October 2003

An Approach for Optimizing Binding Lifetime with Mobile IPv6

Zheng da Wu
Bond University, Zheng_Da_Wu@bond.edu.au

An Approach for Optimizing Binding Lifetime with Mobile IPv6

Z. D. Wu
School of Information Technology, Bond University, Australia
wz@bond.edu.au

Abstract

In this paper an analytical model is presented in order to provide useful insight into the performance behaviour of Mobile IPv6 in the context of a binding lifetime setting. By using this model, the optimal binding lifetime can be estimated in terms of a user mobility, traffic workload, and network structure. An algorithm for dynamically setting the binding lifetime in Mobile IPv6 networks is proposed based on the analytical model. The numerical results of the system simulation are also demonstrated.

1. Introduction

Mobile IP is a global mobility solution that provides mobility management for a wide variety of radio technologies, devices and applications. It allows a mobile node to change its location without the need to restart its applications or terminate any ongoing communication. There are two variations of Mobile IP, Mobile IPv4 and Mobile IPv6. Since IPv6 is becoming a standard for the next generation of IP based networks, Mobile IPv6 has been developed by the IETF with some new functionalities for streamlining mobility support that are missing in Mobile IPv4. The work to be described in this paper is based on Mobile IPv6 (MIPv6) [1]. Significant research results in the extensions for MIPv6 has been reported during the last several years. The works deal with a number of aspects: Hierarchical MIPv6 Mobility Management [2], Fast Handovers for MIPv6 [3], QoS Guarantees with MIPv6 [4], Cellular IP [6] and Mobile IP with Paging [7]. However, there is a practical and common issue, to which less attention is paid. The problem is about the specification of *Binding Update Lifetime* [1].

In MIPv6, a *Binding* is the association of the home address of a mobile node (MN) with a care-of address (CoA) for that mobile node, along with the remaining lifetime of that association. A *lifetime* is specified whenever a MN sends a *Binding Update (BU) Message* to its home agent (HA) and relevant corresponding nodes (CNs). It is identified that specifications of a lifetime impact on the system performance. If the lifetime is set too short, the frequency of sending binding updates from the MN to it's HA and CNs will be increased. If it is set too long, the binding record will eventually occupy more space in the both data structures, *Binding Cache* and *Binding Update list*, of the mobile nodes [1]. In addition, the bindings may be obsolete frequently and thus lead packets delivery from a MN to its CN inefficiently. On other words, the function of *Route Optimization* cannot be benefited well, which is a key of operations with MIPv6 for avoiding "triangle routing" problem. Therefore, an approach for optimizing the binding lifetime is needed in the context of MIPv6 networks.

In this paper an analytical model is presented in order to provide useful insight into the performance behaviour of Mobile IPv6 in the context of a binding lifetime setting. By using this model, the optimal binding lifetime can be estimated in terms of a user mobility, traffic workload, and network structure. An algorithm for dynamically setting the binding lifetime in Mobile IPv6 networks is proposed based on the analytical model. The numerical results of the system simulation are also demonstrated. The paper is organised as follows: A brief description of the binding management with MIPv6 is given in Section 2; Performance modelling and optimization will be discussed in Section 3; an algorithm for dynamically setting the lifetime and the numerical results are presented in Section 4 and 5 respectively. Finally, the conclusion and further work will be addressed

2. Binding Management

Branch and bound is a general concept, and concrete implementation strategies of the idea vary greatly in different situations. In this section the binding management with MIPv6 is briefly described. More detailed specification of MIPv6 can be found in [1]-[4]. As mentioned before, a binding for a MN is the triplet that contains the home address, CoA, and the binding lifetime. With MIPv6, every node has a *Binding Cache*, which is used to hold the bindings for other nodes. If a node receives a binding update, it will add this binding to its Binding Cache. In addition, every MN has a *Binding Update List*, which is used to store information about each binding update sent by this mobile node for which the lifetime has not expired yet. A binding entry is removed from either the Binding Cache or Binding Update List whenever its lifetime has expired.

A binding update is triggered in the following situations: (i) When a MN is away from the home and gets a new CoA successfully, the MN sends a binding update to its HA and relevant CNs; (ii) a CN can initiate a refreshing of binding by sending *Binding Refresh Request* towards a relevant MN if the binding is near expired; (iii) During the tunneling for packets delivery from a CN to its MN through the HA, a binding update can be sent to the CN from the MN, so as to take advantage of the Route Optimization. To make sure, that the intended receiver receives a binding update, a MN can enforce the receiver to acknowledge the receipt of a binding update by responding a *Binding Acknowledgement*. Until receipt of the Acknowledgement the MN continues retransmitting the binding update message periodically.

