Examination of IP Macromobility in OMNeT++ Simulation Environment

Csaba Csegedi, Sándor Imre, Róbert Schulcz, Szabolcs Vajda

Budapest University of Technology and Economics, Department of Telecommunications
imre@hit.bme.hu

Our article deals with IPv6 based mobile networks. First of all we would like to introduce the major novelties of IPv6 from the aspect of mobility – compared to the IPv4. After that we describe the handling of IPv6 mobility, then we introduce the simulation environment made by us and the derived results.

Spreading of mobile computer technology

Spreading of mobile telephones was beyond any expectations of the engineers. By now there is an urging need for mobile transmission of other data than that of speech coding. Because the systems operating nowadays were planned to transmit speech coding data, they are not appropriate for transmitting bigger amount of data.

At the present time the trend in informatics is to introduce services on IP basis from user to user if possible. This technology called All IP. Because of IP being a packet switched data transmission method, it uses resources in a more efficient way than the land-based telephony or the circuit switched GSM system. Thus it needs more complicated protocol to control the packets – especially in case of mobile systems. The IPv4 systems operating today and having almost 20 years of history are not eligible for the current increased demands (appropriate big address field, integrated mobility handling, QoS (Quality of Service) parameters), that is why new systems have to be developed. The solution might be the latest version of IP protocol, the IPv6 [1] [4], which supports integrated mobility at the side of many other novelties.

IPv6 versus IPv4

IPv6 in its operation is basically similar to IPv4 but also integrates several important innovations. Among these innovations the most important are:

Bigger address field

IPv4's most emerging bottleneck is its limited address space. Considering the 4.3 billion addresses provided by 32 bit addressing, one could ask why this would be an issue since the number of computers attached to the Internet is still below 100 million. As a matter of fact, IPv4 addresses are allocated very wastefully. The problem of the limited address space can be alleviated by using address translation techniques like Network Address Translation (NAT) but these methods provoke a number of problems by sacrificing the end-to-end semantics of IP. As the number of subnetworks connected to the Internet grows, routing tables are also becoming slowly unmanageable.

IPv6's perhaps most often discussed feature is its 128 bit addressing [8], offering powerful scalability. This addressing architecture provides 2^{128} (which is 340.282.366.920.938.463.463.374.607.431.768.211.45) addresses for Internet hosts, which means that there will be thousands of IP addresses for each square meter of the Earth's surface. An address space of such an enormous size can probably meet every future demand, independently of wasteful allocation strategies. At least according to the optimistic network planners. In the future not only the computers will have IP addresses but... At the present time there are more mobile handsets than personal computers all over the world and a big segment of these is capable of connecting to any kind of data network, e.g. via WAP. This is not yet to be called real Internet connection but the number of "intelligent" devices that can connect to the network will dramatically increase in the near future. In a few years' time not only the laptops or mobile telephones but PDAs, digital cameras, different Bluetooth devices, telemetry and building informatics system elements, vehicles, and what is more, household devices can access the network with their unique IP addresses.

The new addressing structure also allows more hierarchical backbone routing based on provider topology, which can stop the expansion of routing tables in backbone Internet routers.

Security

While the use of IPSec is optional in IPv4, it is a mandatory, integral part of IPv6. Thus, it is always possible to set up an IPv6 connection in a secure fashion. The multitude of IPv6 addresses also contributes to achieving end-to-end security, since it eliminates the problem of NATs breaking the security during address translation.

IPSec security is implemented with two extension headers in IPv6: the Authentication Header [9] and the Encapsulated Security Payload header [10]. A Security Association (SA) is used to describe how the communicating entities utilize security services in their communication sessions. A SA is identified by three parameters: the Security Parameter Index (SPI), the destination IP address and the identifier of the used security protocol (AH or ESP).

Autoconfiguration

IPv6's tremendous address space would rather become a drawback as a beneficial feature if there were no advanced mechanisms for assignment and management of these addresses. As far as possible, a mechanism like this must provide an automatic, cost-effective and well manageable way for configuring addresses. IPv6 introduces an elegant approach for this task with its Address Autoconfiguration protocol [6]. Besides configuring addresses, the protocol also allows other network parameters to be set up automatically, like the default gateway address, default router, etc. These features allow a host to connect to a network in a plug-and-play manner, without the need of any manual intervention.

