Ambient Networks Gateway Selection Architecture

Mikko Majanen, Kostas Pentikousis and Jukka Mäkelä
VTT Technical Research Centre of Finland
Kaitoväylä 1, FI-90571 Oulu, Finland
Email: {firstname.lastname}@vtt.fi

***Abstract*— Many anticipate a future wireless world filled by a multitude of user devices and wireless technologies. Effective management of this kind of heterogeneous, mobile, and rapidly changing ad hoc networks will be a challenging task. We present and evaluate the Ambient Networks Gateway Selection Architecture (GSA), which provides support for gateway discovery, management, and selection for mobile nodes within dynamic routing groups. A routing group (RG) is a cluster of nodes in physical proximity, aware of the group membership, with a common goal of optimizing mobility management and routing functionality in the group. A gateway is a mobile node that provides packet relaying and connectivity services to other nodes in the RG. GSA can be also used outside the Ambient Networks architecture, and we present how it can be used with two existing mobility management protocols, namely Mobile IP and Host Identity Protocol, especially in the case of moving networks. Our simulation studies show the benefits gained from group formation when compared to same functionalities implemented in every individual node. We also compare the GSA hybrid signaling strategy with proactive and reactive approaches; the simulation results show that the hybrid approach scales better when the routing group size grows.**

***Keywords*—Ambient Networks, gateway selection, Host Identity Protocol (HIP), mobile computing, Mobile IP (MIP), mobility management, moving networks, routing group**

I. Introduction

Many anticipate a future wireless world filled by a multitude of user devices and wireless technologies. Effective management of this kind of heterogeneous, mobile, and rapidly changing ad hoc networks will be a challenging task. The Ambient Networks project [1] addressed this challenge by developing innovative network solutions based on the dynamic composition [2] of networks providing access through the instant establishment of inter-network agreements. The Ambient Networks concept [3] includes the Ambient Control Space (ACS) [4], which provides common control functions to a wide range of different applications and access technologies, enabling the integrated, scalable and transparent control of network capabilities.

Mobility management, a key component of Ambient Networks, can be defined as the set of functions that allow a communications system to adapt itself, seamlessly and optimally, to changes in physical and logical topology of the network. A goal of the Ambient Networks mobility solution is to provide a framework within which existing mobility solutions can be deployed and interoperate, whilst ensuring that new mobility solutions can be added as and when they become available. Novel mobility concepts (e.g., see [5]) have been developed to better support moving groups of nodes and users, such as personal area networks and networks formed in mass transport vehicles, such as commuter trains.

Within Ambient Networks, nodes in a moving network can be linked to form a cluster referred to as the Routing Group (RG) [6]. Let us clarify the distinction between the terms *cluster* and *routing group*. Take a set of mobile nodes, U, and a set of base stations, B, connected to the wired network. Each $b_j \in B$ can provide wireless connectivity to all nodes $x \in U$ within its coverage area. A cluster, $S \subseteq U$, is defined as the set of nodes from U that can (i) communicate with each other, (ii) are physically close to each other and, (iii) are likely to remain so. Although (i) and (ii) can be determined using information from layers 1–3, (iii) can be determined only by taking into consideration other situational and context information. Identification and formation of such clusters can enable communication and shared use of applications, while several other optimizations, related to routing and mobility management can be pursued.

In each cluster one node is elected to act as the *cluster head*. Each cluster head is aware of the cluster topology, including the nodes and their roles. Within each cluster, one or more nodes can act as gateways, relaying packets for other nodes and providing connectivity to other networks. A routing group (RG) is defined as the set of nodes $R \subseteq S$, in which the nodes are aware of group membership. This allows even more possibilities for optimizations than a cluster.

In previous work [7], the Gateway Selection Architecture (GSA) was introduced to provide support for gateway identification, management, and selection within a routing group. Of course, one might expect that by grouping nodes and delegating mobility management to the cluster head and the gateways certain performance optimizations are possible as discussed in [7]. Later, the performance of the GSA was evaluated by simulations in [8]. In this paper, we willl elaborate GSA, provide a detailed description of GSA and present performance evaluation results, delivering for the first time a complete coherent view of GSA.

The paper is organized as follows. In Section II we take a look at related work in the area of mobility management. Section III describes the GSA architecture and Section VI shortly compares GSA to other related work and discusses the possible benefits of GSA. In Sections IV and V we describe how the GSA architecture can be used outside the Ambient Networks framework with existing mobility management protocols, namely Mobile IP (MIP) and Host Identity

Protocol (HIP). Section VII describes the simulation scenario and results and Section VIII concludes the paper. Table I lists the acronyms used throughout the paper for easy reference.

II. Mobility Management

In the context of mobile/wireless networks three approaches have been followed with respect to gateway discovery. The first is a *proactive* strategy whereby the gateways broadcast advertisements to the whole network. The nodes requiring gateway services choose the most suitable gateway based on the advertisements they received. In *reactive* strategies, the initiative lies with the nodes, which broadcast request messages to the network and select the most suitable gateway based on the replies that are unicasted to them. In *hybrid* strategies, gateway advertisements are usually broadcasted only to the nodes “near” the gateway. For instance, the advertisements may have a limited time to live (TTL) value, say, three hops. Nodes farther than this amount of hops have to use request messages to receive gateway services. That is, if a node x does not receive a broadcasted advertisement from any gateway, it will broadcast a gateway request message.

Gatewaying can be seen as a service, so the gateway discovery problem is similar, to some degree, with the general service discovery problem. The Service Location Protocol (SLP) [9] is an IETF protocol for service discovery and advertisement. There are three entities in the SLP: service agents (SAs), user agents (UAs) and directory Agents (DAs). SAs advertise the service to the network or to DAs, UAs try to find services for the applications. DAs cache the information about available services based on SAs’ advertisements.

The most popular way to provide Internet access to nodes within ad-hoc networks and in mobile networking scenarios seems to be extending the Mobile IP (MIP) protocol for either IPv4 or IPv6 networks. In the following subsections we briefly go through the basics of MIP and study how it has been extended to work with moving networks.

A. Mobile IP

In MIPv4 [10], the base station (BS) nodes act as Home (HA) and Foreign Agents (FA) for the mobile nodes. The HA keeps a list of mobile nodes that are attached to it, i.e. the mobile nodes that belong to the same subnet as the HA. When the mobile node moves away from the HA, it eventually starts using another BS as its network connection point. The new BS will act as FA for the mobile node and it provides a care-of address (CoA) from its subnet address space for the mobile node. The CoA is transmitted also to the mobile node’s HA, which establishes a tunnel between the HA and FA. Tunneling means that packets destined to the mobile node are forwarded from the HA to the FA using IP-in-IP encapsulation [11]. The FA decapsulates the packet and transmits it to the mobile node, which is currently in its subnet. Thus, the HA and FA nodes (i.e. BSs) act as gateways for the mobile nodes.

In MIPv4, the HAs and FAs advertise themselves by broadcasting periodically beacons, i.e. a proactive approach is adopted. However, if the mobile node does not have a connection to any BS, it may broadcast a solicitation message to find one. BSs that receive the solicitation message will reply by sending the beacon packet. Thus, MIPv4 supports also the reactive approach, even though it mainly relies on the use of proactive approach. The beacons are not forwarded; MIP supports only one wireless hop.

In MIPv6 [12], the mobile node has the FA functionality built in. When the mobile node is outside its home network, it sends a binding update to its Home Agent informing its current care-of address. It may also send the binding update to its correspondent node if that supports MIPv6. In that case, packets from CNs may be routed directly to the mobile node’s care-of address, without going via the HA. In addition to the optimal route, the overhead is also smaller since instead of IP-in-IP encapsulation, IPv6 routing header can be used.

