Adaptive Server Availability Protocol for Multihop Wireless Networks

M. Alchaita, K. Nasri, M. Nakcha
Higher Institute for Applied Sciences and Technology (HIAST)
Damascus, Syria
mchaita@hotmail.com

ABSTRACT: *With the growing popularity of outdoor locating devices it is both technically and economically feasible for a mobile device to know its physical location. In location-based protocols for multihop wireless networks, routing packets and selecting the correct interface and next hop for a packet being forwarded relies on geographical information of nodes. This requires availability of one static server or number of mobile servers distributed over the network area to provide the destination location. These types of servers suffer from different type of failures due to mobility and single point of failure, which leads to the whole network to go down and heavy load at a single server. In addition, the periods of time spent on looking for a server, and the length of path from a node in the network to the server are other problems. The main objective of this paper is to present a mechanism that overcomes these failures and problems. Results have shown that the proposed mechanism is scalable with the number of nodes, speed and mobility, as well as reducing the time spent on looking for the server, and reducing the path length from a requester node to the server node.*

Keywords: Ad hoc Networks, Location services, Servers availability, Routing protocols, Wireless networks

Received: 18 August 2009, Revised 12 September 2009, Accepted 19 September 2009

1. Introduction

In Mobile Ad hoc Networks (MANet), the communication between nodes goes through intermediate nodes. This, therefore, requires the packet to be forwarded from the source to the destination nodes bypassing the intermediate nodes. The packets could be efficiently forwarded by intermediate nodes with aid of the location information.

The main objectives of most early proposed geographical routing protocols are:

- Reducing the time spent on looking for a server that has information about the position of a required node. The location of this server is known by all nodes.
- Reducing the path length from a requester node to a server node.
- Find solution for the two types of failure caused by node mobility which occurs in most already existing location service approaches. These failures are:

- The server containing the geographical information of nodes may have out-of-date information, which our proposed mechanism solves by using different servers that are updated frequently. Using more than one server in the network minimises the probability of a server being unavailable and hence, minimises the probability of using expired location information.
- The server may move out of its current location which affects the overall functionality of the network.

In geographical routing protocols that rely on server to hold the location of all nodes in the network, the server has to be responsive to the mobility of nodes. When a node moves, its position changes and the server must be updated according to the mobility pattern. In addition, the task of the protocol usually is to provide the requester node with the position of the requested node. Hence, in order to communicate with the requested node, a geographical routing approach should be employed [1-10].

In wireless ad hoc networks, node's location information is useful for efficient routing and location-aware applications [11-15]. A new adaptive server availability protocol for multihop wireless networks is proposed in this paper to overcome the servers suffering from different type of failures due to mobility and single point of failure, which leads to the whole network to go down and heavy load at a single server. In addition, this protocol reduces the periods of time spent on looking for a server, and the length of path from a node in the network to the server.

2. Related Works

In fact, there are three approaches adopted for a possible design of a location service. Firstly, using flooding across the whole network to get nodes locations; mobile nodes broadcast their location information throughout the MANET periodically. So the source knows the up-to-date location of the destination before data transmission. Secondly, central static location server, in which a server is located in one position in the network and all other nodes request the position of requested nodes from this static server. Thirdly, every node acts as server for a few others. Each node is responsible for keeping and tracking the position change of a small number of nodes in the network.

Location services can also be classified according to how many nodes host the service or how many nodes act as a server in the network. In this classification, four classes are available:

I) Some-for-some: in which some nodes act as servers for some other nodes.
II) Some-for-all: in which some nodes act as servers for all nodes in the network.
III) All-for-some: such as each node maintains a routing table for all nodes within two hops.
IV) All-for-all: in which each node maintains the position of all other nodes in the network.

There is a number of existing location service approaches that need to be examined before continuing research in this field.

The Distance Routing Effect Algorithm for Mobility (DREAM) [16] is a flooding-based location service where each node periodically broadcasts a control packet containing its own coordinates with respect to the specific positioning system considered. The routing table stored at each node contains location information for any other node in the network. The disadvantages of this approach are: Excessive flooding messages are used to broadcast the position information across the networks; As a result of the above disadvantage, DREAM consumes significant bandwidth due to the heavy packet flooding across the network.

Quorum-based location service [17] is a some-for-some scheme and originates from information replication in databases and distributed systems where some nodes are chosen to form a backbone network. These backbone nodes are further divided into several quorums such that the intersection of every pair of quorums is non-empty. When a node changes its position, it sends its updated location to one subset containing the nearest backbone node. Each source node then queries the subset containing its nearest backbone for the location of the destination, and uses that location to route the message. The mechanism used in quorum-based protocols is that all nodes in the network agree upon a mapping that associates each node's unique identifier to one or more other nodes in the network. These nodes will be servers for that node where periodical location updates will be sent to and stored into as well as where location queries will be routed to. In other words, the quorum of location servers updated on each location change needs to intersect the quorum of location servers consulted in with a lookup.

The main drawbacks of this mechanism are:

• Heavy computation is required in selecting the backbone and heavy searching is needed for a node that belongs to the backbone to enquire about the location of a node.

• The movement of a backbone node will result in the reallocation of the entire backbone and the topology of the backbone will need to be rearranged.

• The possibility of disconnection of the backbone node from the network.

The Grid Location Service (GLS) [18] is an all-for-some, hierarchical location service that is built upon a number of location servers distributed throughout the network. GLS divides the ad hoc network area into many small squares. The smallest square is called a first order square. Every four adjacent first order squares make up a bigger second order square, and so on (Figure 1).

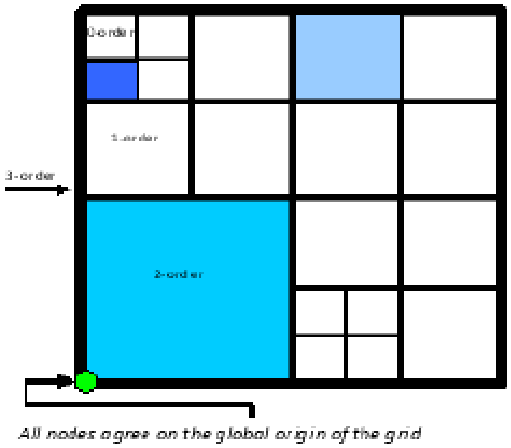


Figure 1. GLS's spatial hierarchy

Each mobile node keeps providing the other nodes available in the same first order square with its up-to-date location. Every mobile node recruits one node as its server from each of four (n-1)-order squares (n > 1) to keep its location information. Selecting the server is done by selecting the node with the least greater node ID in each (n-1)-order square.

