Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777086

PERFORMANCE ANALYSIS OF NETWORK MOBILITY BASIC SUPPORT PROTOCOL FOR INTELLIGENT TRANSPORTATION SYSTEMS USING NS3

Ms. D.Manopriya[1], Dr. M.Manimaran[2]
[1]Research Scholar, [2]Associate Professor
Department of Computer Science, Sri Krishna Adithya College of Arts and Science
Kovaipudur, Coimbatore, Tamil Nadu, India
manovbs01@gmail.com, mani.professor@gmail.com

Abstract
For effective data transfer in an intelligent transport system for a smart city, uninterrupted connectivity for each user while moving is crucial. Due to high mobility, managing handoffs for a group of mobile users in a vehicle (bus, train, or airline) can be difficult. A protocol must be put in place in order to manage handoffs efficiently. To meet this requirement, the Network Mobility Basic Support protocol was proposed. It is MIPv6's (Mobile IPv6) expanded version. However, ns-3, the most popular open-source simulator, still does not allow network mobility with its MIPv6 implementation. In this work, we added to the ns-3 library by changing the existing MIPv6 module the capability of the NEMO-BS protocol.
Keywords: Intelligent Transportation Systems, MobileIPv6, Network Mobility Smart City

1 Introduction

Every user must have a smooth connection to the Internet in the age of the Smart City and the Internet of Things (IoT). One of the key components of the smart city is intelligent transport systems. Due to the mobility of the vehicle [1] in cases of group mobility, or when a group of users is travelling in a vehicle (bus, train, or aeroplane), the wireless interface of the vehicle occasionally switches the on-road access points. The IP handoff is the name of this process. The users of vehicles suffer a serious decrease in service as a result. By extending the fundamental Mobile IPv6 (MIPv6) architecture, the Internet Engineering Task Force (IETF) standardizes the Network Mobility Basic Support protocol to handle it [2].By preserving continuing sessions during handoff, the goal of this is to give vehicle mobile users continuous Internet connectivity. Network simulator ns-3 can be used with UMIP (Usagi Patched MIPv6 stack), an open-source Linux implementation of the MIPv6 and NEMO-BS protocols. However, it can only be run on ns-3 Direct Code Execution with the native Linux stack, which is why users would need to trace the Linux stack if any NEMO changes were to be proposed. But making any changes to the Linux stack is rather difficult for inexperienced users.

To resolve the aforementioned problem, we modified the MIPv6 module in ns-3 [7, 8] and implemented the NEMO-BS protocol using the ns-3 stack. This could greatly benefit ns-3 users.

The paper is set up as follows: The Literature Survey is covered in Section 2, the Protocol's working principles are covered in Section 3, the Class Diagram implementation is covered in Section 4, simulation results are covered in Section 5, and concluding observations are covered in Section 6.

2 Literature Survey

To lessen the signaling issue, they suggested in [1] a group-based network mobility regulating mechanism. In addition, they lessen the handover latency. However, they did not look into the best way to group automobiles together or they did not use the neighboring association while grouping the vehicles.

They presented a NEMO support protocol for Intelligent Transportation Systems in [8], which supports both mobility management and handover. However, they did not back the promised flawless connectivity during a handover.

Hager et al.'s [9] description of MINT, a mobile internet router with sufficient processing power to carry out the necessary communication protocol activities while enabling connections for the nodes, is an example.The MINT router offers transparency of the communication software, therefore using one of these routers to access the Internet does not necessitate any changes to the fundamental mobility support software.

The use of a mobile router for the mobile network is expressly mentioned in the Request for Comments [10], which focused on IP mobility support.

The attempt to implement the NEMO-BS protocol using the ns-3 network simulator, which guarantees flawless network connectivity and IP handoff management, while taking into account all the restrictions of the aforementioned study.