With Mobile IPv6, whenever a node (either stationary or mobile) sends a packet, the Binding Cache of this node is searched. In case of a MN receiving a packet from a CN it is able to detect, if the sending CN has already a Binding Cache entry for that MN, it addresses the packet directly to the mobile node's CoA. Otherwise, the CN sends that packet to the home address of the MN which is tunneled by the HA to the MN. The MN may send a binding update to the CN to enable it, to send future packets directly to the MN without tunneling by the HA, as mentioned in (iii) above. A MN must set the *Acknowledge Bit* in Binding Update Message addressed to a home agent. The MN may also set the acknowledge bit in the binding updates sent to a CN. If the Binding Update was not received by the CN, the MN would recognize this in receiving still tunneled packets from the HA.

3. Models and Analysis

In this section a mathematical model is presented for evaluating the behaviors of a binding lifetime in terms of user mobility, network structure and traffic workload. By using this model, an optimal lifetime can be estimated. Without losing generality, wireless cellular IP networks are considered. Assuming that the mobility management of a wireless IP cellular network is divided into two levels: the macro and micro levels, called *Domain level* and *Location Area (LA) level*. Each domain is assumed to equip with a HA and geographically covers a number of LAs. Each LA aggregates a group of cells (or subnets), which are equipped with Mobile IPv6 routers. When a MN leaves a LA and enters another LA, it initiates a registration or location update procedure as discussed in the last section. An MN traveling through two or more cells within the same LA will not lead to the binding update procedure.

.

3.1 Modeling

Suppose that (i) all the cells are of the same shape and size and form together a contiguous area; (ii) the density of users is uniformly distributed in the area; (iii) the direction of users motion with respect to the border is uniform on $[0,\ 2\pi)$. Let X and Y be independent identically distributed random variables representing the call arrival time and the cell residence time. X and Y are also exponentially distributed with rate λ_c and λ_r respectively. Consider a cell, LA and domain as circular area approximately. The border-crossing rate for a MN out of a cell is given by

$$V_{CL} = \frac{\pi V}{4R_{CL}}$$

where V is the average travelling velocity of a MN and R_{CL} is the radius of circular area of a cell [9]. Similarly, the border-crossing rate for a MN out of a LA is expressed as

$$V_{LA} = \frac{\pi V}{4R_{LA}}.$$

Since a MN crossing a LA border must cross a cell border, the border-crossing rate of the MN still staying in the same LA is computed as

$$V_{CL\in LA} = V_{CL} - V_{LA} = \frac{\pi V(R_{LA} - R_{CL})}{4R_{CL}R_{LA}}.$$

The location update process of a MN is modelled as an Imbedded Markov Chain, which can be illustrate as in Fig. 1, where $\lambda = V_{CL\in LA}$ and $\mu = V_{LA}$.

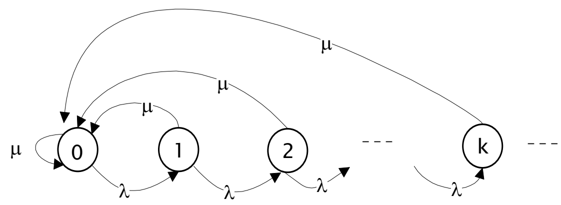

Figure 1: Markov Chain for Location Update Process

A state of the chain is defined as the number of cells in the same LA where the MN has travelled around. Thus, λ represents the state transition rate at which the MN moves from state *k* to state *k+1, k = 0, 1, 2, …*. In other words, the MN is travelling between cells within the same LA at rate λ. μ denotes the state transition rate at which the MN moves from state *k* to state *0*. It means that the MN is moving out of a LA to another LA. In such state transition, the MN will perform the location update procedure. The equilibrium state probability of state *k* is denoted as p_k. Thus, p_k can be expressed as

$$p_k = [\frac{\lambda}{\lambda+\mu}]^k p_0 \qquad (1)$$

where p_0 is the equilibrium state probability of state 0. By using the law of total probability, p_0 can be obtained as

$$p_0 = \frac{\mu}{\mu+\lambda} \text{ and } p_k = (1-p_0)^k p_0 .$$

3.2 Cost Computation

In this subsection, the total cost of mobility management with Mobile IPv6 is estimated. The mobility management cost is divided into two components: (i) location-update cost, which incurred in completing the location update procedure as discussed in section 2; (ii) call delivery cost which incurred in completing the delivery of the first IP packet (or a call) from a CN to the MN, for example of establishing a connection between them. A number of parameters or costs dealing with location-update and packets delivery are denoted as the following:

$D_{MN,HA}$ - The average distance between a MN and its HA in terms of the number of hops;

$D_{MN,CN}$ - The average distance between a MN and its CN (*hops*);

C_l - The average cost for location-update procedure, weighted as *hops*msg/sec*.