The main drawbacks of GLS are:

- The heavy computation of selecting the location of servers for each node, the location query request, and location server update in different grid levels and different distance between the node and its servers, which are some of them in faraway grid.

¨ A node that is acting as a server for other nodes may move out of its current grid. This causes a server failure.

The Home Agent-based System [19] is all-for-some location service scheme. It uses a home agent based strategy for location updates and destination searches. Each node chooses a certain circular area which is the location scope where it first joins the MANET as its home agent. It subsequently sends its location update messages to all the mobile nodes in its home agent. Each node in the network broadcasts its position to neighbouring nodes. Since the location of home agent is transmitted to all the other nodes at the beginning, queries about the position of this node can be sent to its home agent and get the corresponding reply. The drawbacks of this scheme are:

- The inefficiency. When the node moves far away from its home agent, location updates have to go across a long distance. In addition, in some scenarios such as rescue missions or military actions that all nodes possibly move out of the region where all home agents are located. As a result, all homes become ineffective and new homes need to be created. If the movement of nodes is intensive, the home agent-based location service becomes ineffective.

- The home agent-based location service requires a large amount of memory as every node has to keep every other's home agent.

3. Adaptive Server Availability Protocol

Initially, the area covered by the ad hoc network is arranged into a number of concentric circles of increasing diameter size (Figure 2). All nodes agree on the global origin of the concentric circles. The size of smallest diameter of smallest circle is r = R/2, where R is the transmission range of a node. The diameter size of circles is increasing by r. One way to know the global origin by a node joined the network recently could be:

- Either by giving the global origin coordinate to every node manually before joining the network, and this will take some time to setup the devices before using them.
- By giving the global origin coordinate to one node (master node) manually before joining the network, and when a node joins the network, it floods the network using a small beacon packet to enquire about the global origin. When the becaon reaches to the master node, the master node replies with another small beacon packet to inform the requested node about the global origin.

These two above approaches require control over the devices by a network manager. For example, operations that usually involve the saving of life, or prevention of injury by a rescue group in known area such as Search And Rescue (SAR). Another example can be in urban situations when young children or senile people wander away from their homes. The search will be around the homes, and homes will be the origin coordinate.

3.1. Self Determination of Servers

In the very beginning, the smallest circle in the concentric circles network is considered to be a servers' pool, where each node inside this circle is considered to be a server for all other nodes in other circles. In the case of no server in the smallest circle, the next circle to the smallest one is included to the server's pool and the nodes in these two circles will be servers. For example, if the circle of the servers' pool is the smallest circle in the area, this means that servers' pool diameter is r, hence, all nodes with a geographical distance, dn, from the centre of the network area less than or equal to r consider themselves as servers. Formally:

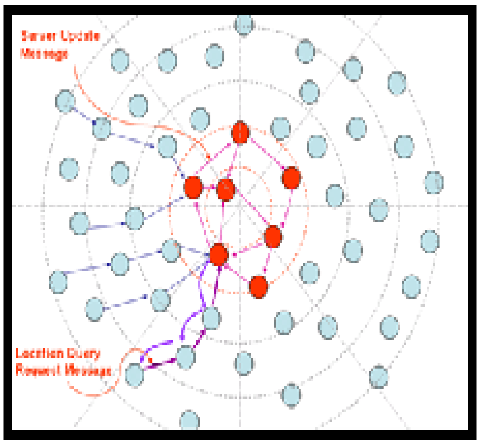


Figure 2. Spatial network

$$f(d_n) = \begin{cases} (d_n \leq r) & \textit{status of node is a server} \\ (d_n > r) & \textit{status of node is a normal node (not server)} \end{cases}$$

The probability of a node to be a server can be calculated using Binomial distribution.

Each node checks its status periodically, if the node is inside the smallest circle, it will change its status from normal node to server node and start receiving the location update messages from other normal nodes (if any message is arrived). In addition, it will exchange the location information of nodes with other servers.

Any node coming inside the servers' pool will change its status to a server, and any node leaving the servers' pool will change its status to a normal node and in the next interval time, it will delete the location information stored in the "Location Table" that was received when the node was a server.

In general, in the binomial distribution with the number of mobile nodes, n, and probability, p, of a node to be a server, the probability of getting exactly k servers to be available in the servers' pool is given by the probability mass function:

$$f(k;n,p) = \binom{n}{k} p^k (1-p)^{n-k}$$

For $k = 0, 1, 2, \ldots, n$

Where

$$\binom{n}{k} = \frac{n!}{n!(n-k)!}$$

is the binomial coefficient "*n choose k*". The formula can be understood as follows: we want k successes (p^k) (servers) and $n - k$ failures $((1-p)^{n-k})$ (not servers). Figures 3 and 4 show the Binomial distribution with different number of nodes and different values of p.

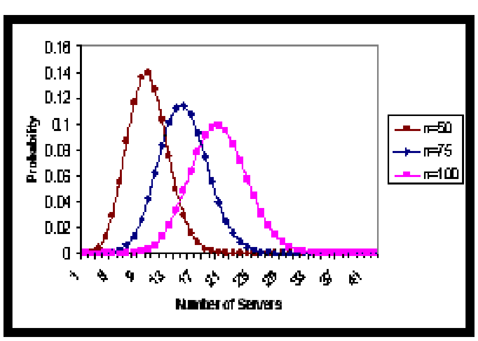


Figure 3. Binomial distribution for p = 0.2

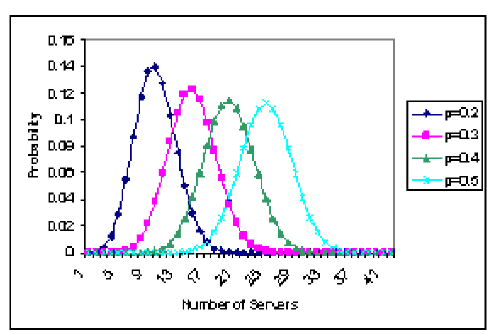


Figure 4. Binomial distribution for n = 50

3.2. Gathering Locations by Servers

Each mobile node in the ad hoc network sends its location information to its neighbour's nodes in an interval time. This is not only useful for knowing the neighbours locations, but it is also useful in determining the size of the servers' pool. For example, when a node in the second circle from the centre has no neighbours in the first circle, the node considers itself as a server and starts doing the task duty of the server. Each node in the network provides the servers with its current position when this node moves. Each node in the network sends an "Update Message" toward the servers' pool. Forwarding the "Update Message" occurs by selecting the next hop that is closer to the servers' pool. It first consults its routing table and chooses to forward the packet to the neighbour closest to the servers' pool (Figure 5).There is, however, a potential problem if a node does not know about any nodes closer than itself to the servers' pool. This dead-end situation can be overcome by using the Greedy Perimeter Stateless Routing (GPSR) algorithm [9].