3 Working Principle of NEMO-BS Protocol

Fig. 1 illustrates the protocol's basic operation. A Mobile Network Node (MNN), Mobile Router (MR), Home Agent of the MR (MR-HA), and Correspondent Node (CN) are the network elements for the handoff process.

First, the MNN establishes a connection with the MR, which serves as the vehicle's default Internet service provider, and obtains a Care-Of-Address (CoA) that is set up using the MR's prefix.

Step 2: MNN links the CoA with its HA (MNN-HA) using a Binding Update (BU) process because its home prefix differs from that prefix.

Step 3: The MR-HA and the BU process are carried out by the MR when the access point changes. Two tunnels between entities (MR and MR-HA, MNN and MNN-HA) are the result of the BU processes.

A packet travels the following path from CN to MNN:MNN - HA MR – HA MR MNN. The MNN-HA encapsulates it first so that it can travel to the MR's house. When a packet reaches MR's home, it is automatically forwarded to the MR-HA because that HA has links to MR's home. The marketed prefixes of the MR are bound to the MR's current CoA via MR-HA. Therefore, the packet is encapsulated by the MR-HA with the CoA of the MR as the destination. The tunnel headers are decapsulated in the MNN first, followed by the MR, in the reverse sequence.

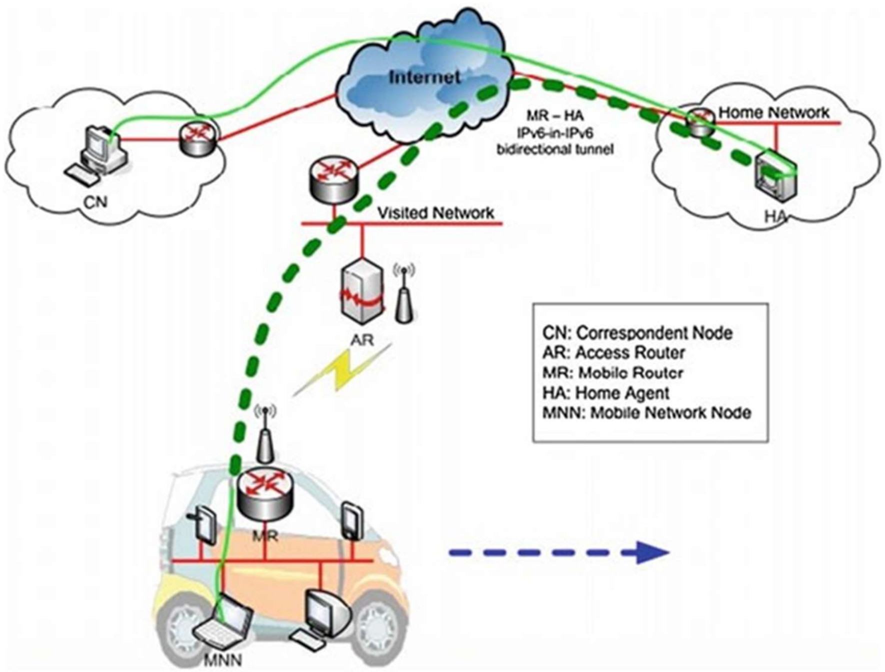

Fig.1NEMO-BSoperation

The NEMO-BS's operational methodology departs from MIPv6 in the following two ways:

• During the BU procedure, the MR registers all of its prefixes in addition to the home address (which it broadcasts to the MNNs).

• Unlike MIPv6, the MR-HA must advertise all of those MR's prefixes to its neighbors in order for the MR-HA to receive all of the packets configured from those prefixes that are intended for the MNNs CoAs.

Here, we'll go into great detail on the NEMO-BS data packet processing and binding update (BU) procedure.