B. Extending MIP to multi-hop ad hoc networks

Since MIP supports only one wireless hop, several approaches have been presented to extend MIP to make Internet connections available for the ad hoc network nodes that do not have a one hop route to the FA. Sun et al. [13] present an architecture where MIP is combined with the Ad hoc On-Demand Distance Vector (AODV) protocol [14] and a reactive approach to solicit FA advertisements is used. Ratachandani et al. [15] on the other hand use a hybrid approach where the FA advertisements are flooded within a limited number of hops from the FA; nodes outside this hop limit use reactive approach.

The simulation studies in MIPMANET (Mobile IP for Mobile Ad Hoc Networks) [16] show that it is highly valuable to be able to choose the closest access point to the Internet since it reduces the overall load in the moving network. In the scenario used in [16], broadcasting the MIP FA advertisements was found better than unicasting them to each MIP using node inside the moving network. Unicasting the advertisement meant in this case that the FA unicasted the advertisement to every moving node that was registered with it. The broadcasting approach provided better options for mobile nodes to change the FA to a better one since the advertisements were broadcasted periodically. In the unicasting approach, the mobile nodes used solicitation messages when they did not have a connection to any FA.

Lee et al. [17] propose a hybrid GW advertisement scheme for connecting ad hoc networks to the Internet. In that approach, Dynamic Source Routing (DSR) [18] is used as the ad hoc routing protocol. Unnecessary flooding of GW discovery packets is avoided by using advertisement schemes based on the mobility and traffic patterns of the moving network.

Ghassemian et al. [19] present a performance comparison between proactive, reactive and a hybrid GW discovery approaches. In the hybrid approach, the GW advertisements’ time to live was limited, and nodes further away had to use a reactive approach to solicit advertisements. In the scenario considered, the proactive approach performed best in case of packet delivery ratio and the packet delay. The reactive approach performed worst and the hybrid one was between these two. On the other hand, with respect to signalling overhead, the reactive approach was better than the proactive one.

C. Network Mobility (NEMO)

In all previous approaches, the GW nodes, i.e. the BSs, are stationary, so they are not moving. Also, all nodes in the moving network perform mobility management actions independently.

The Network Mobility (NEMO) Basic Support Protocol [20] extends MIPv6 to manage network mobility. A similar protocol has been proposed also for IPv4 moving networks in [21]. NEMO enables reachability and session continuity for all nodes belonging to the moving network. With NEMO, mobility is transparent to the moving network nodes. This is achieved by introducing a special Mobile Router (MR) node that connects the moving network to the Internet. The MR binds a network prefix with a care-of address (CoA) indicating its current location together. MR uses binding update messages to inform its current CoA to its HA. Nodes within the moving network are allocated an address from the MR's prefix. Thus, they can connect to the Internet without having to participate in the mobility management since the MR updates the HA for the whole network, not just for itself. Traffic destined towards the moving network (i.e. MR's network prefix) is intercepted at the HA and tunneled to the MR using IP-in-IP encapsulation. MR decapsulates the packets and forwards them to the correct mobile node. In the opposite direction, reverse tunneling is used, i.e. packets are tunneled from the MR to the HA, and then directed towards the correspondent node. The NEMO Basic Support Protocol does not support route optimizations to correspondent nodes.

NEMO solves the basic problem of network mobility but, since it is MIP-based, it has some disadvantages inherent to MIP: MR introduces a single point of failure on the routing path, tunneling adds overhead, and the routes are not optimal (so called dog leg routes) since the binding updates to CNs are not supported.

D. MOCCA

The Mobile Communication Architecture (MOCCA) [22] is designed for inter-vehicular systems that consists of vehicular ad hoc networks, road-side Internet Gateways (IGWs), and a proxy between IGWs and the Internet. MOCCA uses a modified version of Mobile IP (called Mobile IPv6*) to support the mobility of vehicles. The Proxy maintains the vehicles home agents (HAs), IGWs function as foreign agents (FAs) and the vehicles represent the Mobile Nodes (MNs). The Correspondent Node (CN) in the Internet sends its data packets to the MNs home address (i.e. the HA in the Proxy). The Proxy tunnels them to the FA, which decapsulates and forwards them to the MN. The Proxy also separates the transport layer end-to-end connections in order to prohibit e.g., TCP connections to time out.

Since MIP does not support multi-hop ad hoc networks (the MN may be more than one hop away from the IGW), MOCCA employs a modified version of the Service Location Protocol [9] for discovering the IGWs. In MOCCA, the Service Agent is located on the IGWs and it announces periodically its Internet access service. The service advertisements are geocasted, i.e. broadcasted in a geographically restricted area. The Directory Agent is located in the vehicles. It extracts the information from the service announcements and caches it to a local database. The advertisements include information about the current number of clients using the IGW, IGW's available bandwidth, geographical position and optionally some other information. The User Agent (i.e. mobile device) within the vehicle queries the database and configures Mobile IPv6* to use one of the available IGWs as its FA. In case multiple IGWs are available, the UA selects the IGW that fits best to the requirements of the applications. The selection is made based on a fuzzy logic algorithm that predicts QoS parameters like expected delay, dropouts and the probability of disconnection for the connection.

The MOCCA implementation covers both network and transport layer protocols. As such, it does not support the mobility of legacy applications running on devices inside the car. For supporting legacy applications (without modifications to those), MOCCA includes also another proxy inside the vehicle. This proxy hides the device from the Internet, so it is not reachable outside the vehicle. However, the device can access Internet services. To be reachable outside the vehicle, the device should additionally support MIPv4, too.

E. Host Identity Protocol

The problems in MIP-based mobility are based on the TCP/IP stack architecture. The IP address is used for both identifying the host and stating the host's current network attachement point, i.e. the location of the host. When the host is moving, it has to change its attachement point, which means changing its IP address. On the other hand, the transport layer connections are bound to certain IP address and port. Keeping the transport layer connections while moving requires also update on the transport layer.

In the Host Identity Protocol (HIP) architecture [23] the host identifiers and locators are separated. A new layer is introduced between transport and network layers. Transport layer connections are not anymore bound to IP address and port, instead a Host Identifier (HI) is used. IP address is used only for forwarding packets. This allows new possibilities for mobility and multihoming, as described e.g., in [24]. HIP-based mobile routers, like [25] and [26], are especially interesting from the moving network support perspective.

III. Gateway Selection Architecture in Ambient Networks

Both proactive and reactive gateway discovery schemes have their pros and cons. A reactive approach does not create unnecessary traffic, but on the other hand, it exhibits longer delays. A proactive approach comes with smaller delays, but introduces possibly unnecessary traffic. On the other hand, ability to select the most suitable gateway is worth of some

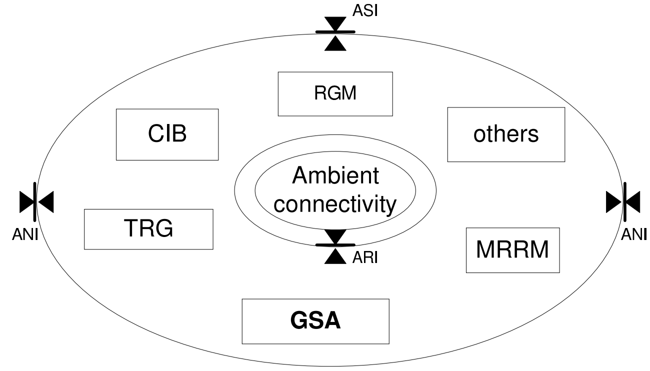

Fig. 1. GSA as part of ACS

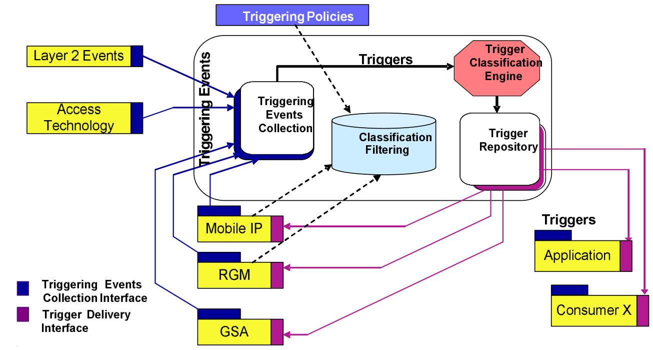

Fig. 2. TRG components

extra traffic, as argued in [16]. Proactive approach suits better for this purpose since the status of the gateway is updated periodically in the advertisements and the new gateways can be discovered earlier.