The "Update Message" does not require a reply message or a long life of the link between the sender and next hop since the sender needs only to hand the "Update Message" off to the next hop. Therefore, for optimisation, the closest neighbour to the servers' pool is selected (farthest neighbour from the sender node), which will decrease the number of hops needed for forwarding the "Update Message".

When a server node receives the "Update Message", the server node updates the location information of the originator of the "Update Message" in its "Sever Location Table" and stops forwarding this message. The "Update Message" contains a sender node's ID and its geographic location. Whilst the "Sever Location Table" contains list of records containing a node's ID and its geographic location.

When a node moves, it must send an update message toward the servers' pool diametric. To avoid excessive update traffic, the Update Frequency is calculated using a "Threshold Distance" and the location of the node depending on the servers' pool. If the node, due to movement, comes inside the servers' pool, it changes its status to server and forwards its location information to other servers in the pool.

The "Threshold Distance" is the distance the node has travelled since the last update. For example, when a node moves a distance d; the node then updates its location servers. In other words, a node updates its location servers at a rate proportional to its speed, and the slower nodes are updated less frequently than the faster nodes.

Each server exchanges its "Location Table" with other servers by sending "Server Update Message" (SUM) in a periodic manner. The SUM includes the locations table of the nodes in the network, and the servers' information table. Each entry in the locations table contains a node's ID and its geographic location. Each entry in the servers' information contains a server's ID, its position, and its radius. As an optimisation, in order to reduce the size of the SUM, the server that needs to send a SUM only includes the updated/changed information in the server location table and the servers' information table. By exchanging the SUM (Figure 1) the servers know each other.

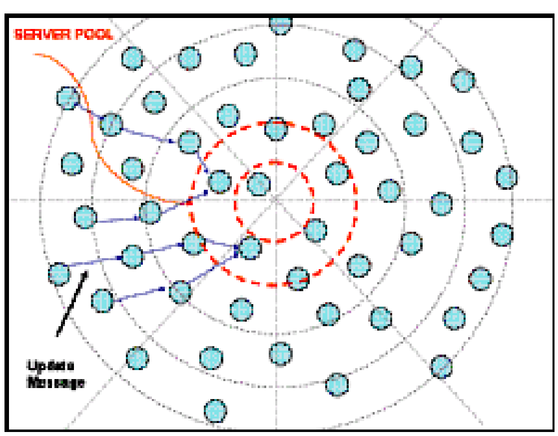


Figure 5. Spatial network

3.3. Requesting Location of Destination

When a node needs a location of a destination node, it triggers a "Location Query Request Message" (LQRM). The LQRM is forwarded using geographic forwarding toward the servers' pool by selecting the nodes that are closest to the centre of pool. A node may not know about any nodes closer than itself to the servers' pool. This dead-end situation is overcome by using GPSR. The LQRM includes information about the originator node and the IP address of requested node. The originator information is used for forwarding the request reply toward the originator.

When one server receives the LQRM, it first updates the sender information in the location table and changes the flag of the updated record to “Updated record”. This is because, at the interval time, this server will send a Server Update Message to other servers including this updated record. After that, the server checks if it has the requested location information for the requested destination, and if the location of the requested node is available in the location table, the server replies by sending a location reply message toward the sender following the same reversed path followed by the LQRM. Whilst If the location of the requested node does not exist in the location table, the server consults other servers for the requested location by flooding the request packet to its neighbours. The server that received this message and has an answer about the requested location replies directly to the originator node of the LQRM. The node that is not a server will forward the request message toward the closest neighbour to the centre.

The originator of the LQRM waits for a reply for a period of time (TLR). If no reply is received, another request message is sent. The number of retries is limited by a number (e.g. 3 times). After that, the originator has two options: either the LQRM is flooded across the network or the destination is considered unavailable, because each node in the network is required to send its location information to the servers' pool. In the validation of our protocol, the second option was chosen, since each node should have registered in the servers and we are trying to minimise the overhead. After receiving the reply message, the node starts sending the data packets toward the destination location.

4. Optimisation

Some optimisations implemented in the proposed protocol are summarised in the following:

- For optimisation during forwarding the update message, the closest neighbour to the servers' pool is selected (farthest neighbour from the sender node), and this decreases the number of hops needed for forwarding the update message.
- To avoid excessive update traffic, the update frequency is calculated using a threshold distance and the location of the node. The threshold distance is the distance the node has travelled since the last update.
- In order to reduce the size of the SUM, the server that needs to send a SUM includes in the message only the updated/changed information in the server location table and the servers' information table.
- One of the optimisations implemented in our proposed protocol uses the expiration time of the geographical information of the destination node. This is will reduce the number of request messages sent when receiving data packets in very short time (shorter than the expiration period) for the same destination.
- During forwarding the query request packet, if this packet meets the requested node, the requested node sends a query reply without continuing to forward the request packet toward the servers.
- During forwarding the data packet, if the packet meets a server that has fresher information about the destination location; the data packet updates the information regarding the destination in its fields. Hence forwarding the packet will be more accurate. This

5. Results and Analysis

To measure the effectivity and efficiency of our proposed protocol in different environments under a range of conditions, and to measure its reaction to network topology changes, the NS-2 simulator is used.

Nodes in the simulation move according to the Random Waypoint model [20-23]. The threshold distance (D) values are 25, 50, and 75 m. Without loss of generality, the parameters considered in the simulation are shown in table I.