3.1 Binding Update (BU) Process of NEMO-BS

A mobile host in MIPv6 transmits a single BU to its HA whenever it configures a new address in its interface. But with NEMO-BS, each MH in the mobile network transmits BU to its corresponding HA through MR when they configure a new address in the mobile network, as opposed to sending a single BU. Similar to this, MR also sends a BU to its HA when configuring a new IPv6 address in a new subnet using a new access router's advertised prefix. A node starts the Duplicate Address Detection (DAD) method as soon as it configures an address on an interface to check its delicacy. DAD's time-out session is followed by a call to the ns3 core's SetState() method.

3.2 Packet Processing of NEMO-BS

A bidirectional tunnel is constructed between MR and MR-HA following the successful completion of the Binding Update procedure, and another tunnel is established between mobile network MN and MNHA (Fig. 2).

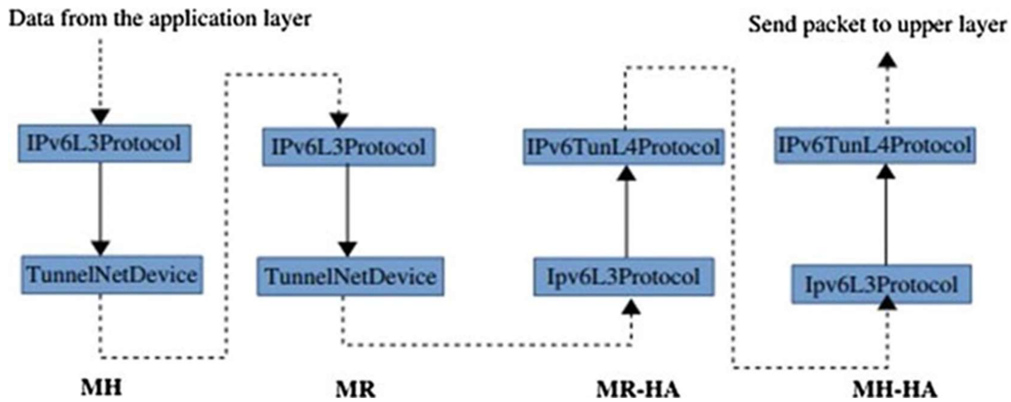

Fig.2 Data packet processing from MH to MHHA

Due to this, the first level encapsulation of a data packet transmitted from a mobile host to a Correspondent Node (CN) in a mobile network is completed by the tunnel between MH and MHHA before it is sent to MR. Following that, MR performs second-level encapsulation due to the tunnel between MR and MR-HA. The MR-HA decapsulates the outer encapsulation of the two-level encapsulated packet before forwarding the remaining packet to the MHHA.

4 Class Diagram Implementation

The existing classes of the implemented MIPv6 module [7] are modified to support the NEMOBS functionalities as shown in the class diagram in Fig.3.We divide our NEMO-BS classes into mainly four modules:

- Header
- Internetstack
- Netdevice
- Helper.

Theimportantoperationsandfunctionalitiesofeachmodulearedetailedasfollows:

4.1 Header

It modified a few of the MIPv6 header message formats already included in ns3 in order to provide the NEMO-BS capability as specified in RFC3963. In the ns3 system, all headers inherit from the basic class Header. The primary data members of NEMO-BS headers and their relationships are shown in Fig. 3 according to RFC3963. We have introduced the m flagR data field to the binding update (BU) and binding acknowledgement (BA) headers for NEMO-BS in ns3 to indicate the state of the mobile node. The HA assumes that a MIPv6 mobile node works as a NEMO-BS MR if this flag value is set in the BU; otherwise, it considers the node to be a MIPv6 Mobile Host (MH). With the current headers in Fig. 3 comes a new mobility option header called Mobile Network Prefix Option Header, which likewise derives from the MIPv6OptionHeader class. The one or more prefixes (128 bits or more) are carried by the BU message from the MR to the HA, and these prefixes are broadcast by the MR-HA so that the mobile host in the mobile network can appropriately receive the packet.

