An Architecture for connecting Ad hoc Networks with the IPv6 Backbone (6Bone) using a Wireless Gateway[1]

Nico Bayer[a], Bangnan Xu[a], Sven Hischke[b]

[a] T-Systems, Technologiezentrum, Am Kavalleriesand 3, D-64295 Darmstadt, Germany
[b] Deutsche Telekom AG, Friedrich-Ebert-Allee 140, D-53113 Bonn, Germany
Email: Nico.Bayer@iem.fh-friedberg.de; Bangnan.Xu@t-systems.com; Sven.Hischke@telekom.de

Abstract **: This paper describes the system design and implementation of a testbed, running at T-Systems in Darmstadt. In this testbed the interworking between mobile wireless ad hoc networks (MANETs) and infrastructure based networks, based on the IPv6 protocol is investigated. This means, access to the internet via a multihop wireless network is demonstrated. In normal cases ad hoc networks are autonomous, without any possibility to get access to other networks. To solve this challenge a "wireless gateway" has been developed that works as an interface between the ad hoc network and infrastructure based networks. In this paper the testbed and its basic protocol architecture are described. The main part of the paper is focused on the major challenges that had to be solved, to realise the demonstrator: wireless gateway, wireless gateway discovery, IPv6 address autoconfiguration and Mobile IPv6 within the MANET.**

Keywords**:** AODV6, IPv6, ad hoc, wireless gateway, Mobile IPv6, address autoconfiguration, Wireless LAN, MANET, testbed

1. Introduction

In future communication systems mobile ad hoc networks (MANETs) will play an important role. Ad hoc networking does not require any infrastructure. So it is possible to establish spontaneous communication between network-enabled electronic devices (e. g. mobile phones, PDAs, Laptops). In areas where only local communication around the sender is required ad hoc networking has major advantages compared to "conventional" wireless systems, such as GSM (Global System for Mobile Communications) and UMTS (Universal Mobile Tele-communications System). For example, MANETs are ideal for establishing an instant tactical communication network needed by emerging military applications such as situational awareness systems for manoeuvring. Other MANET applications are: wireless networks for disaster recovery operations, wireless home and office area networks.

A lot of applications do not need only a communication within a local area covering from an ad hoc network but also global connections to the Internet. One scenario could be that members of a conference have configured an ad hoc network to exchange their data with other conference members. But for another application they also need a connection to the internet. For such a scenario interworking between the ad hoc networking protocols and the protocols used in infrastructure based networks is needed. But one fundamental problem of ad hoc networks is that they have no possibility to reach other networks. Communication is limited to communication with other ad hoc nodes in the own network domain.

This paper tackles this fundamental problem and addresses the interworking between ad hoc and infrastructure based networks. The view is focused on the IPv6 protocol. To achieve the connection between the two kinds of networks, a special node called wireless gateway is used. This wireless gateway works as an interface between the two kinds of networks and contains both protocol stacks.

The connection of the both networks is demonstrated in an IPv6 testbed at T-Systems in Darmstadt. To realise this testbed, some challenges had to be solved. The major challenge was the realisation of the wireless gateway. Another challenge was the gateway discovery mechanism. This mechanism provides relevant gateway information to the ad hoc nodes to set up a route to the gateway and get access to other networks. The ad hoc nodes also need a global routable IPv6 address to be reachable from outside the ad hoc network. The address autoconfiguration of IPv6 can handle this task. It is also desirable that the ad hoc nodes are reachable under only one address. So it is necessary that the Mobile IPv6 protocol will be used within the ad hoc network.

Basis for the connection of the ad hoc network with an infrastructure based network is the used AODV6 (Ad hoc On-Demand Distance Vector for IPv6) implementation. All extensions needed to realise the special functions and mechanisms for the connection are added to the AODV6 source code.

The rest of this paper is organised as follows: Section 2 gives an overview of the testbed. Section 3 describes the protocol stack, the function und the mechanisms of the wireless gateway. Section 4 investigates the gateway discovery. It also presents the results of data throughput measurements. The address autoconfiguration is described in Section 5. The last Section handles the advanced mobility of the ad hoc nodes by using the mobile IPv6 protocol within the ad hoc network.

2. Testbed Description

The testbed consists of an ad hoc network and an in-

[1] This work is partly supported by the German Ministry of Education and Research under IPonAir Project

frastructure network. Both are running the IPv6 protocol.