The following performance metrics are used for evaluation:
1) The average and minimum number of servers

2) The success rate of delivering the location update packets
3) The number of generated location update packets
4) The success rate of delivering the location query request packets
5) The number of generated location query request packets
6) The data packet delivery ratio
7) The average end-to-end-delay of data packet

Each parameter metric mentioned above is evaluated in different scenarios:

a) Network sizes scenario: to evaluate the performance of our proposed protocol with different number of nodes, and to investigate its scalability with number of nodes.

b) Speed scenario: to evaluate and investigate the performance of proposed protocol with different values of speed.

c) Mobility scenario: this scenario shows the performance of our proposed protocol in terms of the mobility with different pause time values.

5.1. The Average and Minimum Number of Servers

This parameter shows the average and minimum number of servers available in the network during the simulation time. Figure 6 (a) shows that the average number of servers increases with increasing the number of nodes in the network.

Parameter Name	*Value*	*Description*
Pause Time (Sec)	*10*	*The pause time of movement*
No. of Nodes (Node)	*50*	*Number of nodes in the network*
Network Area (WxH)	*1000x1000*	*The coverage area of the network*
Node Density(node/m2)	*50/1000x1000*	*It is the number of nodes divided by the total simulation area.*
Coverage Area (m2)	*196,428*	*Coverage area is the area of the circle whose radius is the transmission distance*
Avg. No. of Neighbours	*9.817*	*Dividing the coverage area by the node density.*
Servers' pool Radius (m)	*125*	*minimum radius of server pool*
D_Threshold (m)	*25*	*The threshold distance*
Center_x, center_y	*500, 500*	*Centre of the network area*
No. of Sources (Node)	*10*	*Number of source nodes*
Speed (m/s)	*10*	*The average speed of nodes*
Data Type	*CBR*	*Type of data*
Data Packet size (Byte)	*64*	*The data packet size*
Channel Capacity (Mbps)	*2*	*The channel capacity*
Packet Rate (P/Sec)	*4*	*The rate of sending data packets per second*
Transmission Range (m)	*250*	*The transmission range of the node*
No. of Location Request Retries	*3*	*The number of resending the location request query*
Simulation Time (Sec)	*500*	*The simulation time*
Hello_Interval (Sec)	*1*	*Hello interval time*
Allowed_Hello_Loss	*3*	*Number of Hello packets allowed to be lost*
Am_I_Server_Interval (Sec)	*1*	*Checking whether the node is serve or not*
Update_Loc_Interval (Sec)	*1.5*	*Update the server interval time*
Server_Update_Interval (Sec)	*1*	*Exchange information between server interval time*
Propagation Type	*TwoRayGround*	*The propagation type*

Table 1. Simulation's parameters

Figure 6 (b) depicts that always there is at least a server in the network and the minimum number of servers is always greater than zero. This is due to the adaptive server pool size that increases when the number of servers reaches to zero to include other servers from the next bigger circle following to the circle of servers.

The average and minimum number of servers against speed and pause time are shown in Figures 7 and 8 where the minimum number of servers remains greater than zero and the average also remains greater than 3.

5.2. The Success Rate of Delivering the Location Update Packets

The Success Rate of Delivering the Location Update Packets (SRDLUP) is the ratio of number of location update packets generated by normal nodes (not servers) over the number of location update packets received by servers' nodes. As can be seen in Figure 9 (a) the algorithm is scalable with the number of nodes joining the network. It can be seen that with increasing the number of nodes, the SRDLUP decreases very slightly but still remains above 84%. The slight reduction in SRDLUP with the increased number of nodes is due to the congestion occurred by the nodes in the network. The average of SRDLUP is 0.92. The SRDLUP rate at number of nodes equal to 50 is 0.99, which indicates that almost all location update packets reach and are received by servers.