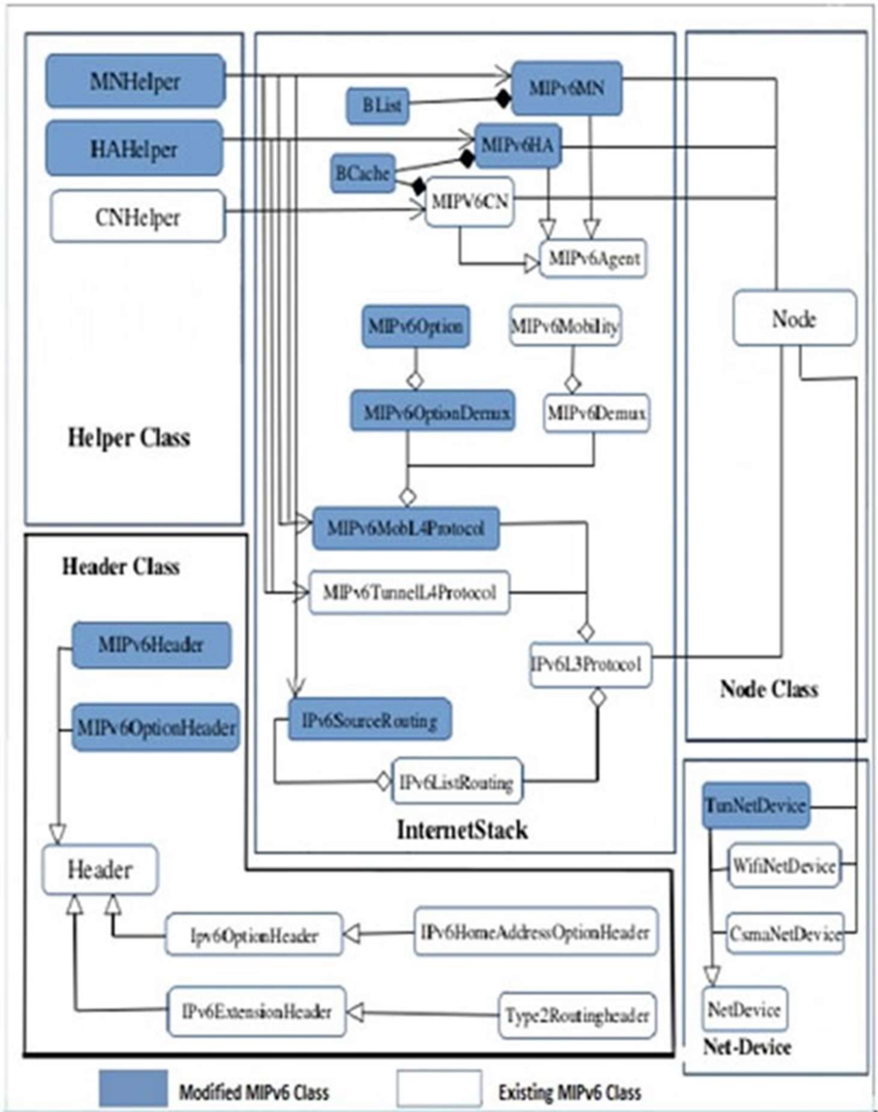

Fig.3 NEMO-BS classes and their functional relationships

4.2 Internet Stack

MIPv6Demux and MIPv6OptionDemux are the two types of Demux classes found in the NEMO-BS implementation. Demux class functionality is detailed in paper [7] in more detail. The MIPv6Option class is expanded to provide a new mobility option for NEMO-BS. While the MIPv6TunL4Protocol deals with data packets during communication, the MIPv6MobL4Protocol deals with mobility messages required for communication. When an IPv6L3 Protocol recognizes a packet from a lower layer, it transmits it to the MIPv6-layer.The MH type value, which differs from a standard data packet, is present in the header of a mobility message or a summarized data packet. The IPv6L3Protocol class checks the value of the MH Type field, and depending on that value, the appropriate higher layer L4 protocol class is invoked. If the MIPv6TunL4Protocol class' Receive () function receives the packet, it verifies the summarized packet tunnel line that might be used to de-capsulate the packet. If a corresponding tunnel interface cannot be discovered, the packet is dropped. A tunnel's MIPv6TunL4Protocol class can be updated, added, or removed.