There are two ways of configuring a host's address automatically: a stateful and a stateless address autoconfiguration mechanism. Stateless and stateful autoconfiguration complement each other.

Neighbour Discovery

The Neighbour Discovery (ND) [7] protocol replaces the old ARP protocol used to determine the link-layer addresses of hosts from their network addresses. It uses ICMPv6 (Internet Control Message Protocol v6) messages to discover a host's surrounding network and neighbour hosts. ICMPv6 is the new version of IPv4's ICMP protocol. Basically, it includes all the messages defined in ICMP, and five additional messages for neighbour discovery.

New address types

Unicast addresses: These addresses resemble most IPv4 addresses. A packet sent to a unicast address is received by one (and only one) interface assigned to that address. A unicast address unambiguously defines an interface (link-local, site-local or global) and thereby a network host.

Multicast addresses: Multicast addresses replace IPv4's broadcast addresses. Just like anycast addresses, the multicast addresses are also assigned to a group of interfaces. However, packets sent to a multicast address are received by all hosts of the group.

Anycast addresses: Anycast addresses are assigned to groups of interfaces, possibly belonging to different hosts. A packet sent to an anycast address is received by only one of the interfaces belonging to the group, usually by the one that is located nearest to the sender.

Streamlined Header Format

IPv6 also streamlines and enhances the basic header layout of the packet. Compared to IPv4, some of the headers were dropped and others were made optional. Omitted fields include:

Fragmentation: Fragmentation can now only be done by end stations of a route. IPv4's method of fragmenting datagrams in intermediate hops requires resources and processing efforts in routers unreasonably.

Options: IPv6 defines the new "Extension header" mechanism instead of IPv4's header options. Extension headers carry optional header information, and are generally not processed by intermediate routers, which contributes greatly to faster end-to-end delivery.

Header checksum: Calculating header checksums requires a lot of processing time and offers little advantages in today's Internet. Up-to-date network technologies have considerably low error rates, while error checking and correcting functions are included in other layers as well. This makes calculating checksums at the network layer unnecessary.

The new header contains only 8 fields compared to the 14 fields of the IPv4 header. It also has a fixed size, which is a further milestone in increasing processing speed. One important additional field is the 20 bit long Flow Label that is responsible for QoS support for traffic flows. Due to the simplified header structure, the total IPv6 header size is only twice as large as the IPv4 header, even though 16-byte IPv6 addresses are four times longer than the 4-byte IPv4 addresses.

Header extension

Header extensions are located between the IPv6 header and the transport layer header, and contain optional network layer information. Usually only the destination host processes them, but some of them require hop-by-hop processing. These extensions immediately follow the base IPv6 header, preceding other extensions. Thus, intermediate routers only have to investigate the first part of a packet, which simplifies processing in comparison with IPv4, where the header length is variable depending on the included options. An IPv6 packet can carry several extension headers.

Mobility

Mobile hosts connected to the Internet via a wireless interface are likely to change their point of access frequently. A mechanism is required that ensures that packets addressed to moving hosts are successfully delivered. During handover, packet loss may occur due to delayed propagation of new location information up to the Home Agent. These losses should be minimized in order to avoid he degradation of service quality as handover become more frequent. Mobility management can be divided into two parts: micro- and macro mobility. Macro mobility handles interdomain handovers, while micro mobility responsible for intradomain handovers (see Figure 1.).