GSA adopts a hybrid approach for gateway discovery, introducing a special kind of nodes called gateway selectors (GWS). In GSA, service advertisements and requests are unicasted to the gateway selectors, thus simplifying information dissemination and updates regarding gateway (or more enhanced mobile router) nodes and their capabilities. This should not only decrease the amount of signaling overhead, but also allow the majority of the nodes to have only limited computational capabilities and battery power by keeping the intelligence in the gateway selectors. By introducing GWS nodes, GSA borrows a little from the Service Location Protocol (SLP) [9], with GWSs resembling to Directory Agents, gateways to Service Agents, and other RG nodes to User Agents.

As illustrated in Figure 1, GSA is part of the ACS and it is supported by many other ACS functional entities such as triggering (TRG) [27], [28], Routing Group management (RG) [6], context information management (CIB) [29], [30], and multi-radio resource management (MRRM) [31], which are capable of providing a wealth of information related to gateway discovery and selection. GSA is designed to utilize this extra information aiming at making optimal gateway selections.

Figure 1 shows also the three interfaces that are used to access the ACS functionalities. The Ambient Service Interface (ASI) is used by higher layer applications and services to issue requests to the ACS concerning the establishment, maintenance and termination of the end-to-end connectivity. The Ambient Resource Interface (ARI) is used for managing the connectivity plane resources such as routers, switches, and radio equipment. The Ambient Network Interface (ANI) is used for transferring information between different Ambient Networks.

The triggering (TRG) functional entity is a vital part of the ACS since it informs the other functional entities about different events in the Ambient Networks. Main elements of the TRG, as detailed in [28], are the entities which create events (producers) and the entities that use the trigger information (consumers). TRG collects the event information from various producers via a specific collection interface, processes the collected events and distributes the created triggers to the interested consumer entities. A producer, as well a consumer, can be any entity implementing the collection interface. In other words, the same entity can act both as a producer and a consumer. Figure 2 illustrates TRG with different producers and consumers.

TRG might have several event collectors, which may be distributed, collecting different types of events. A number of collectors might be needed since the producer might be the entity implemented in kernel space or an application in user space. Having a separate collector per producer entity with a dedicated inteface allows the communication between nodes with different operating systems as well.

In order to use the collection interface, producers need to register their triggers with TRG. By registering, each producer and their triggers can be identified and, further on, interested consumers can subscribe to get certain identified triggers. All this is a part of the processing mechanism that supports also the filtering of triggers. With filtering, consumers get only those triggers they are subscribed to. Using this filtering functionality together with the support for system wide policies, TRG can not only provide the way for efficient distribution of right triggers to the right consumers, but also provides a way to control consumer access to event sources.

Figure 3 shows the internal structure of the GSA functional entity. GSA uses TRG for implementing its signaling, i.e. the gateway advertisements and requests are transmitted as triggers. GSA includes a GSA Trigger Consumer for receiving triggers. Depending on the trigger received, its information may be stored to the GSA Parameter Collection or Policies data storage, and/or it may be further processed by the GSA Decision Engine. Based on the processing results, a new trigger may be generated and sent by the GSA TRG Producer. The actual behaviour of the trigger processing depends on the node's role, i.e. whether the node is a GW, GWS, or GW service user.

The gateway node's GSA Decision Engine uses the GSA Trigger Producer to periodically (or when needed) generate updates about its GW service status by sending a gateway

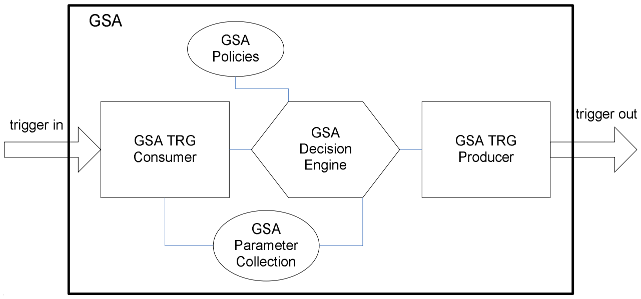

Fig. 3. GSA internal structure

advertisement trigger to TRG. The GW service parameters are maintained in the Parameter Collection data base. The GSA TRG Consumer subscribes to all triggers related to the node's context, RAN status, and so on. Policies may contain rules, such as whether the node is allowed to provide the GW service to other nodes.

An RG node starts the GW request process when its GSA Trigger Consumer gets a request trigger sent by an application. The application communicates with the ACS via the ASI interface. The GSA Trigger Producer creates a GW request message including service requirements, and sends it to TRG.

Subsequently, GWS's GSA Trigger Consumer receives the GW advertisement triggers and stores the status information of the gateway node to Parameter Collection. It also receives the GW request triggers from RG nodes that need GW service. GWS's GSA Decision Engine compares the service request parameters to the available gateways' parameters and selects the best match. The result is transmitted to the requesting node in a form of GW response trigger containing the address of the gateway and its GW service parameters. The actual algorithm to select the best GW is out of the scope of this paper but, for example, it can be a weighted sum over selected parameters (this approach is used e.g., in the selection of the cluster head node in [32]).

Usually, TRG is located on the same node as GWS, so the communication between TRG and GWS is node-internal and does not consume network resources. If the RG has also a cluster head, it is usually collocated also on the same node. If the cluster head (i.e. the RGM entity in Figure 1) or TRG is located at a different node than GWS, the information is then transmitted as triggers between the nodes. Thus, GWS's GSA Trigger Consumer also receives and GSA Decision Engine handles triggers dealing with e.g., topology changes in the routing group. In every case, GWS has always up-to-date information about the RG and its nodes. Actions (e.g., re-selection of the GW for certain RG nodes) are launched whenever deemed necessary.

Although GSA was designed originally to work within the Ambient Networks architecture, there are no reasons why GSA could not be used also outside Ambient Networks. In Ambient Networks the ACS binds the different functional entities together, but on the other hand, these entities, or the information they produce, may be used also separately. For example, TRG and GSA can be set up to any moving network; they are not dependent on any Ambient Networks architecture specific entities. Actually, TRG is the first step in the Ambient Networks migration plan [33]. Gateway selection related triggers are then perhaps produced by some other entities as in Ambient Networks, but still, GWS can make the decisions based on the information that is available. In fact, GSA can be used even without TRG; its principles can be easily applied to existing MIP and HIP implementations for mobility management optimizations for moving networks. In the following two sections we briefly explain how this can be done.