4.3 Net Device

The TunNet Device class creates a tunnel for the virtual MAC layer. The base class Send() function is reimplemented by the TunNet Device class. The Send() function creates a new IPv6 header after receiving a packet from the upper layer in order to complete IPv6inIPv6 encapsulation. The MIPv6TunL4Protocol class 'TunNet Device class is its primary functional component.

4.4 Helper

This is ranked first among all classes that have been introduced. When a user wishes to use the NEMO-BS protocol, they merely need to call the Install() function of the MIPv6Helper class to install the functionality on a specific node. They don't need to worry about the convoluted internal structure.

5 Simulation and Results

Fig. 4 depicts the simulation framework. For the simulation, we used ns-3.25. There are two interfaces on the MR.

It communicates with Access Routers (AR) via Wi-MAX interface and with MNNs via Wi-Fi interface.

Simulator setup: Fig. 5 displays the simulation snapshot. The Mobile Host (MH) and Correspondent Node (CN) function as hosts, while the other nodes act as routers. A middle router R1 links the MR-HA to the AR1 and AR2 via a CSMA interface, while a second middle router R2 connects the CN and MHHA to R1 via a CSMA interface. The MR has two interfaces; one connects it to the access router via Wi-Fi, and the other to the mobile host in the mobile network via Wi-Max.IEEE802.11 radio is used over herein AR1 and AR2. Beacons are generated in every 100ms. Here, the data rate is set at 100.