The infrastructure based network is built up hierarchically and comprises 4 routers, several subnets and a few nodes. One of the routers has connection to the global 6Bone, so it is possible for the nodes to connect to the “IPv6 Internet”. The RIPv6 (Routing Information Protocol for IPv6) protocol is used to exchange routing information between the routers.

The ad hoc network consists of some laptops. These ad hoc nodes use Wireless LAN Cards, configured for the ad hoc mode to communicate with each other. To realise a multihop communication the AODV6 routing protocol is used within the ad hoc domain.

A wireless gateway has been developed, to enable an ad hoc node in the ad hoc network to access global networks. So it is possible for ad hoc nodes to communicate not only with other ad hoc nodes within the own network domain but also to set up a communication to external nodes. External nodes may be nodes within the infrastructure based network, the global 6Bone or nodes, which are part of other ad hoc networks.

To set up a route to an external node, the ad hoc nodes use the “gateway discovery” mechanism to get information about the location of the wireless gateway and the way to reach it. Two different mechanisms of gateway discovery (reactive and pro-active) are realised within the testbed.

An overview of the testbed is shown in Fig. 1.

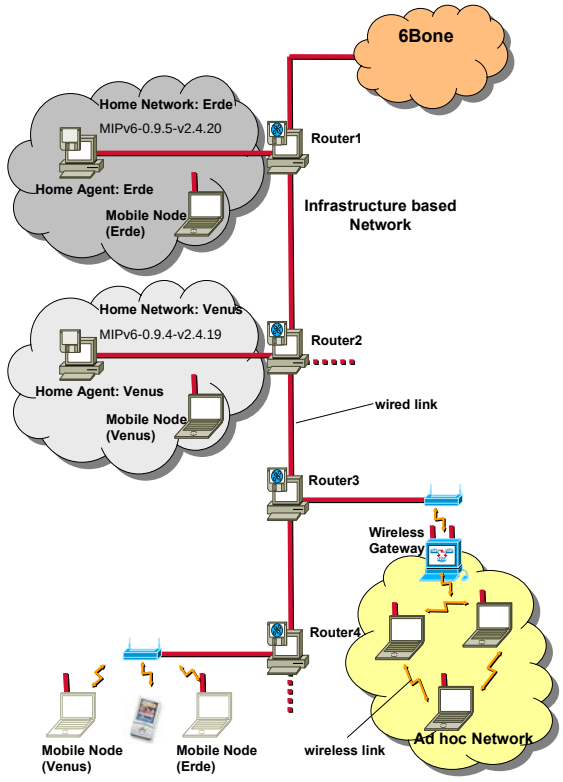

Fig. 1: Testbed

Fig. 1 shows a group of mobile nodes that build a wireless ad hoc network. The communication between the nodes within the ad hoc network is established through wireless multihop paths if no direct wireless link is available. The wireless gateway is connected to the infrastructure based network via one of his WLAN cards that is associated with an access point. This access point has a wired connection with router 3. The gateway’s second WLAN card is configured to communicate with the ad hoc nodes. Thereby the ad hoc network has a connection to the 6Bone through the infrastructure based wireless access network.

To be reachable from internet hosts, the ad hoc nodes need addresses that fit into the hierarchy of the testbed and can be correct routed to the ad hoc network. Therefore the ad hoc network uses flat routing. This means, that there are no subnets within the ad hoc network and all ad hoc nodes use the same subnet prefix. Thereby the ad hoc nodes can also decide between ad hoc nodes and external nodes.

To demonstrate the interworking of ad hoc and infrastructure based network, some IPv6 applications have been installed, e. g. video conferencing, video streaming and web browsing.

3. Wireless Gateway

To connect the ad hoc network with the infrastructure based network, an interface is needed. There are several concepts how to realise this interface. In this testbed the connection is realised with a special node called wireless gateway.

The wireless gateway is the interface between the ad hoc network and the infrastructure based internet.

In order to be able to communicate with both network types, the gateway needs protocols of the fixed internet and the wireless ad hoc network. On the internet side, it needs the usual internet protocols. On the ad hoc side, it has to send and receive packets using the AODV6 routing algorithm. The different protocol stacks are shown in Fig. 2.