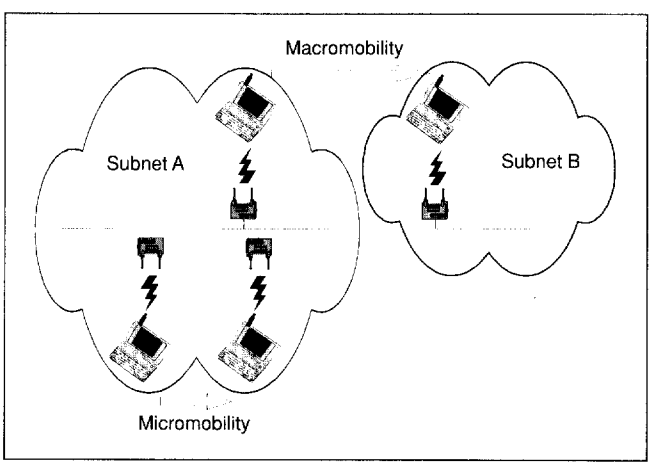

Figure 1. Micro- and macromobility

To improve performance, the frequent handoffs (due to small radio cells) inside a given domain – also so called intra domain handovers – are handled locally by the micro mobility protocols. The role of micro mobility protocols is to hide user movement from the mobile IPv4 or IPv6 protocol, by handling user mobility locally, fast, and simple inside the micro mobility domain. IPv6 or Mobile IPv4 are responsible for wide area mobility support, called macro mobility.

The Mobile IP protocol is considered to have limitations in its capability to handle large numbers of mobile stations moving fast between different radio cells. The handover frequency should typically not exceed once a second. However, Mobile IP is well suited for interconnecting disparate cellular networks effectively providing global mobility. Resulting from this fact, several micro mobility approaches have been proposed within the IETF, which are supporting mobility in a well-defined area. Such as the two most discussed micro mobility protocols: HAWAII and CIP.

The mobile IPv6 (and the mobile extension of IPv4 as well) basically solves the macromobility problem. The basic idea is the following: let us separate the identification and routing role of IP addresses! In the mobile IP each node has a static, so called home address, which identifies the device. This address remains unchanged while roaming. Devices leaving their home network get a temporary, so called care-of address in every foreign network, which topologically belongs to the given network, thus the mobile device remains reachable in the foreign network using this care-of address.

The terms introduced by Mobile IPv6 are the following:

- *Home address:* The IP address assigned to a mobile node within its home link.
- *Care-of address (CoA):* A temporary IP address assigned to a mobile node while visiting a foreign link.
- *Binding:* The association of the home address and a care-of address of a mobile node, along with the remaining lifetime of that association.
- *Mobile node (MN):* A node that can change its point of attachment to the Internet, while still being reachable via its home address.
- *Home Agent (HA):* A router on a mobile node's home link with which the mobile node has registered its current care-of address. While the mobile node is away from home, the Home Agent intercepts packets on the home link destined to the mobile node's home address, encapsulates them, and tunnels them to the mobile node's registered care-of address.
- *Access Point:* A router in the foreign network, which ensures the visiting mobile's connection to the network, through wired or radio interface.
- *Binding Cache (BC):* A conceptual data structure for storing bindings. The Binding Cache should be implemented by all IPv6 nodes.
- *Header extension:* A header extension, which can be sent along with arbitrary (even empty) packet and which contains complementary information, e.g. for handling mobility.
- *Binding Update (BU) message:* A header extension, which contains the current binding and the lifetime of the binding of the sending mobile node.
- *Binding Request (BR) message:* A header extension, in which a communication partner can ask the mobile node to send its current address.

- *Binding Acknowledge (BA) message:* A header extension, which the Home Agent acknowledges the reception of the Binding Update message with.
- *Home Address extension:* Thus the mobile node usually sets up the sending address to the foreign address while sending packages, this extension serves for telling the addressee its identifying home address.
- *Binding List (BL):* This list contains those Binding Update messages that were sent by the mobile node.

Every mobile device in IPv6 can always be addressed with its home address. When the mobile device is not attached to its home network, it obtains a temporary IP address – a care-of address – from the foreign network it is currently attached to. In order to be able to receive packages in this case the mobile always informs its Home Agent – a router in its home subnetwork – about its current care-of address. Correspondent nodes can send packages directly to the care-of address if they know it, otherwise they send them to the home address and the Home Agent forwards them to the mobile. The association between the home address and the care-of address is called binding. In IPv6 networks, every node contains a so-called Binding Cache to store binding information about mobile devices. If the correspondent node uses the home address and the Home Agent forwards the packet to the user, this routeing is called triangled routing. This of course overloads the network.