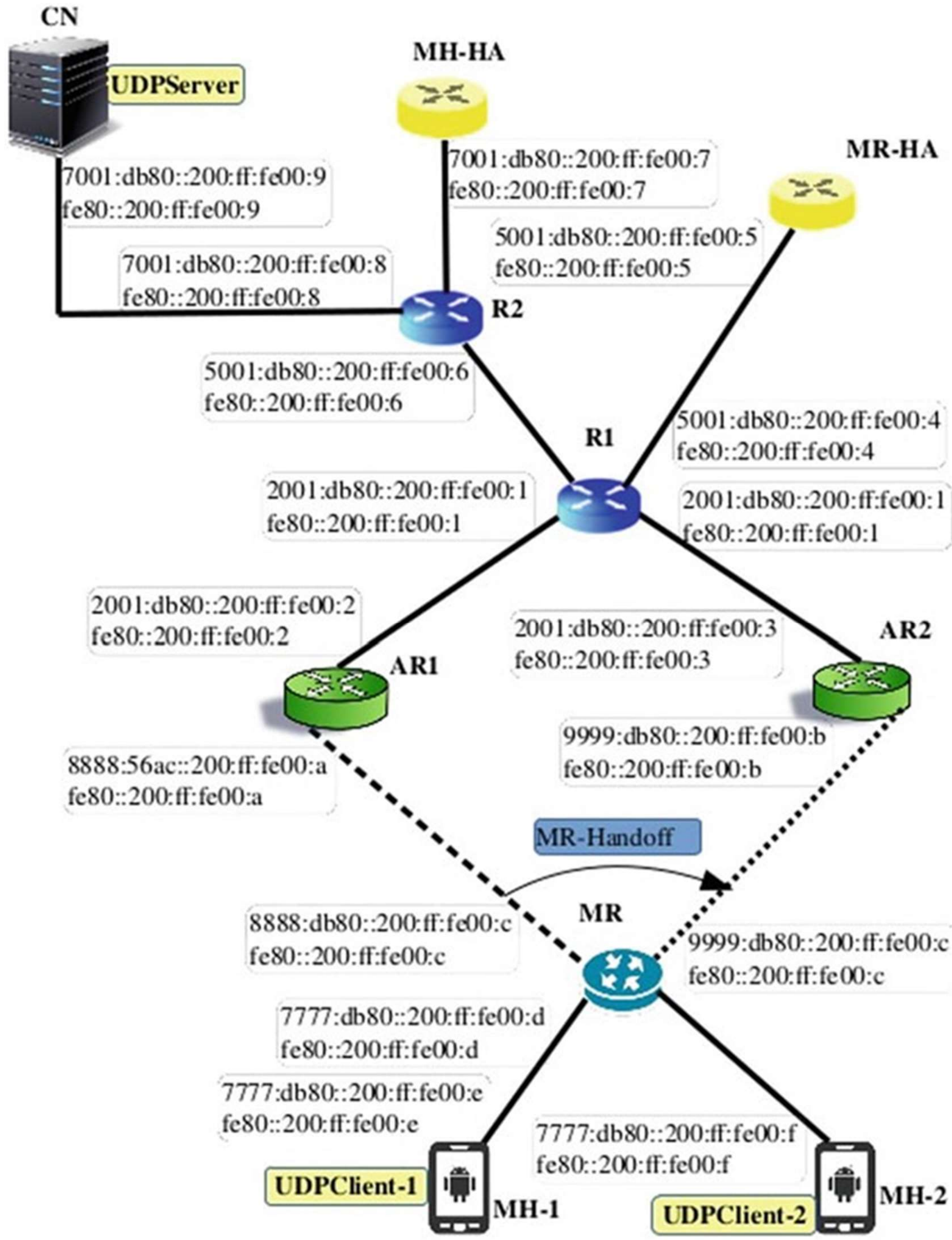

Fig.4 Simulation Framework

Here, AR1 and AR2 employ IEEE 802.11 radio. Beacons are produced every 100 milliseconds. Here, the link delay is set to 1.0 ms and the data rate is 2 Mbps. An programme for a UDP echo-server is deployed on the mobile hosts (MH1 and MH2). An echo-client application with a 1024 packet size and a cap of 100,000 packets is running on the CN. The HoA of MR is expressed from the MR home network prefix, which is 5001:db80:/64 in this case.

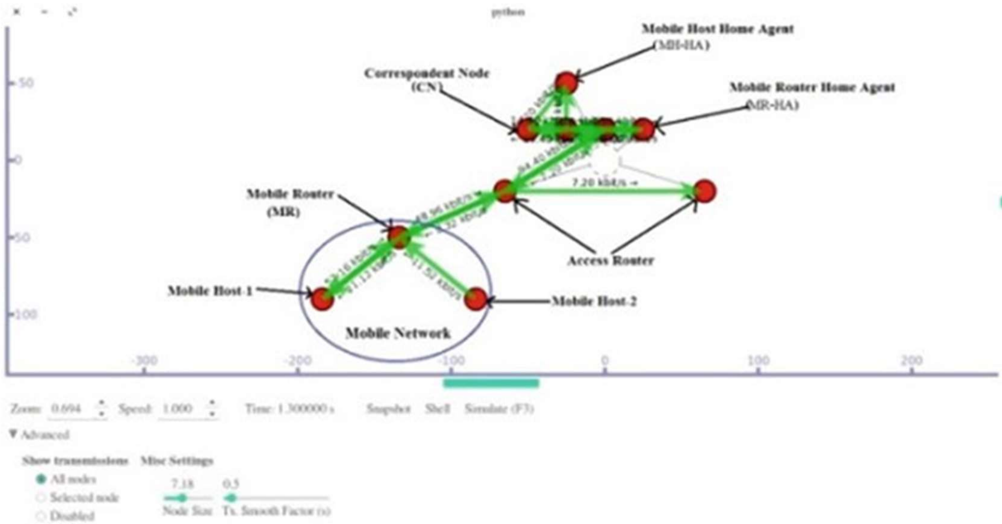

Fig.5 Simulation snapshot of implemented NEMO-BSinNS3

For all MR, a left-to-right motion is employed with a constant velocity model (20 m/s).
Below are the simulation parameters:

• Every 100 ms, the AR1 and AR2 both produce beacons.