C_d - The average cost for call delivery procedure (*hops*msg/sec*);

$C(t)$ - The total cost of mobility management during period *t*.

The calculation for the cost related to the performance of the binding management with MIPv6 is summarized as the following:

Whenever a CN wants to send an IP packet to a MN, it will check the Binding Cache entry associated with that MN.

Case 1: Suppose that the CN has already got a CoA in the MN's entry and the lifetime of the binding has not expired in this case. The CN can send the IP packet directly to the MN by using this CoA. Otherwise, the CN has to send them to the MN's home address. The first packet (or call) delivery can be successful if the MN still stays in the same LA indicated in the binding. The total cost of this situation, denoted as C^1, is given by

$$C^1 = C_l^1 + C_d^1 ,$$

where $C_l^1 = 0$ and $C_d^1 = 2D_{MN,CN}$, since there is no location-update cost in this case and the call delivery only deals with the distance between the MN and its CN. The cost of acknowledgement for a packet delivery is considered in this model.

Case 2: During the packet delivery, the MN may move out to another LA, where it has updated the CoA which is different from the address used in Case 1. In this situation, the total cost is given by

$$C^2 = C_l^2 + C_d^2 ,$$

where $C_l^2 = 2D_{MN,HA} + 2D_{MN,CN} + D_C^2$, since a location update must be carried out. D_C^2 represents the total location update cost for the MN with its all other CNs listed in the Binding Cache. D_C^2 is simplified as a constant in this model. In this situation, the cost for call delivery can be estimated as the following:

(1). If the CN sends the first packet to the home address of its MN, the cost for call delivery is given by

$$C_{d1}^2 = 2D_{CN,MN\_HOME}^2 + 2D_{HA,MN\_NEW}^2$$

In this case, the tunnelling takes place between the CN and the MN via a HA.

(2). If the CN sends the first packet to the CoA of its MN indicated in the Binding Cache entry, the cost for call delivery is estimated by

$$C_{d2}^2 = 2D_{CN,MN\_CoA} + C_{d1}^2$$

Before the tunnelling occurs, the packet has to experience a travel from the CN to the MN's CoA at first. It is noted that the CoA becomes an obsolete in this case. The Fast Handover is not considered in this model.

Case 3: It is noted that Case 1 and 2 deal with the scenario of binding lifetime, *T*, that does not expire in a Binding Cache entry during the packet delivery. In contrast, if the lifetime nears expiration, a Binding Update Refresh Request should be sent to the MN as discussed in Section 2. The cost for this binding update is given by

$$C^3 = C_l^3 + C_d^3$$

where $C_d^0 = 0$ and $C_l^3 = 2D_{MN,CN} + 2D_{MN,HA} + D_C^3$. Similar to Case 2, D_C^3 represents the total cost of location update from the MN to its all other CNs.

Based on the discussion above, the total average cost for the binding management with a MN is

$$C = \begin{cases} C,^3 & lifetime-out \\ \alpha C^1 + \beta C^2, & otherwise \end{cases} \quad (2)$$

where $\alpha = \sum_{k=1}^{\infty} kp_k$ and $\beta = p_0$. Recall equation (1), α represents the equilibrium probability of a MN staying within a LA and β denotes the probability of a MN moving out of a LA. Thus, the total average cost for mobility management with a MN during period *t* is formulated as

$$C(t) = \begin{cases} (\alpha C^1 \lambda_c + \beta C^2 \lambda_r)t, & t < T \\ C^3, & otherwise \end{cases} \quad (3)$$

It is noted that C_3 is independent of the lifetime of *t*. By using the equations above, $C(t)$ can be computed.

