each MR has to perform address translation in the NAT, resulting in  $L$  address translation time ( $P_t$ ) for  $L$  levels. Thus, the transmission delay in SINEMO is,

$$\delta_{TD}^{sinemo} = (L_p + \theta/R_{bw}) * (L + h_{mn} + h_{in} + h_p) + L * P_t \quad (4)$$

2) *Throughput*: Using the same argument as for NEMO BSP, the RTT in SINEMO is,  $\Delta_{RTT}^{sinemo} = 2 * \delta_{TD}^{sinemo}$ , and the throughput is,

$$\Gamma_{TH}^{sinemo} = (\theta/\Delta_{RTT}^{sinemo}) * (C/p) \quad (5)$$

3) *Number of Binding Updates*: In SINEMO, each time handover occurs, the top level MRs only update the central location manager (CLM) and the CNs communicating with the MHs in the MN. Therefore, the number of binding updates is,

$$\mu_{BU}^{sinemo} = N_{cn} + N_{mr} \quad (6)$$

## V. PERFORMANCE ANALYSIS

In Sec. IV, we developed the analytical models for performance of NEMO BSP and SINEMO. In this section, we evaluate and compare the performance of these two architectures using the following parameter values (adopted from [9]):  $L_p = 5$ ,  $R_{bw} = 2MB$ ,  $\theta = 2KB$ ,  $P_t = 5$ ,  $h_{mn} = 1$ ,  $h_{in} = 5$ ,  $h_{ha} = 3$ ,  $h_p = 1$ ,  $h_{cn} = 3$ ,  $C = \sqrt{3/2}$ ,  $p = 0.1$ ,  $N_{mr} = 5$ ,  $N_{mh} = 10$ .

Fig. 4 shows the impact of number of nested levels on the transmission delay of NEMO BSP and SINEMO. Although the delay increases sharply with the nested level in the case of NEMO BSP, there is no significant increase in delay in the case of SINEMO. We can also see that the transmission delay of SINEMO is much lower compared to NEMO BSP for all nested levels. This is due to bidirectional tunnelling in NEMO BSP where data packets have to be routed through all the home agents of the MRs in various nested levels; this inefficient routing path increases the transmission delay. On the other hand, in SINEMO, data packets travel through the

current access point (in Fig. 1, access point A), and are thus transferred between CN and MHs through an efficient routing path as determined by the routing protocol of IP layer.

Fig. 5 indicates that SINEMO has better overall throughput compared to NEMO BSP for different nested levels in the mobile network. Higher transmission delay in NEMO BSP results in lower throughput as compared to SINEMO. Fig. 6 depicts the impact of increasing number of communicating CNs on the number of binding updates. MRs in SINEMO have to only update the central location manager and CNs, and hence fewer binding update messages as compared to NEMO BSP.

From the above performance comparison, it is very clear that SINEMO performs better as a network mobility protocol compared to NEMO BSP. To support our claim, we provide below some experimental results of SINEMO from our SINEMO testbed.

## VI. SINEMO EXPERIMENTAL RESULTS

The analytical models developed in Sec. IV show the effect of nested levels on throughput, transmission delay, and binding updates for SINEMO and NEMO BSP. We also want to measure SINEMO handover delay and real-time throughput that is achievable by SINEMO when managing network mobility. This section describes the experimental testbed of SINEMO, and measurements of handover delay, packet loss rate and throughput of SINEMO extracted from the experimental testbed.

### A. Testbed Architecture

Fig. 7 shows the experimental testbed of SINEMO with one level of nested mobility. More levels nested mobility is not feasible in the restricted setup of laboratory as big vehicles like trains, or ships are required to execute nested mobility experiments. In Fig. 7, there are two access points which manage two subnets, 10.1.8.0/24 and 10.1.6.0/24. A trolley with an MR and two MHs (two laptops) works as the mobile network (MN). The MN has its own private address space (10.1.5.192/26), and MR manages this address space.

While MN is in the first subnet (10.1.8.0/24), MR gets a public address space (10.1.8.192/26) for serving the mobile hosts. When MN moves from subnet 1 (10.1.8.0/24) to subnet 2 (10.1.6.0/24), MR gets a new public address space (10.1.6.192/26) from access point B. MHs inside the MN use only the private addresses (10.1.5.192/26) for communication.

### B. Hardware and Software Configuration

The hardware configuration in the testbed consists of four desktop computers and two laptops with processor speeds between 1.8 and 3.6 GHz and memory sizes between 512MB and 1 GB. We used D-link and Linksys wireless routers as access points with 802.11b wireless networks. The following wireless cards are used in different machines: SMC2802W PCI, Allnet PCI, Integrated Intel IPW2100, and Avaya PCMCIA card.

Our testbed uses three client server programs with SINEMO functionalities in CN, MH and MR. CN works as data server and MHs work as data clients. We use FC3 and Red Hat 9 Linux OS in different computers with linux kernel SCTP (`1ksctp`) (SCTP is underlying transport protocol of SINEMO).

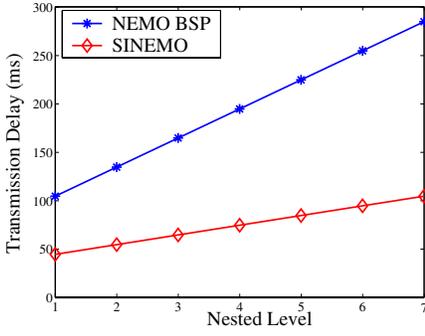


Fig. 4. Transmission delay of NEMO BSP and SINEMO vs. no. of nested levels.