• All CSMA links have a link delay of 1.0 ms and a data rate of 2 Mbps.

• The MNNs (MNN is referred to as MH) (MNN-1 and MNN-2) are fixed with a UDP echo-server (let's say at port 9).

• The MR in the mobile network uses constant velocity mobility and moves with a 20 m/s velocity from left to right.

It can determine the delay MNN1 experienced in the mobile network as a result of the handoff of MR from AR1 to AR2 by using the packet capture (PCAP) data. The MNN1 successfully receives the previous packet from the CN at t = 7.240 s. After MR switches points of attachment, MNN1 successfully receives the subsequent packet at t = 12.245 s after MR successfully completes the handoff. Therefore, MNN1's delay was equal to (12.245-7.240) s = 5.005 s.

6 Conclusion

By facilitating a quick and flawless hand-off for a group of users travelling from one location to another, the NEMO-BS protocol offers seamless Internet connectivity. In this study, we added this protocol to MIPv6 in the ns3 environment so that it may be added later as a new library file, which will benefit the uninitiated users. The aforementioned implementation complies with the IETF-defined standard. However, some additional work must be done in the future for the research community to effectively utilize it. Our solution must be validated against some of the best currently available implementations because the open-source simulators like ns-3 currently demand validation for any new implemented module. Above all, it would be quite intriguing to integrate NEMO into LTE.

References

1. Kim,M.S.,Lee,S.K.:Enhancednetworkmobilitymanagementforvehicularnetworks.IEEETrans.Intell.Transp.Syst.17(5),1329–1340(2016)

2. Perera,E.,Sivaraman,V.,Seneviratne,A.:Surveyonnetworkmobilitysupport.SIGMOBILEMob.Comput.Commun.Rev.8(2),7–19(2004)

3. NetworkMobilityBasicSupportProtocol.RetrievedJuly25,2018fromhttps://github.com/prasanta2018/NetworkMobility

4. Bernardos, C.J., De La Oliva, A., Caldern, M., von Hugo, D., Kahle, H.: Nemo: networkmobility.Bringingubiquitytotheinternetaccess.DemonstrationatIEEEINFOCOM(2006)

5. Khan, M.Q., Andresen, S.H., Khan, K.N.: Pros and cons of route optimization schemes fornetworkmobility(NEMO)andtheireffectsonhandovers.In:Proceedingsofthe8thInternational Conference on Frontiers of Information Technology (FIT ’10). ACM, NewYork,NY,USA,Article24

6. Bauer,C.,Ayaz,S.,Ehammer,M.,Grupl,T.,Arnal,F.:Infrastructure-basedrouteoptimizationforNEMObasedoncombinedlocalandglobalmobility.In:MobilityConference,p.61(2008)

7. Rana,M.K.,Rana,B.,Mandal,S.,Saha,D.:ImplementationandperformanceevaluationofamobileIPv6(MIPv6)simulationmodelforns-3.Simul.Model.Pract.Theory72,122(2017)

8. Lee, J.H., Ernst, T., Chilamkurti, N.: Performance analysis of PMIPv6Based network mobilityforintelligenttransportationsystems.IEEETrans.Veh.Technol.61(1),74–85(2012)

9. Hager,R.,Klemets,A.,Maguire,G.Q.,Reichert,F.,Smith,M.T.:MINT-AMobileinternetrouter. In: 1st International Symposium on Global Data Networking, Cairo, Egypt, Dec.13–15,1993(1993)

10. Perkins,C.:IPMobilitySupport,IETF.RFCOctober1996(2002)