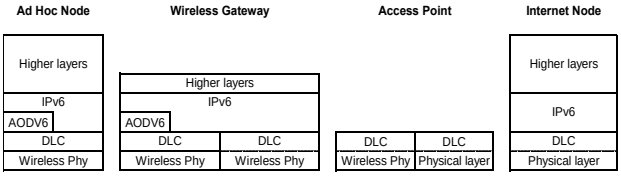

Fig. 2: protocol stack wireless gateway

Ad hoc nodes use a protocol stack that is similar to the protocol stack used in internet nodes. The main difference between these two stacks lies in the network layer. Ad hoc nodes use the AODV6 routing protocol to route packets. In the physical and date link layer, ad hoc nodes run protocols that have been designed for wireless channels. In this demonstrator the IEEE 802.11 WLAN standard is used.

To realise two different protocol stacks in the gateway, it needs two interfaces. Therefore the gateway has two WLAN cards. The first one is configured for the infrastructure mode and has the protocol stack of normal internet nodes. With this card the gateway is able to connect to an access point. This access point is the connection between wireless and wired communication. It converts from wireless to wired and the other way around. The other card is configured for the ad hoc mode and has the ad hoc protocol stack.

To configure a WLAN card you need to set some important parameters. The “wireless_mode” is needed

to set the card for the infrastructure (value: managed) or ad hoc (value: ad-hoc) mode. The "ESSID" (Extended Service Set Identifier) parameter is the identifier of a WLAN network. With this identifier it is possible to ensure, that the node always connects to the correct wireless LAN, if there are other WLANs reachable. For the infrastructure card you have to configure the same ESSID as in the access point. The ad hoc card must be configured for the ESSID that is valid in the ad hoc network.

The other values "channel/frequency" and "data rate" should be set to automatic.

The wireless gateway is also central part of the gateway discovery (Section 4) and the address autoconfiguration mechanism (Section 5).

It also handles the routing of packets between the ad hoc network and the infrastructure based network. Basis for the development of the wireless gateway is the AODV6 implementation. All extensions needed for the realisation of the gateway are added to the AODV6 source code.

4. Gateway Discovery

AODV6 is an ad hoc on demand distance vector routing protocol. It uses RREQ (Route Request) and RREP (Route Reply) messages to search for routes to other ad hoc nodes. This protocol is used to realise multihop within the ad hoc network. Periodic HELLO messages are used to determine connectivity. A detailed description of the AODV6 protocol can be found in [5].

In this demonstrator the HUT (Helsinki University of Technology) AODV6 implementation is used as basis. The implementation works with an internal routing table. This routing table stores information about the neighbours, old routes and active routes. Furthermore the Kernel routing table provided by the operating system is used to store all active routes.

The gateway discovery mechanism is used by the ad hoc nodes, to get information about the gateways IPv6 address and the way to reach it. With this information it is possible for ad hoc nodes, to set up a route to the gateway and get access to other networks outside the ad hoc network. The gateway discovery can be initiated by the gateway (proactive gateway discovery) or by the ad hoc node (reactive gateway discovery). Both of them are implemented in the testbed.

In the proactive gateway discovery method the gateway periodically sends HELLO messages that contain a special option called PROAGW option. This option is new and especially designed and developed for the proactive gateway discovery. It had to be implemented into the AODV6 implementation. This option has information about the gateways IPv6 address and how to reach it. This message can be received by all nodes, within the gateways transmission range. The multihop environment needs an extension to this approach: the receiving ad hoc node also adds the PROAGW option to its HELLO messages, as long as it has valid gateway information. If a node did not receive the PROAGW option for a while, it marks its stored gateway info as invalid and does not add the PROAGW option to its own HELLO messages anymore. In normal cases, the ad hoc nodes do not add any option to the HELLO messages. To realise this, the ad hoc nodes and also the gateway have been extended with new mechanisms and functions. With these extensions the gateway information can also be received from ad hoc nodes that are beyond the range of the gateway. With this method, every ad hoc node has gateway information every time and can always build up a route to the gateway. The proactive gateway discovery produces a lot of overhead but the delay is very small because the ad hoc node already has all information to set up a route to the gateway.