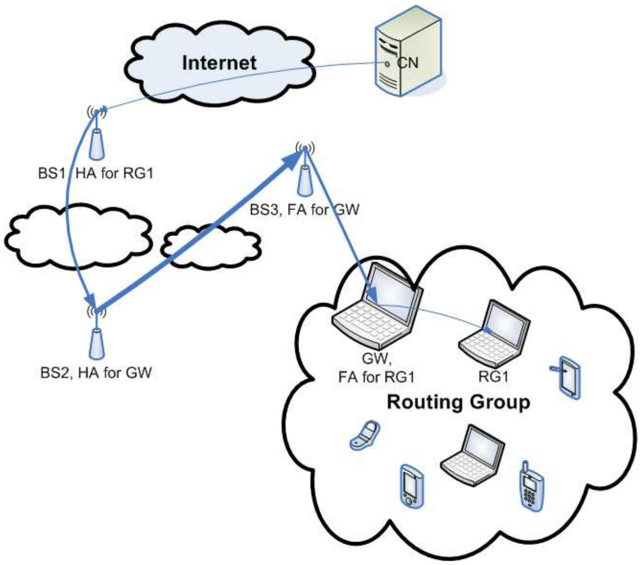

Fig. 4. Moving FA as a gateway for Routing Group nodes

IV. GSA WITH MOBILE IP

In MIPv4 [10], base stations act as Home and Foreign Agents (HA and FA, respectively) for mobile nodes. In moving networks, the gateway nodes can act as FAs for all nodes in the RG, forming a hierarchical set of FAs as illustrated in Figure 4. HAs are still located at the base stations. The gateway nodes use base stations as their FAs, so they handle their own mobility like normal mobile nodes in MIP. Alternatively, we can call these gateways as NEMOv4 Mobile Routers since this is how they work. The traffic destined to RG nodes goes via two HAs and two FAs before reaching the destination. The line width in Figure 4 illustrates the tunneling overhead between HAs and FAs.

The gateway discovery and selection process starts with the election of GWS. Normally this functionality lies with the same node as the cluster head. The cluster head collects and manages information related to the RG management. Gateway issues are part of this management so it is natural to include GWS functionality in the same node. The GWS and cluster head can be also on different nodes but that may add some more overhead due to the information exchange between them. The elected cluster head (and GWS) node informs the whole

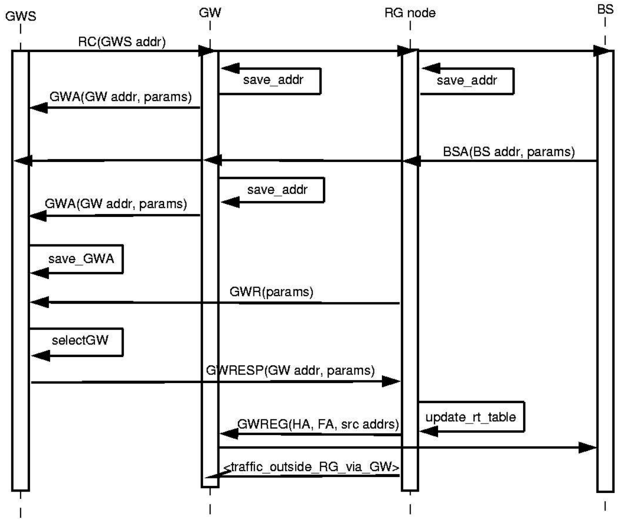

Fig. 5. GSA signalling in MIP-like mobility management

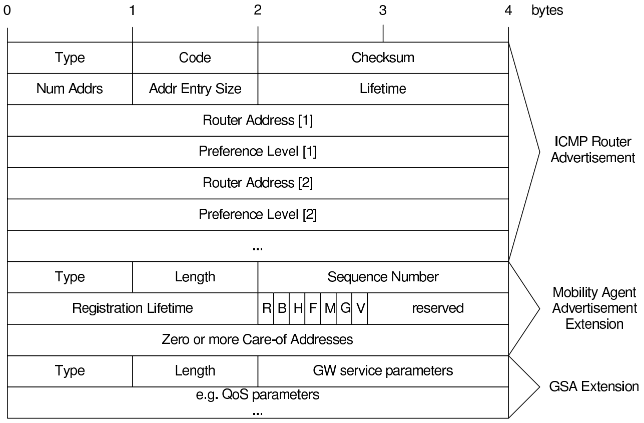

Fig. 6. ICMP Router Advertisement message header extended with MIP and GSA extensions

RG about its role by broadcasting a role claim (RC) message (see top part of Figure 5). RG nodes save the address of the GWS.

MIP makes use of ICMP [34] Router Advertisement and Solicitation messages. The ICMP Router Advertisement message is extended by a Mobility Agent Advertisement Extension that contains e.g., the care-of address(es) of the BS and the lifetime of the advertisement. The ICMP Router Solicitation message is used unchanged by MIP. GSA utilizes the same messages as MIP, but extends them by using optional TLV-encoded fields for providing extra information (e.g., QoS parameters), as depicted in Figures 6 and 7. Note that the messages are no longer broadcasted, as depicted in Figure 5.

Base stations (BS) broadcast periodically their own advertisements that contain the care-of address(es) of the BS and optionally some other information about the BS (e.g., QoS parameters). We call these extended MIP BS beacons Base Station Advertisement (BSA) messages. BSA messages are not forwarded inside the RG (they carry a TTL set to one). RG nodes that receive a BSA message and are willing to act as a gateway exploit the information contained in the BSA in forming a Gateway Advertisement (GWA) message describing the gateway service it can provide.

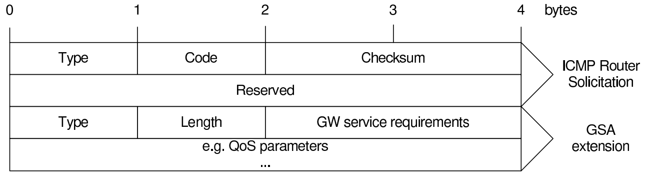

Fig. 7. ICMP Router Solicitation message header extended with GSA extension

GWA messages are unicasted to GWS by gateway nodes in response to the received BSA message. A GWA message is also sent as a response to a received RC(GWS) message, that is, when the RG is formed and the GWS is selected. The GWA message includes the address of the GW and its parameters, for instance, bandwidth, battery state of charge, supported radio access networks, and connection monetary cost (free of charge vs. charge based on traffic volume or connection duration), to name a few. In short, a GWA is another kind of MIP BS beacon, extended with some additional information just like BSA, but instead of broadcasting it with TTL=1 it is unicasted to the GWS (with TTL>1). So, in GSA, both the BS and GW nodes send extended MIP BS beacons (as illustrated in Figure 6).

The gateway selector saves the gateway's parameters to the list from the received GWA message (Figure 5, middle part). The RG nodes make a gateway request (GWR) to GWS when they need gateway service. The GWR message contains requirement parameters for the GW service (same as the GWA message). As depicted in Figure 7, it is a type of extended MIP solicitation message. When GWS receives a GWR message it browses through its list of gateways and selects the most suitable one. GWS replies with a response message (GWRESP) that includes the address of the selected GW and its parameters (a sort of extended MIP BS beacon). In case the node is not satisfied with the service chosen/available, it may cancel/postpone the connection or make a new request. Otherwise, it updates its routing table so that the traffic destined outside the RG is routed via the selected gateway. It also sends a registration message (GWREG) to its HA (as is the case in MIP).

V. GSA WITH HOST IDENTITY PROTOCOL

In the Host Identity Protocol (HIP) Architecture [23] hosts are identified by public keys (Host Identities), not with IP addresses. This helps in mobility and multi-homing issues since the nodes can change their IP addresses and still be reachable via the same Host Identity.

The HIP base exchange [35] allows any two HIP-enabled hosts to authenticate with each other and create a HIP association between them. The base exchange consists of four packets: I1 is the trigger packet sent by the Initiator to the Responder. I1 contains only the Host Identity Tag (HIT) of the Initiator and possibly the HIT of the Responder (if known). The second packet, R1, starts the actual exchange. It contains a puzzle, initial Diffie-Hellman parameters and a signature covering part of the packet. I2 contains the solution to the puzzle, Diffie-Hellman parameter for the Responder. The packet is signed. R2 is a signed message finalizing the base exchange.