3.3 Optimal Lifetime

According to the analytical model and the cost computation as discussed above, it is recognised that the total average cost of the binding management with a mobile node is a function of several parameters, which characterize the user mobility, traffic load, network structure, and binding lifetime. In practice, the value of lifetime, *T*, must be specified in the implementation of Mobile IPv6 networks. *T* is a critical parameter for the system performance as argued in Section 1. In order to achieve the best performance, a method for finding the optimal lifetime, T_{op} ,is given as the following:

Rewrite $C(t)$ expression as

$$C(t) = \begin{cases} At, & t \le T \\ C, & otherwise \end{cases} \quad (4)$$

where $A = (\alpha C^1 \lambda_c + \beta C^2 \lambda_r)$ and $C = C^3$. In general $C(t)$ can be plotted in a graph with the two lines, $y = At$ and $y = C$, shown as in Fig.2. The two lines have an intersection at $t = T$. We now argue the value $t = T$ is the optimal choice for the lifetime. Suppose that we choose t_{small} as the lifetime of a binding, the cost $C(t_{small})$ jumps from point A up to B when it expires ($t \ge t_{small}$). It means the resource waste occurs, since $Y = C(t) < C$, $T > t > t_{small}$. Alternatively, if we choose $t_{large} > T$ as the lifetime value of a binding, the cost $C(t_{large})$ will be at point F instead of point E. It also means that an extra cost occurs, comparing with choosing *T* as the lifetime value. According to equation (4), the optimal lifetime value can be determined by $C(t) = At = C$, thus,

$$T_{op} = \frac{C}{A}$$

In other words, the optimal lifetime T_{op} is a function of several factors, expressed as

$$T_{op} = f(R_{CL}, R_{LA}, \lambda_c, \lambda_r, D_{MN,CN}, D_{MN,HA}, D_{MN,CN}^{old}, D_{MN,CN}^{new}),$$

and T_{op} can be evaluated easily based on the equations as given above.

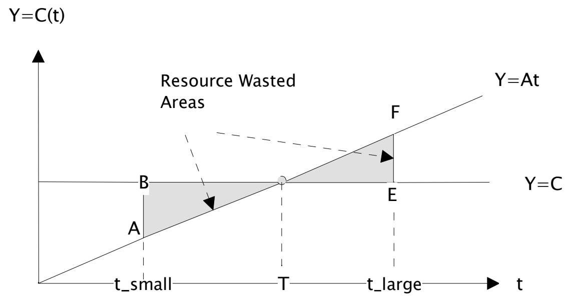

Figure 2: The Optimal Lifetime of a Binding

4. The Algorithm

As discussed in the last section, an optimal lifetime depends on a set of parameters which characterize a use mobility, network structure and traffic load. In order to apply this analytical result in practice, an algorithm for dynamically setting binding lifetime is proposed based on the existing Mobile IPv6 draft by the IETF Mobile IP Working Group [1]. The goal of this algorithm design is to achieve the best system performance with minimal overhead or cost in bandwidth, computation and storage.

The design is divided into two stages. At the first stage, a way to specify all the parameters based on their measures from a MIPv6 network is given. At the second stage, a procedure for setting binding lifetime is proposed in the environment of Mobile IPv6 networks.

Stage 1:

Recall the analytical model above, an optimal lifetime is a function of three kinds of parameters: (i) λ_c represents traffic load to a MN; (ii) λ_r and $D_{i,j}$ characterize a user mobility, where *i* and *j* represents two different MIPv6 nodes, such as MN and HA (or CN) as discussed in Section 3.2; (iii) R_{CL}/R_{LA} reflects a wireless IP cellular network structure. In practice, a mobile node can be responsible for taking measures of λ_c and λ_r. Since it is equipped for counting the number of calls from its CN and the number of crossing cells or subnets, the mobile node can compute λ_c and λ_r periodically. Their formulae are given as below,

$$\lambda_c = \frac{\sum_{i=1}^{n} k_i}{n\bar{t}} \quad \text{and} \quad \lambda_r = \frac{\sum_{i=1}^{n} a_i}{n\bar{t}}$$

where, k_i and a_i represents the number of calls received and the number of crossing cells during each time slot $\bar{t}$ respectively. The period for calculating λ_c and λ_r is $n\bar{t}$. For example, we can specify $\bar{t}$ = 1 hour and n = 24 and thus, the period is a day or 24 hours. Since distances $D_{MN,HA}$ and $D_{MN,CN}$ are measured by the number of hops, the value of TTL field in each MIPv6 packets have to be used. Before a MN sends a Binding Update Message in an IPv6 packet to its CN or HA, the MN has to copy the original TTL value, TL_O, from the packet header into its optional field of the IPv6 packet, which can be accessed by the destination node later. The detail of this procedure will be given in Stage 2. Once the CN or HA receives a binding update packet, it will extract the current TTL value, TL_C, from the packet header. Thus, the distance between nodes, such as $D_{MN,HA}$ and $D_{MN,CN}$, can be simply determined by (TL_C - TL_O).