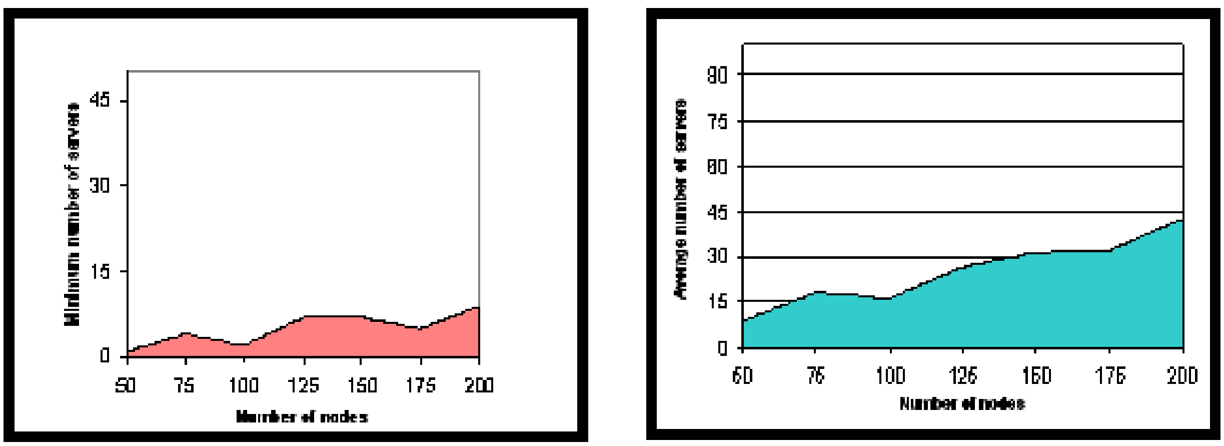


Figure 6. Average and minimum number of servers vs. number of nodes

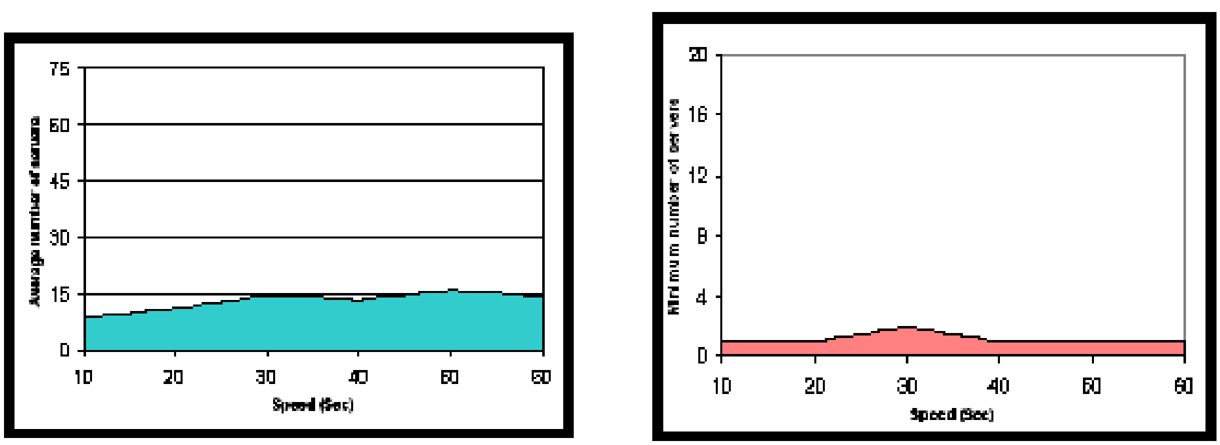


Figure 7. Average and minimum number of servers vs. speed

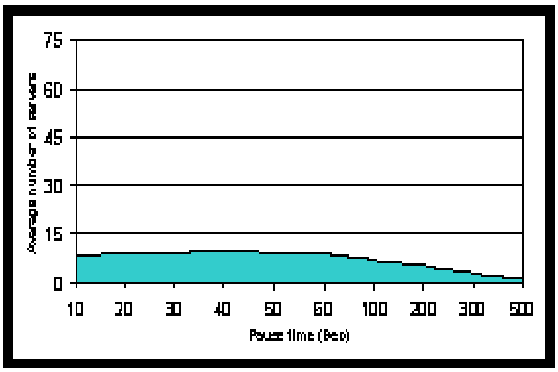
In terms of the scalability with the speed of nodes, Figure 9 (b) shows that the SRDLUP rate is approximately 100% with different values of speed. The average value of SRDLUP is 0.985 and the maximum value is 0.99 at speed 10m/s. This indicates that our proposed protocol is also scalable with the speed of nodes.

The performance of the algorithm in terms of SRDLUP against the mobility (pause time) is shown in Figure 9(c). As can be seen in this figure, the SRDLUP is almost constant with different values of pause time and is nearly about 100% although it does decrease slightly when all nodes in the network are stationary (pause time =500 s).

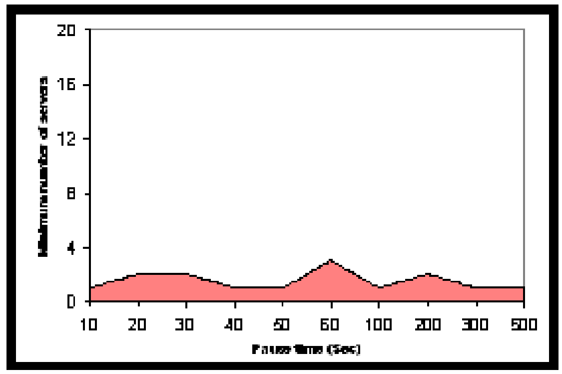
The average value of SRDLUP is equal to 0.973, and the maximum value is 0.99 which is at pause time 60 sec. this is because the location update packet just needs to be forwarded to nodes that are always closest to the centre, irrespective whether it is mobile or stationary.

5.3. The Number of Generated Location Update Packets

This parameter shows the number of location update packets generated in the network by the nodes in order to update the servers with nodes' geographical locations.

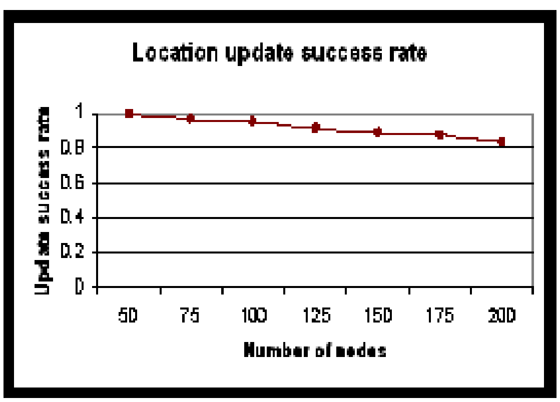


8 (a)

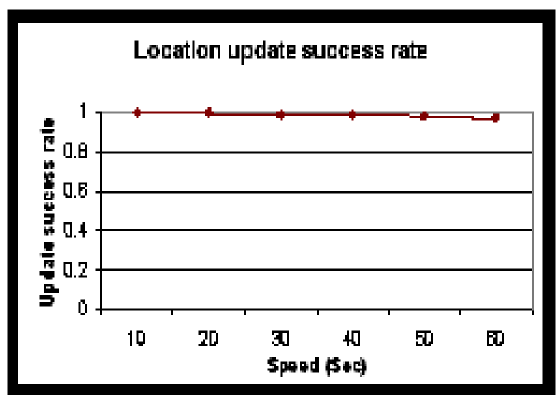


8 (b)

Figure 8. Average and minimum number of servers vs. pause time



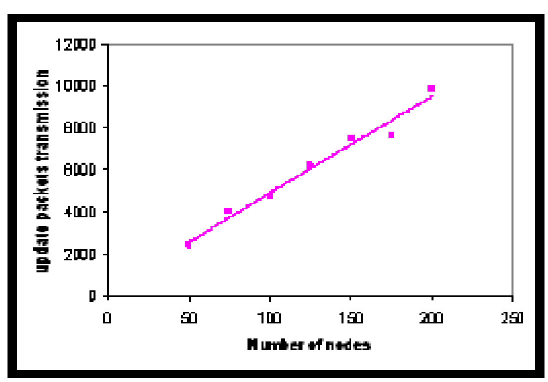
9 (a)



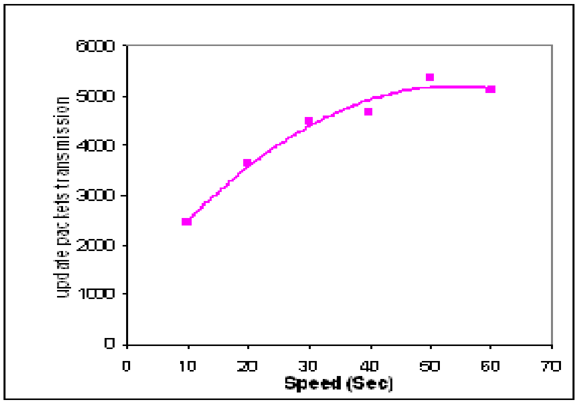
9 (b)