The reactive gateway discovery only provides the gateway information if the ad hoc node requests them to get access to external networks. Therefore the ad hoc node sends a GWSOL (gateway solicitation) message via broadcast into the ad hoc network. If the wireless gateway receives this message, it replies via Unicast with a GWADV (gateway advertisement) message, that contains the gateway information. The reactive gateway discovery has a high delay if the ad hoc node wants to connect to an external network. Therefore it produces only a little overhead. GWSOL and GWADV messages are added to the AODV6 implementation. The GWSOL is a slightly different RREQ message. A new Flag called I-Flag (Internet-Flag) signals that this message contains to the reactive gateway discovery and can only be answered by the gateway. The GWADV message is a slightly different RREP message and also contains the I-Flag. But it also has some additional options. These options are needed by the ad hoc node to set up a route to the gateway, e. g. gateway address and lifetime. To fill the GWSOL and the GWADV message with the correct information some new functions had to be added to the implementation. The implementation also had to be extended that the ad hoc nodes understand these two messages and process the information.

4.1. Determination between internal and external nodes

Another problem is how do the ad hoc nodes and the gateway know if it is an external or internal node? If an ad hoc node wants to find a route to another node, it must know if this node is internal or external. This information is very important because the node must know if it has to send a RREP for an internal or a GWSOL message for an external node. To solve this, the nodes compare the subnet prefix of the destination address with the ad hoc's subnet prefix. This means that all addresses that have the ad hoc's subnet prefix (ad hoc address) belong to ad hoc nodes and all other addresses belong to external nodes.

One disadvantage of this method is that signalling within the ad hoc network only works with ad hoc addresses. Further developments of the testbed will also realise signalling with non ad hoc addresses, like for example with the home address of a mobile node.

4.2. Processing gateway information

The next challenge was to store the gateway information inside the ad hoc nodes. Regardless of what gateway discovery method is used, the information is processed and stored in the same way. Fig. 3 shows

how the internal routing table should look like, if the ad hoc node has valid gateway information.

Destination Address	Next Hop Address
Default	Gateway
Gateway	MN_A

Fig.3: internal routing table

The first entry is the default entry and contains the gateway address as next hop. The second entry is the next hop to the gateway. It gives information about the next hop on the route to the gateway. If a source node S wants to communicate with the fixed node FN, S looks for these two entries in its internal routing table and adds the following entry into the kernel routing table:

This entry shows that S has an active route to FN.

Destination Address	Next Hop Address
FN	MN_A

Fig. 4: kernel routing table

All packets addressed to FN can use this route to reach the destination.

4.3. Performance measurements

Bandwidth measurements should give information about the differences in relation to data rate between the reactive and proactive gateway discovery. Therefore the environment shown in Fig. 5 was used to do the measurements.

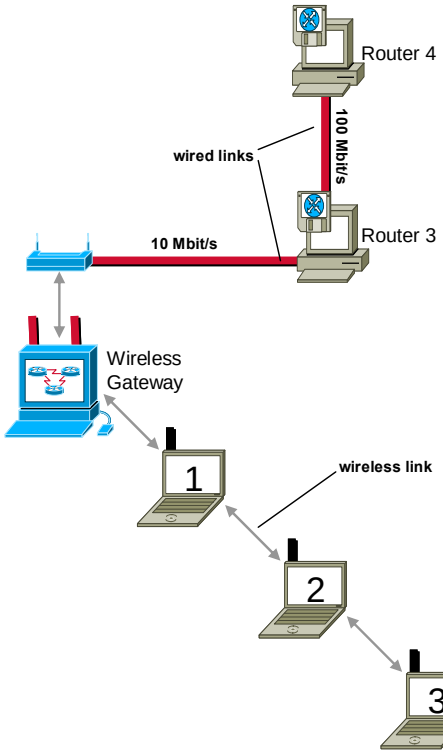

Fig.5: measurement environment

Only TCP (Transmission Control Protocol) traffic was used to measure the data rate in relation to the packet size and the number of nodes between sender and gateway. To realise the following environment MAC filters are used to tell every node which packets it has to ignore, e. g. ad hoc 2 ignores all packets with the gateway's MAC address as source address. So it has only direct connections to ad hoc 1 and ad hoc 3.

The measurements were progressed in the following way: The ad hoc nodes send one after another TCP traffic to router 4. The packet size was a parameter and set to the values 1500 Byte, 1024 Byte, 768 Byte, 512 Byte and 256 Byte. Every measurement took 30 seconds.

Fig. 6 shows the results of the reactive measurement. Depending on the packet size the throughput varies in a way that small packets are causing more access to the wireless media than large packets do. Furthermore the relation between data and overhead is less than with bigger packets. The number of nodes between sender and gateway means, that every packet allocates the wireless media not only once but as often as it is forwarded on the way to the gateway. All ad hoc nodes use the same channel (frequency), so they have to share the 11Mbit/s bandwidth (shared media).