With the limited capability of mobiles and network overhead caused by triangle routing the optimization of the Binding Cache's size and the binding entries' lifetimes is very important. Our simulation demonstrates this issue in different network scenarios. We investigate different statistics like end-to-end delay time, rate of packets sent via triangle routing, rate of packet loss, handover frequency, etc.

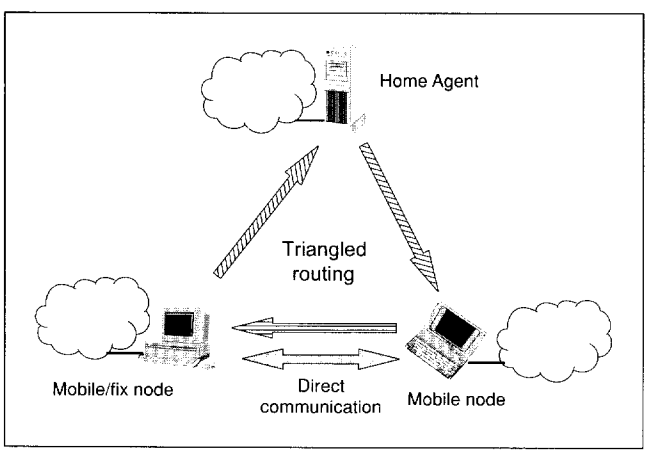

Figure 2. Triangled routing and direct communication

The simulation environment

We have developed a simulation environment to prove our concepts of Mobile IPv6 under OMNeT++. OMNeT++ (Objective Modular Network Testbed in C++) is a free, open-source discrete event simulation tool, similar to other tools like PARSEC, NS, or commercial products like OPNET. Our Mobile IPv6 model can be freely downloaded along with many other models.

Modules of the macro mobility environment

Our simulation environment deals with the IPv6 Mobility Extension, especially with the binding management methods. With the simulator, we can easily build different network scenarios by providing a few simple parameters from which the simulator constructs the network automatically.

According to OMNeT++, the structure of our simulator is modular. We defined the modules and their connections in the NED language, and implemented their functions in C++. The modules are the following:

Mobile: This component represents a mobile device, which changes its location and speed periodically, and sends data requests to servers and other mobiles, as well as receives data from these.

Air: It represents the radio interface, but now it simply connects Mobiles to the wireline network. There is only one Air module, because OMNeT++ can not handle dynamic connections properly.

Access Point: These elements represent all physical radio access points belonging to the same subnet. Macro mobility handovers happen between these Access Points, micro mobility handovers happen inside an Access Point.

Router: This component stands for the whole wired network between Access Points, Servers and Home Agents. It is responsible for routing packets and simulates network delays as well.

Server: Common Servers generate data packets as a reply to Mobile data requests.

Home Agent: Home Agents implement the mobile extension management by maintaining the binding between a Mobile's home and foreign address.

The modules are connected to each other according to the following figure (Figure 3.)

Simulation results

As seen before the Binding Cache has a very important role in mobility support. We simulated how the Binding Cache size and the bindings stored in the cache effect the ratio of the triangled packets. In the network we had 50 mobile nodes, 9 subnets, 5 servers and 7 Home Agents.

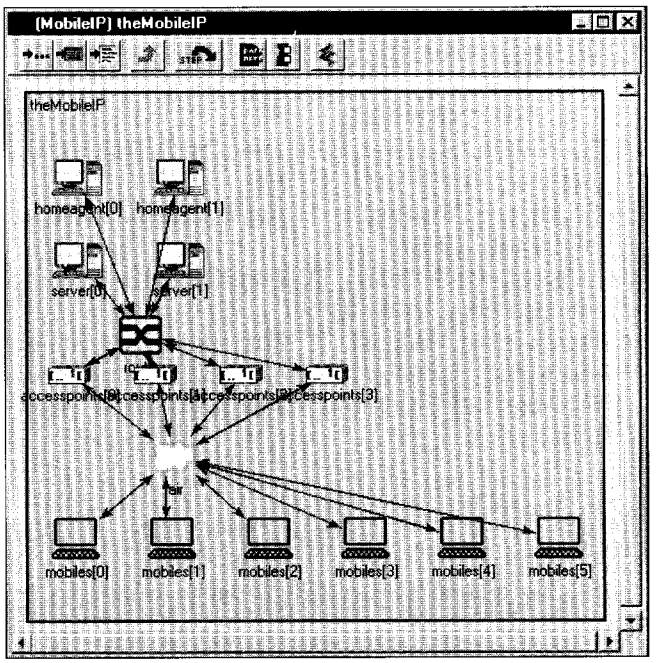

Figure 3. The Simulator

In our case two types of packets could be delivered on the triangled route, that is why we tracked these packets in the network:

Data request: the ratio of the triangled data request to the total number of data request packages. Every packet contain the Binding Update extensions the reply for these packets are delivered on the direct route.

Spontaneous data packets: in this case a remote mobile node or a server send a packet to the mobile. Here also the ratio of the triangled data packets to the total number of data packets are measured.

Examining the Binding Cache's size

By running our simulator with different Binding Cache sizes between 1 and 55 (the number of mobile devices is 50 in this case), we came to the following results seen on Figure 4.

It can be seen that increasing the cache's size linearly decreases the rate of the triangled packets. This is because bigger Binding Cache can store more binding. It is self evident as seen on the figure, that when the Binding Cache is smaller, more data packets are triangled, and this leeds to an overload in the network In case of a big Binding Cache, the size of the portable devices were increased, but this is not good in the mobility point of view.

Examining the binding entries' lifetimes

By running our simulator with different entry lifetimes between 0 and 100 seconds, we came to the following results, as seen on Figure 5.

It can be seen that by increasing the lifetimes first quickly decreases the rate of the triangled packets,

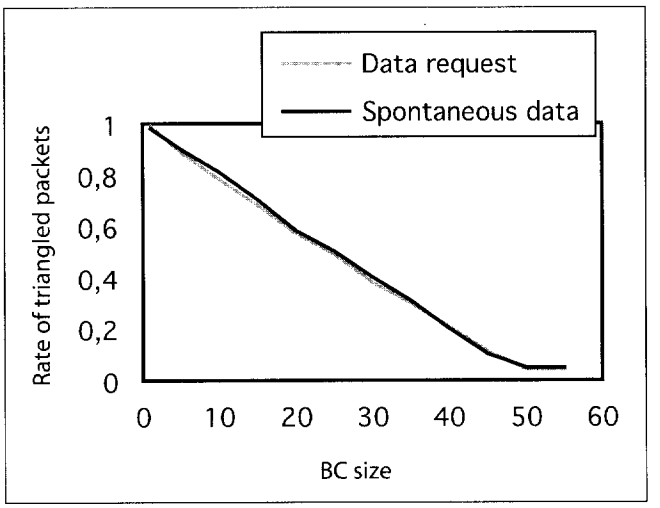

Figure 4. The effect of BC size

until it reaches the fifty percent level. (The Binding Cache's size in this scenario was 25, that is, the half of the mobile's number.) When the lifetime is 0, all the packets are delivered on the triangle rout. If we start to increase the lifetime, the ratio of the triangled packets

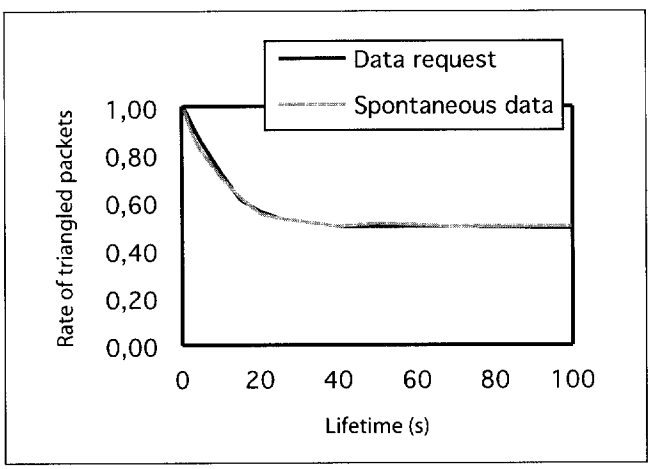

Figure 5. The effect of BC entries' lifetime

is decreasing, until a point, while at this point the new entries are replacing the old ones. In this particular case this number at around 20-25 seconds.