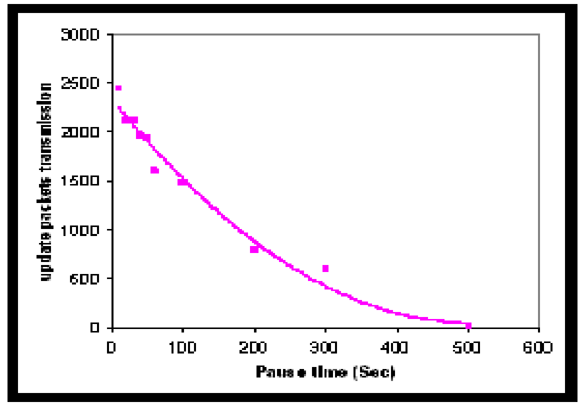
9 (c)

Figure 9. Location update success rate



10 (a)

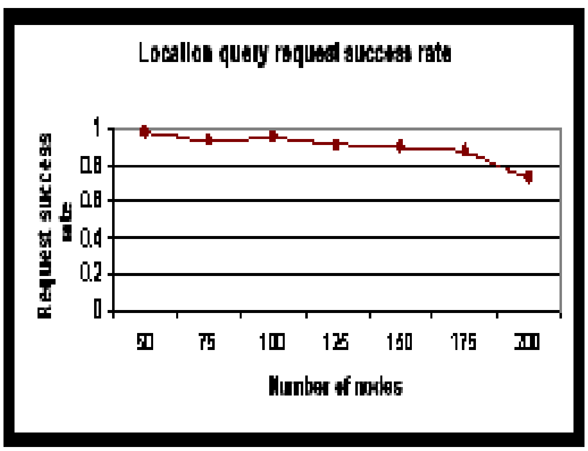


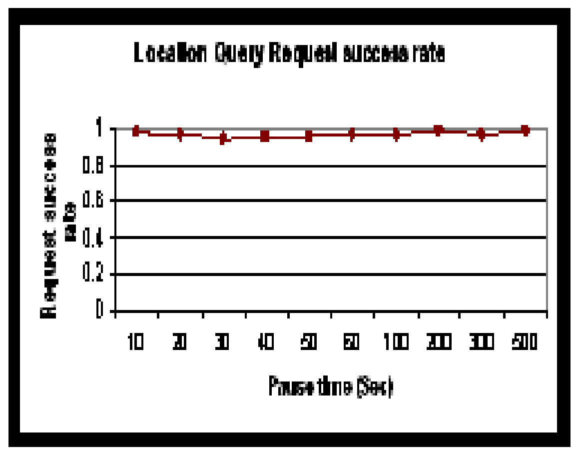


10 (b)

10 (c)

Figure 10. The Number of generated location update packets





11 (a)

11 (b)

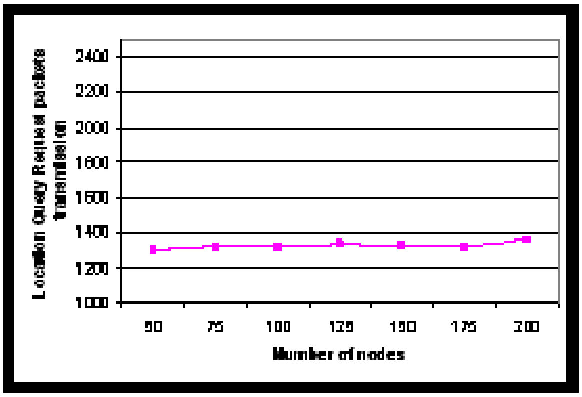
Figure 11. The success rate of delivering the location query request packets

Figure 10 shows the the number of generated location update packets against the number of nodes, speed, and mobility (pause time). As can be seen the generated packets are increasing gradualy with increasing the number of nodes and speed while decreasing gradually with increasing the pause time (toward more stationary) until it reaches to nearly zero at pause time 500 where all nodes are stationary and no longer need to generate update packets.

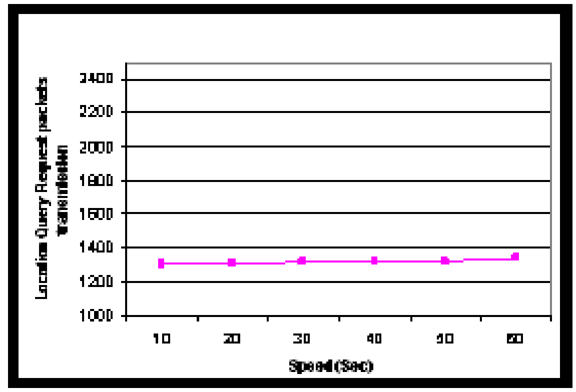
5.4. The Success Rate of Delivering the Location Query Request Packets

The Success Rate of Delivering the Location Query Request Packets (SRDLQRP) is the ratio of number of location query request packets generated by the nodes in the network over the number of location query request packets received by the originator nodes of the requests. As can be seen in Figure 11 (a) that the algorithm is scalable with the number of nodes joining the network, since with the increasing number of nodes, the SRDLQRP is decreased very slightly and remains above the 87%. The slight reduction in SRDLQRP with the increased number of nodes is due to the congestion incurred by nodes in the network. The average of SRDLQRP is 0.89. The SRDLQRP rate at number of nodes equal to 50 is 0.97, which indicates that almost all location query request packets generated by the generator nodes have got replies about the location of requested nodes.

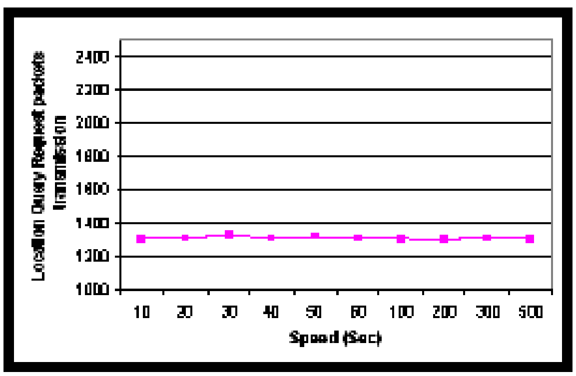
In terms of the scalability with the speed of nodes, Figure 11 (c) shows that the average value of SRDLQRP is 0.95, and the maximum value is 0.97 at speed 10m/s. this indicates that our proposed protocol is also scalable with the speed of nodes.