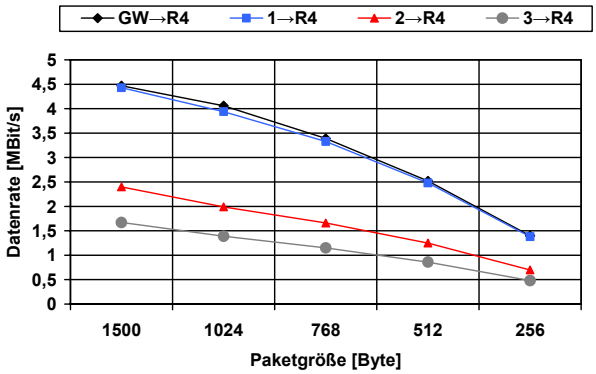

Fig.6: Performance with the reactive gateway discovery

In Fig. 7 you can see the result of proactive measurement.

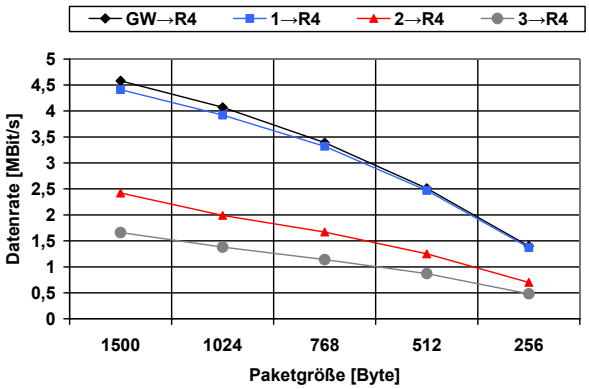

Fig. 7: Performance with the proactive gateway discovery

Comparing the two measurement results, it is obvious that they are not very different. The data rate with reactive gateway discovery is nearly the same as with proactive gateway discovery. So in our environment it does not matter whether the reactive or proactive method is used.

Using the proactive gateway discovery every node adds a PROAGW option to its HELLO messages. Every PROAGW option has a size of 32 Byte. Every node sends one HELLO per second. This means we have an additional overhead of 4*32Byte/s (128 Byte/s). This does not really affect WLAN environments with 11 Mbit/s of total bandwidth. Furthermore the ad hoc network in the testbed is very static, the ad hoc nodes do not move. So there is no additional overhead caused by changing environment. In larger networks, with hundreds of ad hoc nodes and more movement, this result is different. But in small ad hoc networks like our testbed, we will not notice any performance difference between the two gateway discovery methods. Simulations with larger ad hoc networks and a huge number of ad hoc nodes that move around will have different results than our environment.

For further developments the reactive method will be more important than the proactive one. Because reactive gateway discovery fits better into the idea of AODV.

5. Address Autoconfiguration

The address autoconfiguration is important for the ad hoc nodes to generate a global routable IPv6 address. This address is needed, in order to be able to establish communications from outside. But there are three different concepts that handle this issue.

The first method is based on the stateless autoconfiguration mechanism defined in IPv6 [9]. In this mechanism the ad hoc node uses its link-local address to send a router solicitation message to the all router multicast address. If a router receives such a message, it answers with a router advertisement that contains all information to create a global routable IPv6 address on the ad hoc node. This mechanism does not work in multihop networks because link-local addresses are not applicable for multihop communication. So the stateless autoconfiguration must be slightly modified. Instead of using the link-local address the ad hoc nodes generate a temporary side-local address on start-up, to be able to receive the router advertisement message in a multihop environment. Furthermore the ad hoc node must not send the router solicitation message to the all router multicast address but to any special gateway multicast address that works with multihop. This could be a new address called all gateway side local multicast address.

The second method is based on the stateful autoconfiguration method that uses DHCPv6 (Dynamic Host Configuration Protocol for IPv6) [10]. In this case the gateway is configured as a DHCP server. With modified signalling the ad hoc nodes can request an address from the gateway (DHCP server). To realise this for multihop, the stateful autoconfiguration must be modified in a way, that the nodes do not use their link-local address but a temporary side-local address. It is the same problem as mentioned above. The side-local address can also be generated on boot. The ad hoc nodes also must not send their request to the all DHCP server multicast address. This address does not work in multihop networks. To solve this problem an already mentioned all gateway side local multicast address should be used.