Before starting the base exchange, the Initiator has to acquire the Responder's IP address. The HIP Rendezvous Extension [36] introduces a Rendezvous server (RVS) that serves as an additional initial contact point for its client HIP nodes. With RVS, the initial contact can be made by using RVS's IP address. RVS's clients become reachable via RVS's IP address. This is very beneficial in case of mobile nodes that change their network attachment point, and thus also their IP address, frequently. After the base exchange, the communication is based on Host Identities, even though the IP address changes have to be signalled to the peer hosts so that packets can be routed correctly at the IP layer. Address changes are made by sending an UPDATE packet containing the new location information.

The base exchange can also include information about available or requested services [37]. A HIP host capable and willing to act as a service provider includes also the REG_INFO parameter in its R1 packets, thus announcing its available services. The UPDATE packet can also be used for this purpose if new services become available after the HIP association has been established. To request registration with a service, a requester includes a corresponding REG_REQUEST parameter in an I2 or UPDATE packet.

There are two ways for HIP nodes to initiate the service discovery process [38]. In the so-called on-path service discovery a HIP node sends a Service Discovery Packet (SDP) towards the peer node in the Internet (for example its own RVS). Each host on the SDP's path that provides services responds with a Service Available Packet (SAP). SAP may contain information on all services it provides. Alternatively, in case the SDP requested only a particular service, only those services are included in the SAP. SAP also includes the R1 parameters. Thus, after receiving a SAP, the HIP base exchange can be completed with I2 and R2 messages; SDP corresponds to the I1 packet in this case. If the HIP node wants to search services available only on a certain network region, it may use different multicast addresses instead of the address of the peer node in the Internet.

In certain cases it is not feasible to use the on-path service discovery. The HIP hosts can then use the so-called passive discovery method. In this method, the HIP service providing nodes "sniff" passing HIP packets. If a packet fulfilling certain conditions is detected, a SAP can be created and sent to the HIP node that originated the matched packet.

GSA can be used also with HIP, especially in moving networks with HIP-based Mobile Routers. The same principles apply as with MIP: certain messages are extended with some additional information and the destination of some messages may be different than in normal HIP service discovery. All HIP packets contain a common header part and optional TLV-encoded parameters, as shown in Figure 8, so extending HIP packets with MR selection related extensions is straightforward. The signalling is depicted in the Figure 9.

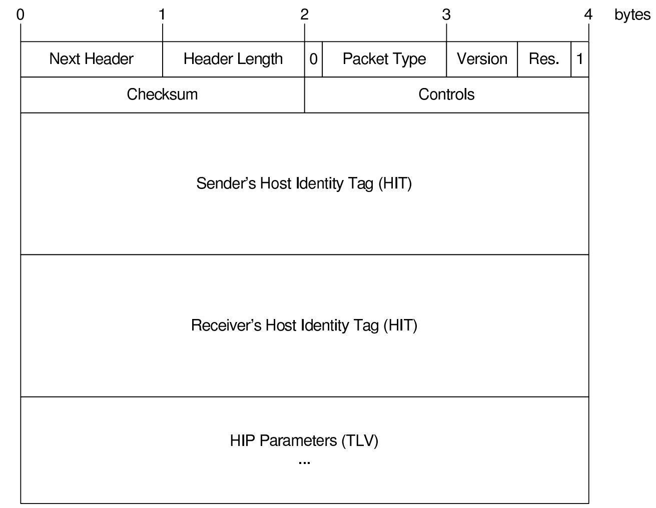

Fig. 8. HIP packet header format

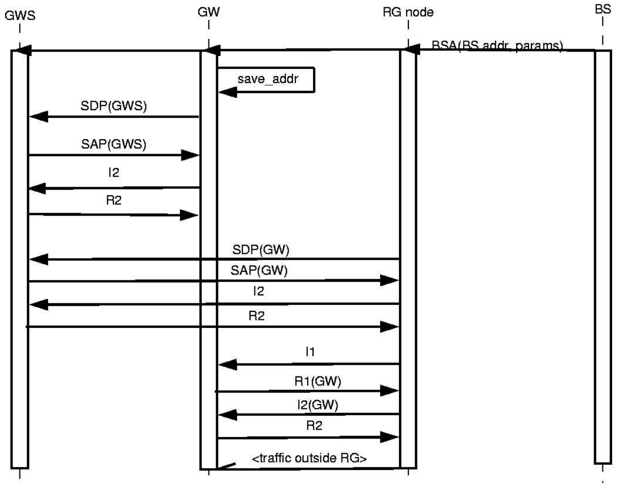

Fig. 9. GSA signalling in HIP-like mobility management

In case of HIP, GWS can be seen as a service. GWS provides the MR selection service. Nodes capable and willing to provide mobile router service register with GWS. Four-way base exchange extended with service discovery and registration information is needed for that at first time. The IP address of the GWS is known since it is signalled during the routing group formation process. After registration, MRs can use UPDATE packets as their MR service advertisements. HIP nodes requiring MR services can send their requests to the GWS, which replies with the best available MR for the requester's needs. After that the requester registers with the MR service and starts using it. UPDATE packets can be used in case of any changes regarding the service, e.g., location updates.

VI. Discussion

Of course, one might expect that by grouping nodes and delegating mobility management to the cluster head and the gateways certain performance optimizations are possible as discussed in [7]. The motivation for including the GWSs aims at simplifying the signalling overhead regarding gateways and their capabilities. By using GWSs, the topology for advertising and finding a GW is a unicasted star rather than the whole network flooded with messages. As a hybrid solution it has the benefits of both proactive and reactive approaches: the status of the gateways is known in the GWS all the time due to the advertisements, but the whole network is not unnecessarily flooded with them. The nodes requiring gateway service, request it from the GWS; the requests are not flooded to the whole network. New gateways are discovered as they become available, and GWS can direct the nodes to use them if they are more suitable for the nodes. GSA also allows the majority of the nodes to have limited computational capabilities and battery power because the intelligence is kept in the GWSs; thus, there is no need for spending so much resources (e.g., battery or computation power) in the other RG nodes.

A moving network such as a NEMO network can be seen as a RG with only a single MR. As identified in section II, dog leg routing is an issue in NEMO (or in general, MIP) based moving networks. This is especially true if there are nodes belonging to another home network as opposed to that of the MRs home network. In that case, all packets destined to or sent by such nodes need to go through two Home Agents, as illustrated in Figure 4. GSA is advantageous in this situation since the RG may have also other GWs as the MR. One GW could have the same home network as the other nodes in the RG and then the GWS, using context information, may direct them to use that GW instead of the MR. There might be also some other situations (e.g., load balancing) when the MR is not the best option for all RG nodes; in these situations GSA can provide for better GW selection results for the nodes.

Another drawback of the NEMO architecture is that the MR adds a possible single point of failure to the moving network. In GSA, GWS could be able to direct the nodes to use possible other GW nodes in case the MR fails. On the other hand, GWS nodes may fail in the GSA. This is why the RGs can have also secondary GWS nodes containing the same functionalities.

In the MOCCA architecture [22], the service agent inside the moving car caches the service advertisements sent by the road side gateways. Nodes inside the car ask the list of available gateways from it and select the most appropriate GW into use based on a fuzzy logic algorithm. So, MOCCA uses partly the same approach as GSA since the advertisements are gathered in one place and service users ask them from there. However, in MOCCA, and in the approaches that extend MIP to work in multi-hop ad hoc networks (such as [13], [15], [16] and [17]) the mobile nodes handle their mobility individually (as opposed to NEMO or HIP Mobile Routers) and make the GW decision by themselves. In GSA, the GWS makes the decision. The gateway or mobile router has to be moving along with the moving network so that mobility management can be hidden from the nodes in the moving network, which is the case in GSA architecture.