12 (a)

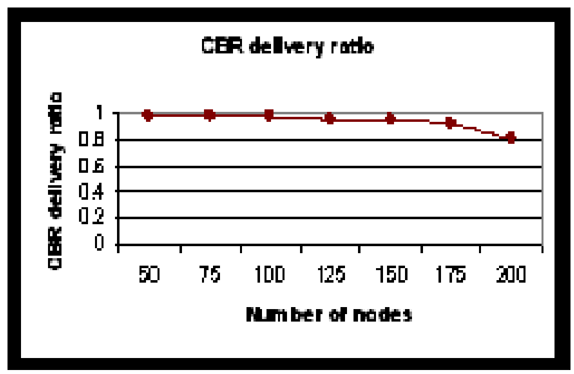


12 (b)

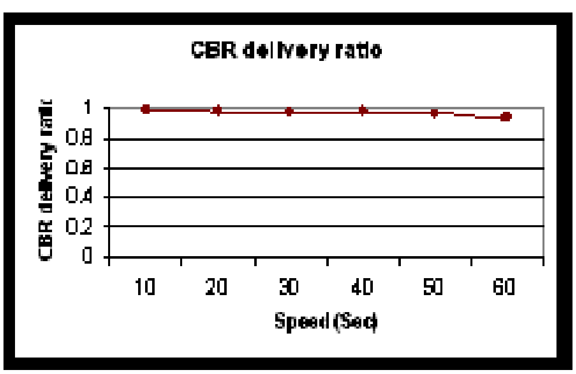


12 (c)

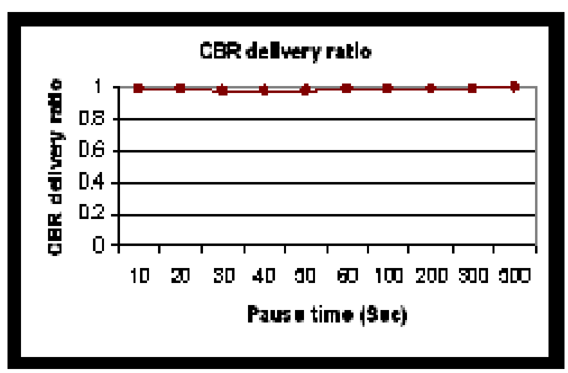
Figure 12. The location Query Request packets transmission



13 (a)



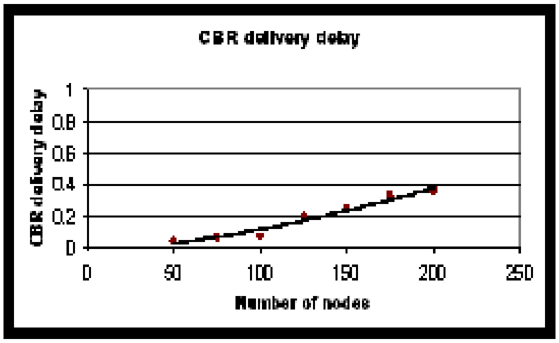
13 (b)



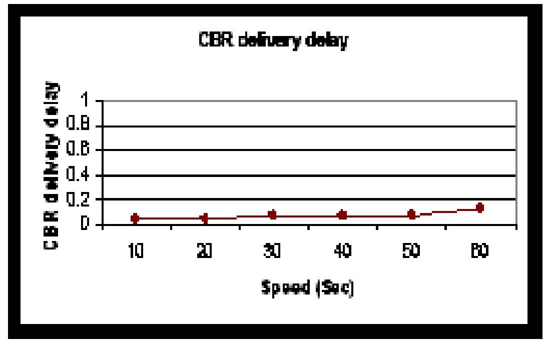
13 (c)

Figure 13. CBR delivery ratio

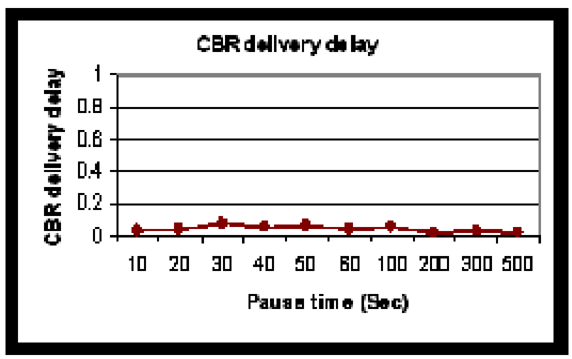
The performance of algorithm in terms of SRDLQRP against the mobility (pause time) is shown in Figure 11 (b). As can be seen in this figure, the SRDLQRP is almost constant with different values of pause time. The average value of SRDLQRP is equal to 0.97, and the maximum value is 0.99. This is because the location request packet just needs to be forwarded to nodes that are always closest to the centre,and request reply packets just needs to be forwarded to nodes that are always closest to theoriginator of the request packet, no matter if it is mobile or stationary.



14 (a)



14 (b)



14 (c)

Figure 14. End-to-end-delay of data packets

5.5. The Number of Generated Location Query Request Packets

This parameter gives an idea about the number of location query requests generated in the network by all source nodes that required geographical location of their destinations. When a source node has data packets to send to a destination node, it first sends a location query request packet toward servers in order to get the location of that destination, when the source receives the request reply packet that includes that location, the source starts sending the data packets to the destination. It can be seen in Figure 12(a,b,c) that the number of location query request packets is nearly constant and scalable with the number of nodes , speed and mobility (pause time).

5.6. The Data Packet Delivery Ratio

The Data Packet Delivery Ratio (DPDR) is the number of data packets that are received by the intended destinations over the number of data packets sent by the source nodes.

As can be seen in Figures 13 (a) (b) that DPDR is slightly decreasing with increased the number of nodes and speed due to the congestion and mobility caused by nodes. The DPDR remains about 0.95 up to number of nodes equal to 150 nodes and decreases after that to up to 0.815 at number of nodes equal to 200 nodes. The average of DPDR against the number of nodes is 94%. Whilst the average against the speed is 0.973, and the average against the mobility (pause time) is 0.986. In Figure 13(c), the DPDR is constant with the mobility.