The last method uses the AODV6 messages to request an IPv6 address from the gateway. The ad hoc nodes can use a special RREQ message to ask the gateway for relevant information to create a global routable address. The gateway answers with a special RREP message that contains this information.

6. Mobile IPv6 in Ad hoc networks

The described extensions enable an ad hoc node to reach external networks and be reachable from external nodes. Therefore it needs to configure a global routable address. Once such address is available, global mobile-initiated sessions, such as web browsing or DNS queries, can be used. A topologically correct address in the IP header source field is sufficient for packets sent from the ad hoc node in such sessions.

To provide an always-on reachability from the fixed internet, the ad node needs a permanent address. This address can be used as a Mobile IPv6 [8] home address. In such a case, reachability can be provided even when the node moves between different ad hoc networks and different points of attachment.

A mobile node should use Mobile IPv6 when it is not on its home link and registers with its home agent using a globally routable address from the visited network.

If a mobile node moves to an ad hoc network, it uses its home address for the address autoconfiguration signalling, described in Section V.

The mobile node then uses the globally routable address acquired from the wireless gateway as its care-of-address when possibly performing a home registration. If no home registration is needed, the mobile node is at home in this ad hoc network and the prefix of its home address belongs to its wireless gateway.

In a foreign ad hoc network the mobile node has two addresses. It can use the care-of address for global communication. The care-of address identifies the current location of the mobile node. Only with this address it is possible, that packets from the internet can be routed to the mobile node. The care-of address can also be used for ad hoc communication, e. g. address autoconfiguration, route discovery and gateway discovery. The home address can only be used for ad hoc communication because it identifies the mobile node but not its location.

7. Conclusions

This paper considers the internet access for mobile ad hoc devices via a wireless gateway. It describes the challenges that had to be solved to realise the interworking of this two networks. The functionality of the wireless gateway is described. Furthermore the two different gateway discovery mechanisms that give the ad hoc nodes the possibility to build up a route to the gateway and get access to external networks are explained. It also displays the results of data rate measurements, which compare the proactive und reactive gateway discovery.

Topics for further development include the address autoconfiguration and the mobile IPv6 protocol within the ad hoc network. The address autoconfiguration handles the problem of how the ad hoc nodes could generate a global routable IPv6 address. Mobile IPv6 within the ad hoc network makes it possible for the ad hoc nodes to be reachable under the home address.

REFERENCES

[1] Jin Xi, Christian Bettstetter. "*Wireless Multihop Internet Access: Gateway Discovery, Routing, Addressing*". Technical University Munich, Institute of Communication Networks.

[2] Alex Ali Hamidian. "*A Study of Internet Connectivity for Mobile Ad Hoc Networks in NS2*". Department of Communication Systems, Lund Institute of Technology, Lund University.

[3] R. Wakikawa, J. T. Malinen, C. E. Perkins, A. Nilson and A. J. Tuominen. "*Global Connectivity for IPv6 mobile ad hoc networks*". Internet Draft, November 2001, Work in progress.

[4] C. E. Perkins, E. M. Royer and S. Das. "*Ad hoc on demand distance vector (AODV) routing*". Internet Draft, March 2001, Work in progress.

[5] C. E. Perkins, E. M. Royer and S. Das. "*Ad hoc on demand distance vector (AODV) routing for IP version 6*". Internet Draft, November 2000, Work in progress

[6] D. B. Johnson and C. E. Perkins. "*Mobility support in IPv6*". Internet Draft, July 2000, Work in progress.

[7] S. Thomson and T. Narten. "*IPv6 stateless address auto-configuration*". RFC 2462.

[8] D. Johnson and C. Perkins. "*Mobility support in IPv6*" (work in progress). Internet Draft, March 2001.

[9] S. Thomson and T. Narten. „*IPv6 Stateless Address Autoconfiguration*" IETF RFC 2462, December 1998.

[10] R. Droms, J. Bound, Bernie Volz, Ted Lemon, C. Perkins, and M. Carney. "*Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*". Internet Draft, November 2002.

[11] B.Xu, S. Hischke, E. Weiss. "*Integration of Ad hoc Networks with Wireless Access Networks using Ad hoc Gateways*". In Proc.WWRF9, Zurich, Juni 2003

[12] B.Xu, S. Hischke, B. Walke. "*The Role of Ad hoc Networking in Future Wireless Communications*". In Proc. ICCT, Beijing, 2003