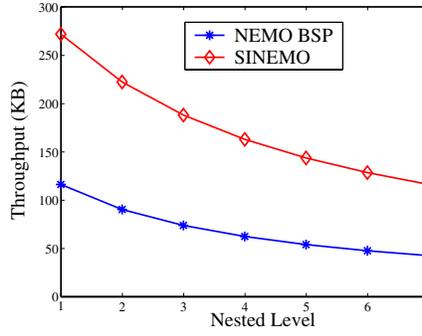


Fig. 5. Throughput of NEMO BSP and SINEMO vs. no. of nested levels.

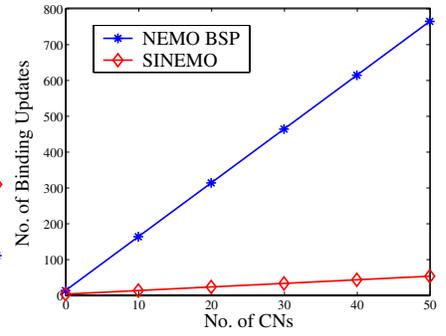


Fig. 6. No. of binding updates of NEMO BSP and SINEMO vs. no. of CNs communicating.

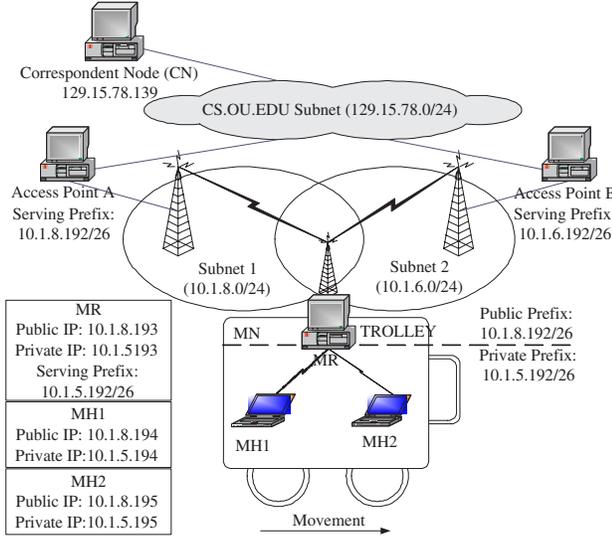


Fig. 7. SINEMO experimental testbed.

### C. Results

From our testbed, we captured packet flows at MHs, CN and MR using Ethereal [11], a network protocol analyzer.

1) *Trace Graph*: Fig. 8 shows the packet trace at MR during a handover of the MN from subnet 1 to subnet 2 with data transmitted from CN to MH2 (10.1.5.195). We present only one representative SINEMO handover scenario here, but SINEMO behaves in the same way during all other handovers. The trace graph shows the sequence numbers of data and acknowledgement packets versus time. During handover, the segment sequence numbers ranged from 2955353701 to 2955353751. In Fig. 8, segment sequence numbers are presented as mod 2955353700. We can see that SCTP data segments are sent to MH2 through MR using subnet 1 address (10.1.8.195 in Fig. 7) until 41.954 sec. (point  $t_1$ ), and then through the subnet 2 address (10.1.6.195 in Fig. 7) at  $t_2$ . *Handover latency* is defined as the time interval between the last data segment received through the old path and the first data segment received through the new path from the CN to MR. We find that  $t_2 = 41.111231$  sec. and  $t_1 = 41.053183$  sec., giving a very small handover latency of  $t_2 - t_1 = 60$  msec. This small handover latency consists of the time needed for dynamic address update, NAT rules update, and routing table update at MR during SINEMO handover (Sec. II).

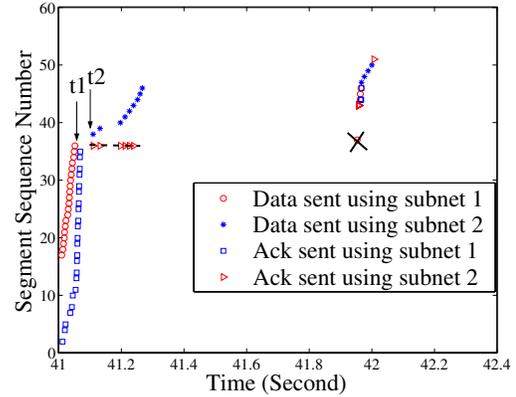


Fig. 8. Trace graph of SINEMO.

As shown in Fig. 8, almost all the packets are successfully delivered to MH2; only one data packet (packet 37, marked with  $\times$ ) and a few acknowledgement packets (marked with dotted lines) are lost during SINEMO handover. We define *packet loss rate* as the number of lost packets due to handover divided by the total number of packets sent by CN. In our experimental results, packet loss rate is negligible as only one data packet is lost during SINEMO handover. Thus, *SINEMO experiences low handover latency, and low packet loss rate* during handovers of mobile networks. The lost packet (no. 37) is transmitted later at 41.954 sec. through the secondary path (through 10.1.8.195) during handover.

2) *Packet Flow Diagram*: In Fig. 9, we show the packet flow from CN to MH2 through MR during SINEMO handover. We see that the public destination addresses (10.1.8.195 and 10.1.6.195) of all packets destined to MH2 are converted to the private address (10.1.5.195) of MH2.

Using this packet flow diagram, we determine SINEMO handover latency. Last data packet through 10.1.8.195 arrives at time 41.053 sec. and the first data packet through 10.1.6.195 comes at 41.111 sec; these packets are marked with boxes in Fig. 9. Handover latency is  $41.111 - 41.053$ , i.e. 60 msec.

During SINEMO handover (from time 41.053 to 41.111 sec.), the packet flow diagram shows that data packets are sent to MH2 through the primary path (initially through 10.1.8.195 and later through 10.1.6.195), and acknowledgement packets are sent through the secondary path (through 10.1.8.195) (Fig. 9).