Usually, the structure or topology of a wireless IP cellular network is managed by the network provider. The configuration of a home agent is also given by the network administrator. Thus, it is assumed that the HA administrators have knowledge about R_{CL}/R_{LA}. For example, we may assume each LA contains 10 cells or subnets as a default value. It is noted that the evaluation of these measures is involved in MNs, CNs and HAs. The question is how to put them together for determining an optimal lifetime, T_{op}, dynamically. This solution is discussed in the next stage.

Stage 2:

In this algorithm, it is proposed to perform calculation of an optimal lifetime in the HA and CN for each binding of a MN. Therefore, both the HA and CN must obtain the values of all relevant parameters before this calculation. This can be achieved by using the existing MIPv6 communication mechanisms, since these values or data, including a lifetime, can be exchanged between a MN and its HA or CN by using the optional field defined in the Binding Update Message and its Acknowledgment Message without any extra packet.

A data structure, called *Lifetime Parameters Block* (*LPB*), is proposed to contain a sequence of data items, which are the values of λ_c, λ_r, R_{CL}/R_{LA}, $D^{Old}_{MN,CN}$, $D^{New}_{MN,CN}$, $D^{Old}_{MN,HA}$, $D^{New}_{MN,HA}$, TL_O and T_{life}. The *LPB* can be linked to the Binding Cache of each MIPv6 node in its implementation. For a MN, the values of λ_c and λ_r in the *LPB* can be generated or updated periodically. The values of other parameters, can be filled in the *LPB* dynamically during data exchange between the MN and its HA or CN, since the data in a *LPB* can be delivered by a Binding Update Message and its Acknowledgement Message between them. By using this communication mechanism the HA and CN of a MN can obtain these data dynamically and perform the calculation for an optimal lifetime. Figure 3 illustrates the format of the *LPB* encapsulated in the optional field of MIPv6 packets. The detailed steps of the data exchange and calculation is described as the following:

λ_c, λ_r, R_{CL}/R_{LA}, $D_{MN,CN}$, $D_{MN,HA}$, TL_O, T_{life}

Figure 3: The format of the *LPB* encapsulated in MIPv6 packets.

1. Whenever a MN sends a Binding Update Message to its HA, the MN will copy its current LPB data into the optional field of the BU message. Especially, the values of TL_O, λ_c and λ_r, are originally generated by the MN as discussed before.

2. Once the HA receives the BU message, it performs two operations: (i) calculate $D_{MN,HA}$ using TL_O and TL_C; (ii) compute T_{life}, since the HA has obtained the values of all the parameters related, except for $D^{New}_{MN,CN}$. An approximate calculation for $D^{New}_{MN,CN}$ may be carried out as the

following: $D^{New}_{MN,CN} \approx D^{Old}_{MN,CN} + d$, where $d = D^{New}_{MN,HA} - D^{Old}_{MN,HA}$. (iii) Set the lifetime calculated into the Binding Cache entry related.

3. The lifetime calculated in the HA is sent back to the MN together with R_{CL}/R_{LA} in the *LPB* format by using BU Acknowledgement Message.

4. Upon receiving the BU ACK Message, the MN updates its relevant *LPB* and also set the lifetime value into the entry of Binding Update list corresponding to this binding update.

5. If a MN has multiple active CNs, denoted as CN_i, where i = 1, 2,...N, in the Binding Cache, it will send the Binding Update Message to them one by one [1]. The Binding Update Message will carry a copy of the latest *LPB* from the MN to CN_i.

6. Once CN_i obtains the BU Message, it calculates D_{MN,CN_i} and then T_{life}.

7. It is similar to Step 3. The lifetime calculated in CN_i is sent to the MN by using BU Acknowledgement Message.

8. It goes to Step 4.

The algorithm design above has some advantages: (i) No extra packets are required and the data exchange is based on the existing MIPv6. Thus, the signalling cost is very low. (ii) The computation for lifetime is distributed in the HA and CNs of a mobile node and thus this releases computing load from the MN since some kind of mobile nodes has limited power in computing, such as mobile phones. The drawback of the design is that $D_{MN,CN}$ has to be evaluated in HA approximately.