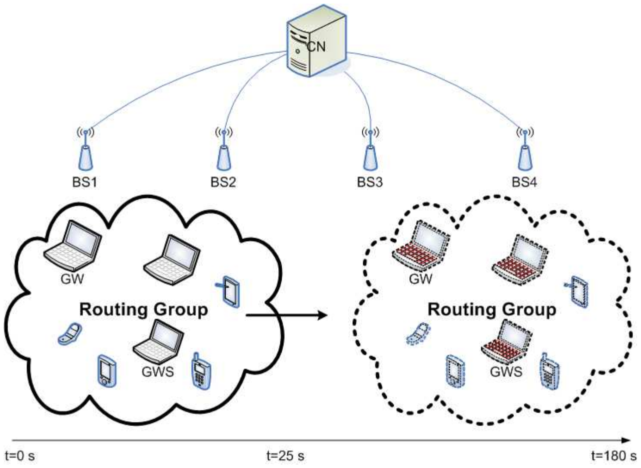

Fig. 10. The simulation scenario

VII. Evaluation

In the following two subsections we attempt to quantify the benefits of the GSA in simulation scenarios where several nodes move together in a mass transit vehicle.

A. Methodology

We use the ns-2 network simulator (version 2.28) [39] to evaluate GSA's MIP-like mobility management (as illustrated in Figure 4) in a commuter train scenario, as illustrated in Figure 10 (the figure is not to scale). We are interested in quantifying (a) the gains of GSA vs. standard MIP and (b) the advantage of GSA vs. general proactive and reactive strategies. The scenario includes a commuter train (total length=70 m, approx. 3 wagons; wagon width=3 m), and n passenger devices, which are randomly distributed inside the 210 m^2 area of the commuter train. For the purposes of this study, we configure only one mobile device to act as a gateway. The gateway functionality was implemented in ns-2 by adding the BS node's FA functionality to a mobile node, too.

During the first 25 s of the simulation, the mobile devices form a single, stable routing group. At $t = 25$ s the train starts moving at a constant speed of 11 m/s along a straight railway track. From $t = 25$ till $t = 180$ s the train passes by four base stations located along the railway track. The base stations are placed far from each other so that there is no coverage area overlapping. The first one, BS1, was configured to be the HA for all mobile nodes. The base stations were connected to each other via wired links and a wired node. The wired links have a bandwidth of 100 Mb/s with propagation delay set to 2 ms. The wired node also acted as a correspondent node to a mobile node, by sending constant bit rate UDP traffic to one of the mobile devices on the train. The IEEE 802.11 MAC layer data rate was set to 11 Mb/s, and it was used by all nodes in the train, using the free space signal propagation model and the

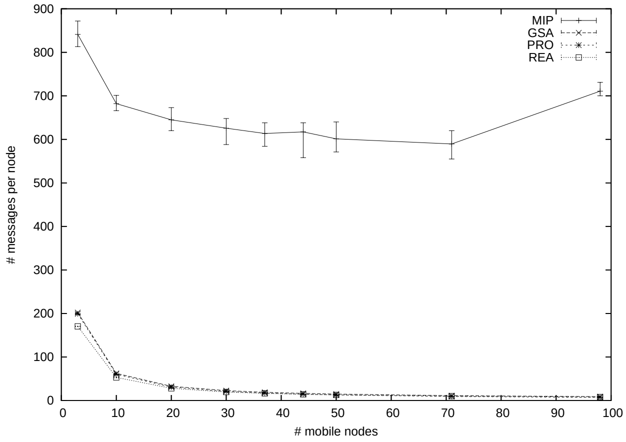

Fig. 11. Mean number of sent messages per node using MIP, GSA, proactive (PRO), and reactive (REA) approaches, excluding RG formation and routing protocol messages; error bars indicate min and max values

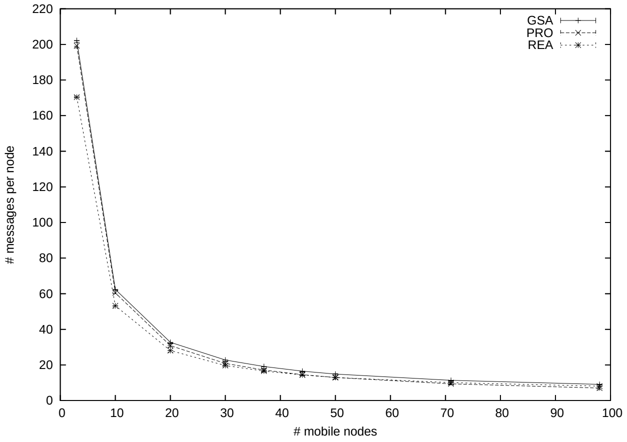

Fig. 12. Zoom of Figure 11 for GSA, proactive (PRO) and reactive (REA) approaches

DSDV routing protocol inside the RG. For RG formation and management we used the stability-based clustering protocol described in [32]. Unless mentioned otherwise, we run the simulation using the default values and settings in ns-2.

We evaluate the performance of GSA centering on the amount of required control signaling and compare it primarily with proactive and reactive algorithms, but also with the case where every mobile node manages its own mobility using MIP. As such, in all results reported below, we consider only the signaling required to provide the *same* functionality that MIP provides, and we exclude, for instance, routing group formation related signaling and DSDV messages. Further studies of MIP covering, for example, the effect of the velocity to the handoff, throughput and packet loss are presented in [40] and the references therein. The following subsection presents results from ten independent replications, for each of the scenario configurations presented above.

B. Results

First, we consider the number of sent messages per mobile node in either of the four alternative strategies. Figure 11 presents the mean number of sent control messages per mobile node. The error bars indicate the min and max values. The standard deviation σ varies between 9 and 24 with MIP and 0.02 and 0.6 in all other cases. Overall, as the routing group size increases, on average, nodes send fewer control messages. Clearly, forming a routing group is beneficial as compared to having each and every node use normal MIP to manage its mobility. The gains are typically an order of magnitude and increase as more nodes are added to the routing group. For example, in the case of $n = 3$, on average per node, MIP has to send more than four times the number of messages than GSA. At the other end of the range we explored, with $n = 98$ the difference is over 78 times. This is because there are many more nodes replying to BS advertisements and sending registration updates to HAs in MIP than in case of a routing group.

Comparing GSA with proactive and reactive approaches only (Figure 12), which also take advantage of group formation, we note that GSA underperforms. When employing GSA, on average, nodes have to send more messages than if they had used a proactive and reactive approaches. This is due to its hybrid strategy. When $n = 3$, using GSA nodes transmit 1.5% and 18.7% more signaling packets than when using proactive and reactive approaches, respectively. As n increases, the proportional difference between the number of messages sent by GSA and proactive approaches increases, reaching 29.1% when $n = 98$. Similarly, GSA's hybrid strategy underperforms the reactive approach as the number of nodes increases, although the proportional difference becomes smaller: with $n = 98$, GSA nodes send, on average 12.4% more messages.

This underperformance is due to the hybrid strategy GSA adopts. Both gateway nodes and routing group members send advertisements and requests, respectively, to the gateway selector. If a proactive strategy is used, then only the former messages are sent, while if reactive is opted for, only the latter messages are needed. Nevertheless, in GSA all messages are *unicasted* whereas in the proactive and reactive approaches, all messages are *broadcasted*, which forces mobile nodes to spend resources processing these messages regardless of whether they are useful in their current state. Broadcasting is a considerably "heavier" operation when compared to unicasting. Moreover, when employing the reactive approach, as implemented in the simulation model, it was not possible to re-select the gateway before the connection was lost (break-before-make handover). On the other hand, in both GSA and proactive approaches the status of the gateways is reported in the advertisement messages periodically, enabling seamless, make-before-break handovers and better load sharing, which may assist avoiding congestion incidents.