Summary

In our article we introduced the functions of the IPv6 needed for the mobility handling. To inspect these functions we wrote a general IPv6 simulator. We analysed the effect of the Binding Cache sizes and the lifetime of the bindings on the network performance. With our simulator the optimal Binding Cache size and lifetime can be calculated.

The simulator is capable of simulating mobile terminals operating at willing IP environments. The

simulator still contains some simplifications that is way we want to improve it to be used to design real IPv6 based mobile networks.

References

1. David B. Johnson: Mobility Support in IPv6, Intenet Draft, 2000
2. Charles E. Perkins: Mobile IP – Design Principles and Practices, Addison-Wesley, 1998
3. Charles E. Perkins: Route Optimization in Mobile IP, Intenet Draft, 2000
4. Thomas Eklund: IP version 6 – The next Generation Internet Protocol, 1996
5. Preetha P. Kannadath and Hesham El-Rewino: Simulating Mobile IP Based Network Enviroments, University of Nebraska at Omaha, 2000.
6. Susan Thomson, Thomas Narten: IPv6 Stateless Address Autoconfiguration, RFC 2462, 1998
7. Thomas Narten, et. al.: Neighbor Discovery for IPv6, RFC 2461, 1998
8. R. Hinden, S. Deering: IP Version 6 Addressing Architecture, RFC 2373, 1998
9. S. Kent, R. Atkinson: IP Authentication Header, RFC 2402, 1998
10. S. Kent, R. Atkinson: IP Encapsulating Security Payload, RFC 2406, 1998
11. Derek Lam, Donald C. Cox, Jennifer Widom: Teletraffic Modeling for Personal Communications Services, IEEE Communications Magazine, 1997. február

News

The Bluetooth Special Interest Group (SIG), an association of over 2000 technology companies, has just resolved a standar for the use of ISDN over Bluetooth. Bluetooth devices such as PCs PDAs and GSM telephones gain, for the first time, unlimited access to ISDN data and telephone communications services.

The new standart defines the communication between the so-called ISDN Clients and the ISDN Access Points using the international norms ETSI 300 838 and GSM 07.08. Wireless ISDN communication over Bluetooth is possible with the full ISDN data transfer rate and a coverage of 100 meters and more

„The CIP version 0.95 technical specifications are presently available at www.bluetooth.org/specifications.htm. The Bluetooth SIG issues first of all the version number 0.95 to all officially adopted and published Profiles.

CIP is modeled directly on the basis transport protocol L2CAP can be used parallel to PAN (BNEP), DUN (RFCOM) and CTP (TCS, SCO). CIP-capable end devices can therefore avail of all ISDN capabilities and functions including acting as a network gateway (with compression), support telephony and relevant CTI applications or simply connecting computers. As the ISDN software interface CAPI has been integrated, all computer programs such as telephony and multimedia applications, answering machines, fast Internet over network access using single or multiple channel access are possible, without using any cables. Additionally when using CIP a PC or a PDA can dial via ISDN directly to a host and swap data. The highly versatile ISDN interface CAPI can now be used for Bluetooth telephones or headsets.

ISDN applications based on the CAPI interface have proven themselves in a variety of different fields over the past years. Because of the CAPI specifications software from manufacturer A is compatible with the hardware from Manufacturer B. This principle is also valid for ISDN applications over Bluetooth.

BlueFRITZ! can be used with applications from other manufacturers.

AVM will release the CMTP source code to facilitate the fast and widespread usage of ISDN over Bluetooth. This source code allows for the integration of the CMTP protocol as the basis of CIP conveniently into existing Bluetooth structures under Linux.

Using the Bluetooth protocol stack under Linux it will be possible to use CIP-capable end-devices and it will allow the use of existing CAPI based ISDN applications for data and speech without modification under Bluetooth.

BlueFRITZ! products support CIP as a central Profile, the BlueFRITZ! products update policy guarantees the implementation of the newest standards. In the same manner all other important protocols will be made available