5.7. The Average End-to-End-Delay of Data Packets

The Average End-to-End-Delay (AEED) of data packet is the average time required to deliver the data packets from the source nodes to destination nodes. This time includes all possible delays caused by buffering during route discovery, queuing at the interface-queue, retransmission delays at the medium access control layer, propagation and transfer times and ARP delay that has a considerable value. It can be seen in Figure 14 (a) that AEED increases with increased number of nodes due to the congestion and contention occurred by nodes. The speed of nodes has a small effect on the AEED because the packets are not relying on an already established and complete route from the source to the destination. Instead, each node forwards the packet to next downstream node that is as close as possible to the destination node. During forwarding, the packet, if it is received by a server node or a node that has fresher information about the location of the destination, will update the location of the destination included in the packet itself.

6. Summary

In this paper, a new adaptive server availability protocol was presented. This protocol tracks mobile node locations and overcomes the main problems of centralised static servers in location services approaches, which were mentioned in the beginning of this paper. These problems indicate the failure of the server, which leads to the whole system to go down and heavy load at a single server. In addition, this protocol guarantees the scalability and the adaptiveness with the number of nodes, speed, and mobility.

Results have shown that the proposed protocol is scalable with the number of nodes, speed and mobility. As shown by the results, the average location update success rate against the number of nodes, the speed, and the mobility were 0.92, 0.98, and 0.97 respectively. The average of success rate of delivering the location query request packets against the number of nodes, the speed, and the mobility were 0.89, 0.95, and 0.97 respectively. The data packet delivery ratio remained round 0.95 for number of nodes ranging between 50 and 150 nodes. The above ratio started decreasing after that to reach 0.815 for number of nodes ranging between 50 and 200 nodes. The average of data packet delivery ratio against the number of nodes, the speed, and the mobility were 0.94, 0.97, and 0.986 respectively.

7. References

[1] Boleng, J. Camp, T. (2004). Adaptive Location Aided Mobile Ad Hoc Network Routing, *In*: Proceedings of the 23rd IEEE International Performance, Computing, and Communications Conference (IPCCC '04), p. 423-432.

[2] Witt, Matthias., Turau, Volker (2005). BGR: Blind Geographic Routing for Sensor Networks., *In*: Proceedings of the Third Workshop on Intelligent Solutions in Embedded Systems (WISES'05). Hamburg, Germany.

[3] Câmara, Daniel., Alfredo, Antonio., Loureiro, F. (2001). A Novel Routing Algorithm for Hoc Networks", Baltzer Journal of Telecommunications Systems, 18 (1-3) 85-100. Kluwer Academic Publishers.

[4] Bose, Prosenjit., Morin, Pat., Stojmenovic, Ivan., Urrutia, Jorge (1999). Routing with guaranteed delivery in ad hoc wireless networks, *In*: DIALM '99: Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications, Seattle, Washington, United States, p. 48-55.

[5] Ko, Y.-B., Vaidya, N. H (2000). Location-Aided Routing (LAR) in mobile ad hoc networks, *Wireless Networks* 6.

[6] Ko, Y.-B., Vaidya, N.H. (1998). Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, *In*: Proc. the ACM/IEEE Int. Conf. Mobile Computing and Networking (MOBICOM), Dallas, TX.

[7] Joa-Ng, M., Lu, I.-T. (1999). A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad Hoc Networks, *IEEE Journal on Selected Areas in Communications*, special issue on Wireless Ad Hoc Networks, 17 (8).

[8] Su, W., Gerla, M. (1999). IPv6 Flow Handoff in Ad Hoc Wireless Networks Using Mobility Prediction. *In:* Proc. the IEEE Global Telecomm. Conf. (GLOBECOM), Rio de Janeiro, Brazil, December.

[9] Karp, B., Kung, H.T (2000). Greedy Perimeter Stateless Routing for Wireless Networks, *In:* Proc. the Sixth Annual ACM/IEEE Int. Conf. Mobile Computing and Networking (MobiCom 2000), Boston, MA.

[10] Karp, B. (2000). Geographic Routing for Wireless Networks, PhD thesis, Harvard University, Cambridge, MA, October 2000.

[11] Biaz, S., Ji, Yiming (2005). A survey and comparison on localisation algorithms for wireless ad hoc networks, *Int. J. of Mobile Comm*. 3 (4) 374 – 410.

[12] Toh, C.K. et. al. (2007). Service Location Protocols for mobile wireless ad hoc networks, *Int. J. of Ad Hoc and Ubiquitous Computing*, 2 (4) 250 – 262.

[13] Varshney, U. (2003). Location management for wireless networks: issues and directions, *Int. J. of Mobile Comm*. 1 (1/2) 91-118.

[14] Raya, M., Hubaux, J.-P. (2007). Securing vehicular ad hoc networks, *Journal of Computer Security*, 15 (1) 39–68.

[15] Kim, Y.-J., Govindan, R., Karp, B., Shenker, S. (2005). Geographic Routing Made Practical, *In:* Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI '05), USENIX Association.

[16] Basagni, S et. al, (1998). A distance routing effect algorithm for mobility (DREAM), *In*: Proc. the Fourth Annual ACM/ IEEE Int. Conf. Mobile Computing and Networking (Mobicom'98), Dallas, TX, .

[17] Liang , B., Haas, Z. (2000). Virtual Backbone Generation and Maintenance in Ad Hoc Network Mobility Management, *In*: INFOCOM.

[18] Guba, N., Camp, T. (2002). GLS: a location service for an ad hoc network, *In*: Proc. the Grace Hopper Celebration (GHC '02).

[19] Stojmenovic, I. (1999). Home Agent Based Location Update and Destination Search Schemes in Ad Hoc Wireless Networks, Tech. rep. TR-99-10, Comp. Science, SITE, Univ. Ottawa.

[20] Hyytiä , E., Virtamo, J. (2006). Random waypoint model in n-dimensional space, *Operations Research Letters*, 33/6.

[21] Hyytiä, Esa., Lassila, Pasi.,Virtamo, Jorma (2006). Spatial Node Distribution of the Random Waypoint Mobility Model with Appli-
cations. *IEEE Trans. Mobile Computing*.

[22] Hyytiä, Esa., Virtamo, Jorma (2005). Random waypoint model in n-dimensional space, *Operations Research Letters*, 33/6, p. 567-571.

[23] Hyytiä, Esa., Lassila, Pasi., Virtamo, Jorma. (2006). A Markovian Waypoint Mobility Model with Application to Hotspot Modeling, *In:* Proceedings of IEEE ICC 2006, June.