Figure 13 presents the amount of processed control messages and tells a similar story with Figure 11 with respect to the benefits of forming a routing group as opposed to using standard MIP. We refer to *processed control messages* as all

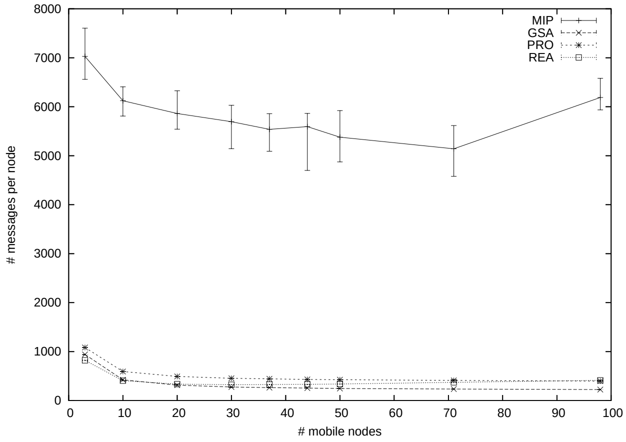

Fig. 13. Mean number of processed messages per node using MIP, GSA, proactive, and reactive approaches, excluding RG formation and routing protocol messages; error bars indicate min and max values

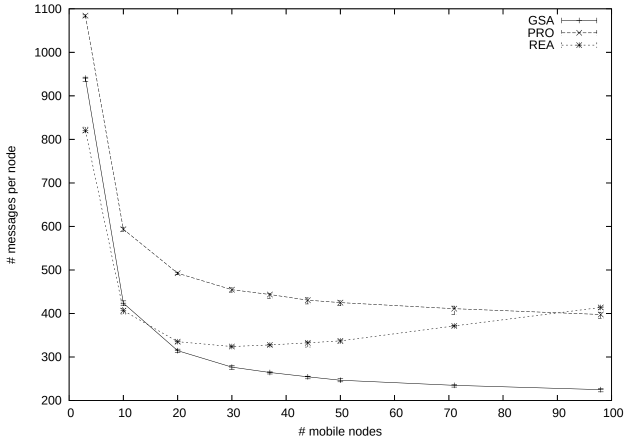

Fig. 14. Zoom of previous Figure 13

sent, forwarded, received and dropped packets handled above the MAC layer. As before, RG management and DSDV routing protocol messages were excluded, and so the average number of processed control messages is a good indicator of the total resource costs needed for gateway discovery and selection. MIP clearly underperforms the other three approaches. The standard deviation σ varies between 159 and 395 with MIP and 1.3 and 6.6 in all other cases.

The real gains when using GSA instead of proactive or reactive strategies are illustrated in Figure 14. First, GSA's hybrid signaling algorithm outperforms a proactive approach in all configurations. In fact the gains increase with n: for $n = 3$, on average, GSA nodes process 13.2% less control messages; with $n = 98$, they process 43.4% less messages. Second, GSA underperforms a reactive approach in small routing groups ($n \leq 15$). For $n = 3$, GSA nodes process, on average, 14.7% more messages (940.7 vs. 820.4). As n increases, GSA's relative performance against the reactive approach improves. With $n = 98$, GSA nodes need to process 50% less control messages than nodes using a reactive approach.

We note no other significant differences besides those mentioned above. Connection lost time between base stations was effectively the same in all scenarios, with small variations due to the locations of the nodes. The delay introduced by gateway discovery was not studied here because the gateway selection was triggered well before the GW service was actually needed. Nevertheless, we can say that GSA has a smaller (or equal) delay than reactive approaches, because the requesting node has to wait for only one response from the GWS; in reactive approach the node has to wait for a certain time in order to gather responses from all possible gateways and do the selection among those. Proactive approaches typically have very small delays—gateway selection occurs whenever needed among the saved advertisements.

VIII. Conclusion

We presented the Ambient Networks Gateway Selection Architecture, which manages the gateway discovery and selection mechanisms. We also showed how GSA can be used outside the Ambient Networks architecture, with Mobile IP and Host Identity Protocols.

We evaluated GSA with respect to sent and processed control messages in a moving commuter train scenario using simulation. The number of nodes in the train was varied between 3 and 98, which is quite large value for typical simulation studies, but on the other hand, also a representation of a real case on limit. We found that GSA has a considerable advantage over other alternatives. In particular, although GSA nodes transmit slightly more but unicasted control messages, as opposed to broadcasted control messages used in reactive and proactive strategies, GSA nodes need to process only 75% or less of the control messages processed using the alternative strategies, for medium-size routing groups. As an aside, we also verify the benefits from forming a routing group as opposed to having each node use Mobile IP independently. Our results show that GSA has more lightweight signaling than proactive or reactive approaches and that it scales much better as the routing group size grows.

Our future work includes the development of the gateway selection algorithm based on the parameters in gateway advertisements and requests. This includes also the definition of more detailed GSA extensions to ICMP Router Advertisement and Solicitation messages. With the selection algorithm we can study further the effects of selecting the most suitable gateway for the routing group nodes. The effects of the gateway discovery delay to service quality are also part of our future work.

Acknowledgment

Part of this work has been earlier published in [8]; the invitation to submit this extended version is highly appreciated. The simulation work was carried out in the framework of the Ambient Networks project, which was partially funded

TABLE I

List of acronyms

ACS	Ambient Control Space
ANI	Ambient Network Interface
AODV	Ad hoc On-Demand Distance Vector
ARI	Ambient Resource Interface
ASI	Ambient Service Interface
BS	Base Station
BSA	Base station Advertisement
CIB	Context Information Base
CN	Correspondent Node
CoA	Care-of Address
DA	Directory Agent
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing
FA	Foreign Agent
GSA	Gateway Selection Architecture
GW	Gateway
GWA	Gateway Advertisement
GWR	Gateway Request
GWREG	Gateway Registration
GWRESP	Gateway Response
GWS	Gateway Selector
HA	Home Agent
HI	Host Identity
HIP	Host Identity Protocol
HIT	Host Identity Tag
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGW	Internet Gateway
IP	Internet Protocol
MIP	Mobile IP
MIPMANET	Mobile IP for Mobile Ad Hoc Networks
MN	Mobile Node
MOCCA	Mobile Communication Architecture
MR	Mobile Router
MRRM	Multi-Radio Resource Management
NEMO	Network Mobility
PRO	Proactive
QoS	Quality of Service
RAN	Radio Access Network
REA	Reactive
RG	Routing Group
RVS	Rendezvous Server
SA	Service Agent
SAP	Service Available Packet
SDP	Service Discovery Packet
SLP	Service Location Protocol
TCP	Transmission Control Protocol
TLV	Type, Length and Value
TRG	Triggering
TTL	Time To Live
UA	User Agent
UDP	User Datagram Protocol

by the Commission of the European Union. Writing of this work was supported by TEKES (Finnish Funding Agency for Technology and Innovation) as part of the Future Internet program of TIVIT (Finnish Strategic Centre for Science, Technology and Innovation in the field of ICT). The views expressed in this paper are solely those of the authors and do not necessarily represent the views of their employers, the Ambient Networks project, the Commission of the European Union, TEKES, or TIVIT. The comments and ideas from people involved in the Ambient Networks project's mobility research and from the anonymous reviewers are gratefully acknowledged.

References

[1] N. Niebert (ed.), A. Schieder (co-ed.), J. Zander (co-ed.), and R. Hancock (co-ed.), *Ambient Networks: Co-operative Mobile Networking for the Wireless World*, Wiley, April 2007.