5. Numerical Results

As discussed in Section 3 and 4, optimal lifetime, T_{op}, is a function of three kinds of parameters: traffic load (λ_c), user mobility characteristics (λ_r and $D_{i,j}$) and network structure (R_{CL}/R_{LA}). In order to show how these factors impact on the lifetime of a binding, a simulation work based on the analytical model was carried out. Some of numerical results from the simulation are demonstrated in Table 1, 2 and 3.

In Table 1 it is assumed that *R_C/R_{LA}=0.1, $D_{MN,HA}$=5 hops, $D_{MN,CN}$=10 hops*, and *d=4* hops. λ_c is measured as the number of calls per hour and λ_r is measured as the number of cells crossed by the MN per hour. It demonstrates how λ_c and λ_r impact on the optimal lifetime under such assumption. For example, if $\lambda_c = 1.6$ calls per hour and $\lambda_r = 2.0$ cells per hour, the optimal lifetime is 0.50 hour.

Table 1

λ_c	λ_r				
	0.1	2.0	6.0	12.0	20
0.8	1.05	0.96	0.80	0.65	0.51
1.6	0.53	0.50	0.46	0.40	0.35
3.2	0.27	0.26	0.25	0.23	0.21

In Table 2 it is supposed that D_{MN,CN_i} =5 hops, $D_{MN,CN}$ =10 hops, *d=4* hops and λ_c =4 calls per hour. It shows how $R = R_{CL}/R_{LA}$ and λ_r influence the lifetime value under this condition.

Table 2

R	λ_r				
	0.1	0.8	2.4	3.6	4.4
0.05	4.78	1.02	0.37	0.25	0.20
0.1	3.14	0.87	0.33	0.22	0.19
0.4	1.03	0.46	0.20	0.14	0.12

Finally, Table 3 gives the numerical result in order to investigate how d and λ_r impact on the lifetime value. It is assumed that *R_C/R_{LA}=0.1, $D_{MN,HA}$=5* hops, *$D_{MN,CN}$=10* hops, and λ_c = 4 calls per hour.

Table 3

d	λ_r				
	0.1	2.0	6.0	12.0	29.0
4	0.24	0.23	0.21	0.18	0.16
8	0.22	0.22	0.20	0.18	0.16
12	0.20	0.19	0.18	0.16	0.14

Conclusions

In order to find a way to specify the optimal lifetime of a binding in Mobile IPv6 networks, a mathematical model is introduced based on imbedded Markov chain. By using this model, the cost of a mobile node for the binding management can be estimated and the optimal lifetime of a binding associated with the mobile node can be approximately determined. As an application of the analytical model in MIPv6 networks, an algorithm for dynamically setting the binding lifetime is proposed in context of existing Mobile IPv6 standards. The numerical results are also demonstrated.

Further work is needed to investigate the performance of the algorithm proposed.

References

[1] David B. Johnson, Charles E. Perkins and Jari Arkko, “Mobility Support in IPv6 <draft-ietf-mobileip-ipv6-18.txt>, *IETF Mobile IP Working Group*, 1 June 2002.

[2] Hesham Soliman et al., “Hierarchical Mobile IPv6 mobility management (HMIPv6)”, *IETF Mobile IP Working Group*, October 2002.

[3] Rajeev Koodli, “Fast Handovers for Mobile IPv6 <draft-ietf-mobileip-fast-mipv6-05.txt>, *IETF Mobile IP Working Group*, 30 September 2002.

[4] Hemant Chaskar, “Requirements of a QoS Solution for Mobile IP”, <draft-ietf-mobileip-qos-requirements-03.txt>, *IETF Mobile IP Working Group*, 30 July 2002

[5] J. Gomez and A. Campbell, “An Overview of Micro-Mobility Protocols in the IETF”, *ACM SIGMOBILE* MC2R, 2001.

[6] Campbell, J. Gomez, C-Y. Wan, Z. Turanyi, and A. Valko, “Cellular IP”, *Internet Draft*, IETF, 1999.

[7] C. Castelluccia, “Extending Mobile IP with Adaptive Individual Paging: A Performance Analysis”, *INRIA*, Nov. 1999.

[8] Zhang, X., Gomez, J. and A. T. Campbell, "P-MIP: Paging Extensions for Mobile IP Paging" , *ACM Journal* on Mobile Networks and Applications (MONET), Vol 7, No. 2, March 2002.

[9] J. Schuringa, “Performance Modelling of Location Management Strategy in Mobile Networks”, *Master Thesis* 1995, Dept. of Computer Science, University of Twente.