[2] N. Akhtar, C. Kappler, P. Schefczik, L. Tionardi, and D. Zhou, *Network Composition: A Framework for Dynamic Interworking between Networks*, Second International Conference on Communications and Networking in China (CHINACOM'07), August 2007.

[3] N. Niebert, M. Prytz, A. Schieder, L. Eggert, N. Papadoglou, F. Pittmann, and C. Prehofer, *Ambient Networks: A Framework for Future Wireless Internetworking*, IEEE VTC2005-Spring, June 2005.

[4] B. Ohlman, L. Eggert, M. Smirnov and M. Vorwerk, *The Ambient Networks Control Space Architecture*, 15th World Wireless Research Forum, Paris, December 2005.

[5] R. Agüero Calvo, A. Surtees, J. Eisl, and M. Georgiades, *Mobility Management in Ambient Networks*, IEEE VTC2007-Spring, Dublin, Ireland, April 2007.

[6] A. Surtees, R. Agüero, J. Tenhunen, M. Rossi, and D. Hollos, *Routing Group Formation in Ambient Networks*, 14th IST Mobile & Wireless Communications Summit, Dresden, Germany, June 2005.

[7] M. Eyrich, M. Majanen, E. Perera, R. Toenjes, R. Boreli, and T. Leinmueller, *GSA: An Architecture for Optimising Gateway Selection in Dynamic Routing Groups*, 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[8] M. Majanen and K. Pentikousis, *An Evaluation of the Ambient Networks Gateway Selection Architecture*, Third International Conference on Wireless and Mobile Communications (ICWMC 2007), Guadeloupe, French Caribbean, March 2007.

[9] E. Guttman, C. Perkins, J. Veizades, and M. Day, *Service Location Protocol, Version 2*, RFC 2608, June 1999.

[10] C. Perkins (ed.), *IP Mobility Support for IPv4*, RFC 3344, August 2002.

[11] C. Perkins, *IP Encapsulation within IP*, RFC 2003, October 1996.

[12] D. Johnson, C. Perkins, and J. Arkko, *Mobility Support in IPv6*, RFC 3775, June 2004.

[13] Y. Sun, E. Belding-Royer, and C. Perkins, *Internet Connectivity for Ad hoc Mobile Networks*, International Journal of Wireless Information Networks, special issue on Mobile Ad Hoc Networks (MANETs): Standards, Research, Applications, 9(2), April 2002.

[14] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, RFC 3561, July 2003.

[15] P. Ratanchandani and R. Kravets, *A hybrid approach to internet connectivity for mobile ad hoc networks*, Proceedings of WCNC 2003, Volume 3, pages 1522-1527, March 2003.

[16] U. Jönsson, F. Alriksson, T. Larsson, P. Johansson, and G.Q. Maguire Jr., *MIPMANET — Mobile IP for Mobile Ad Hoc Networks*, in 2000 First Annual Workshop on Mobile and Ad Hoc Networking and Computing, MobiHoc, 2000.

[17] J. Lee, D. Kim, J.J. Garcia-Luna-Aceves, Y. Choi, J. Choi, and S. Nam, *Hybrid gateway advertisement scheme for connecting mobile ad hoc networks to the internet*, Proceedings of VTC 2003, Volume 1, Pages 191-195, April 2003.

[18] D. Johnson, Y. Hu, and D. Maltz, *The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4*, RFC 4728, February 2007.

[19] M. Ghassemian, P. Hoffmann, C. Prehofer, V. Friderikos, and H. Aghvami, *Performance Analysis of Internet Gateway Discovery Protocols in Ad Hoc Networks*, WCNC 2004 — IEEE Wireless Communications and Networking Conference, Vol. 5, no. 1, March 2004.

[20] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, *Network Mobility (NEMO) Basic Support Protocol*, RFC 3963, January 2005.

[21] K. Leung, G. Dommety, V. Narayanan, and A. Petrescu, *Network Mobility (NEMO) Extensions for Mobile IPv4*, RFC 5177, April 2008.

[22] M. Bechler, W. Franz, and L. Wolf, *Mobile Internet Access in FleetNet*, 13. Fachtagung Kommunikation in verteilten Systemen, Leipzig, Germany, April 2003.

[23] R. Moskowitz and P. Nikander, *Host Identity Protocol (HIP) Architecture*, RFC 4423, May 2006.

[24] P. Nikander, T. Henderson, C. Vogt, and J. Arkko, *End-Host Mobility and Multihoming with the Host Identity Protocol*, RFC 5206, April 2008.

[25] J. Melen, J. Ylitalo, and P. Salmela, *Host Identity Protocol based Mobile Router (HIPMR)*, Internet-Draft, draft-melen-hip-mr, work in progress, July 2008.

[26] S. Nováczki, L. Bokor, and S. Imre, *A HIP based Network Mobility Protocol*, Proceedings of the 2007 International Symposium on Applications and the Internet Workshops (SAINTW'07), 2007.

[27] J. Mäkelä, R. Agüero, J. Tenhunen, V. Kyllönen, J. Choque, L. Munoz, *Paving the Way for Future Mobility Mechanisms: A Testbed for Mobility Triggering & Moving Network Support*, 2nd International IEEE/CreateNet Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (Tridentcom 2006), Barcelona, 2006.

[28] J. Mäkelä and K. Pentikousis, *Trigger Management Mechanisms*, Proc. of IEEE International Symposium on Wireless Pervasive Computing, San Juan, Puerto Rico, February 2007.

[29] R. Ocampo, L. Cheng, Z. Lai, and A. Galis, *ContextWare Support for Network and Service Composition and Self-Adaptation*, MATA 2005, Montreal, Canada, October 2005.

[30] R. Giaffreda, K. Pentikousis, E. Hepworth, R. Agüero, and A. Galis, *An information service infrastructure for Ambient Networks*, Proc. 25th International Conference on Parallel and Distributed Computing and Networks (PDCN), Innsbruck, Austria, February 2007, pp. 21–27. ACTA Press.

[31] F. Berggren, A. Bria, L. Badia, I. Karla, R. Litjens, P. Magnusson, F. Meago, H. Tang, and R. Veronesi, *Multi-Radio Resource Management for Ambient Networks*, 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[32] J. Tenhunen, V. Typpö, and M. Jurvansuu, *Stability-Based Multi-hop Clustering Protocol*, 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Berlin, September 2005.

[33] N. Charkani, E. Hepworth, M. Johnsson, M. Cano, and J. Eisl, *Unbiased Approach to Ambient Control Space Migration*, Proc. First Ambient Networks Workshop on Mobility, Multiaccess, and Network Management (M2NM), Sydney, Australia, October 2007.

[34] S. Deering (ed.), *ICMP Router Discovery Messages*, RFC 1256, September 1991.

[35] R. Moskowitz, P. Nikander, P. Jokela (ed.), and T. Henderson, *Host Identity Protocol*, RFC 5201, April 2008.

[36] J. Laganier and L. Eggert, *Host Identity Protocol (HIP) Rendezvous Extension*, RFC 5204, April 2008.

[37] J. Laganier, T. Koponen, and L. Eggert, *Host Identity Protocol (HIP) Registration Extension*, RFC 5203, April 2008.

[38] P. Jokela, J. Melen, and J. Ylitalo, *HIP Service Discovery*, Internet-Draft, draft-jokela-hip-service-discovery, work in progress, June 2006.

[39] S. McCanne and S. Floyd. ns Network Simulator. http://www.isi.edu/nsnam/ns/.

[40] E. Hernandez and A. Helal, *Examining Mobile-IP Performance in Rapidly Mobile Environments: The Case of a Commuter Train*, LCN 2001, Tampa, FL, November 2001.