Mobile Information Systems 2 (2006) 21–50
IOS Press

Modelling and quantitative analysis of LTRACK – A novel mobility management algorithm

Benedek Kovács, M. Szalay* and S. Imre
Department of Telecommunications, Budapest University of Technology and Economics, Pf. 91,
H-1521 Budapest, Hungary

Abstract. This paper discusses the improvements and parameter optimization issues of LTRACK, a recently proposed mobility management algorithm. Mathematical modelling of the algorithm and the behavior of the Mobile Node (MN) are used to optimize the parameters of LTRACK. A numerical method is given to determine the optimal values of the parameters. Markov chains are used to model both the base algorithm and the so-called *loop removal* effect.

An extended qualitative and quantitative analysis is carried out to compare LTRACK to existing handover mechanisms such as MIP, Hierarchical Mobile IP (HMIP), Dynamic Hierarchical Mobility Management Strategy (DHMIP), Telecommunication Enhanced Mobile IP (TeleMIP), Cellular IP (CIP) and HAWAII. LTRACK is sensitive to network topology and MN behavior so MN movement modelling is also introduced and discussed with different topologies.

The techniques presented here can not only be used to model the LTRACK algorithm, but other algorithms too. There are many discussions and calculations to support our mathematical model to prove that it is adequate in many cases. The model is valid on various network levels, scalable vertically in the ISO-OSI layers and also scales well with the number of network elements.

Keywords: LTRACK, mobility management, location tracking, Markov chain

1. Introduction

As the world goes mobile, there are more and more multimedia services that can be accessed using wireless technology. In 3G networks the core network, (and perhaps also the access networks) are based on the IP. For this reason, many IP mobility management algorithms were developed. The most common ones are Mobile IPv4 and Mobile IPv6. These solutions have weaknesses such as scalability issues and signalling overload on the network.

The LTRACK algorithm can reduce signalling load on the IP backbone by introducing a new handover management strategy. The basic modelling of the algorithm and the comparison to other approaches is presented in this paper using a Markov chain model which is extended and the effect of *loop removal* is also discussed.

The advantages of using LTRACK are shown by comparing it to other mobility management protocols such as Hierarchical Mobile IP-like solutions. Comparison is made using various network setups and

---

[*] Corresponding author. Tel.: +36 20 3187012; Fax: +36 1 4633263; E-mail: szalaym@hit.bme.hu.

scenarios. It is also considered how network, traffic and user mobility parameters affect the performance of our algorithm.

This paper is structured as follows:

After introducing general IP mobility issues, and especially Mobile IP in Section 2, we provide an overview of existing solutions that improve the performance of Mobile IP in Section 3. Then, in Section 4 our new algorithm, LTRACK is introduced. In Section 5 we introduce our own network and mobility models that are going to be used in the following sections. Later, our Markov chain model for LTRACK handovers is introduced in Section 6. This model can be used to describe other mobility solutions, not just LTRACK, this is also discussed. Section 7 describes a cost function for LTRACK and other mobility algorithms, and presents how it can be used to compute the gain of LTRACK over other algorithms, and how various parameters of LTRACK can be optimized. Finally, Section 8 outlines our conclusions and future work.

2. IP mobility and Mobile IP

2.1. Mobility

The need for mobile communication is essential and one of the most rapidly growing areas in our present world. In the past century the use of portable phones became widespread. Cellular and other mobile services started to extend simple voice communications with data communication and a lot of protocols were introduced to provide data mobility. The Internet infrastructure has also grown and made us able to communicate large quantities of information between static devices.

These two together naturally ended up in a request for the information shared on the Web to be accessible for mobile hosts as well. Because of this, the need to make the IP protocol capable of mobile usage came up as an evident idea.

One may find many solutions for such a problem in different ISO-OSI network layers, namely on the network, transport or application layer. New layers or semi-layers can also be introduced. The request of application transparency is almost evident so that the user should be able to use the same application as he/she used for static Internet.

It is important that the suggested protocol for global IP mobility should not be dependent on some specific access system like CDPD (Cellular Digital Packet Data) or GPRS (General Packet Radio Service). The design of Mobile IP protocol is aware of these requirements.

2.2. The Mobile IPv4 protocol

Regarding the static Internet, IP packets are routed depending only on their IP designation address. In this manner, the IP address is used to route the packet. On the other hand, it is also used to identify a host or network. Having a mobile environment, the node identified by an IP address might move away while the packets are still delivered along the same static route. To resolve this problem in the Mobile IP protocol, two IP addresses are used. One to identify the host and another one for routing purposes. With this strategy the identification role of the IP address is kept while the other address can be used for routing.

2.2.1. Terminology

Before getting into more details, a few terms are defined to provide clear understanding.

- *Mobile IP* (*MIP*): This term is used to identify the Mobile IP [6–8,11,16,23] protocol. (In many other works this term may refer to all the IP mobility protocols.)
- *Mobile Node* (*MN*): The mobile node is the IP node that changes its point of connection to the IP network. The mobile node has a unique IP address that identifies it. This address can be used by the DNS servers and anyone who wants to communicate with the MN.
- *Home Address*: The IP address identifying the MN.
- *Care-of Address* (*CoA*): This is a temporary IP address used to deliver the packets to the MN as the end point of a tunnel. It is important to note that this address is used for routing.
- *Home Agent* (*HA*): This is a network entity in the home network of the MN that maintains a database where the records contain the Home Address assigned with the current location information of the MN.
- *Foreign Agent* (*FA*): A router that cooperates with the HA and the MN to get the packets delivered.
- *Correspondent Node* (*CN*): Either a mobile or static end system that communicates with the MN.
- *Home Network* (*HN*): This is the network where the MN is registered.
- *Foreign Network* (*FN*): A network where the MN can possibly roam to. link. (This is like the Media Access Control (MAC) address.)
- *Mobility Agent*: Either a Home Agent or a Foreign Agent.
- *Node*: A host or a router.
- *Visited Network*: The network where the MN is currently connected except if it is its home network.

2.2.2. Mobile IP protocol structure

The protocol consists of three different sub-protocols.

The *Agent Discovery* is where Mobility Agents advertise their availability on each link on which they provide service. After receiving such an advertisement the MN decides whether it is in the Home Network or in a Foreign Network. (We do not discuss the case when the HN is detected since then no mobility support is needed.) For the second step, the *Mobile Node Registers* its Care-of Address with its Home Agent. The CoA is obtained from the FN either with any external assignment mechanism or with the advertisements received. For more details on *Care of Addresses*, see [11].

When a correspondent node wants to communicate with the MN, it sends the datagrams to its Home Address. The Home Agent intercepts the communication and *tunnels* the datagrams to the Care-of Address of the MN. The endpoint of the tunnel can be either a Foreign Agent or the Mobile Node itself. If it is delivered to a FA, the FA has to forward the packets to the MN.

When the MN replies to the Correspondent Node it sends the packets directly to it. This is called the *Triangle Routing*. (It is useful to note that if the CN is a mobile node as well, the protocol automatically works since having a mobile or a static node as a peer is indifferent. This can be considered as one of the main advantages of MIP.)

This routing system is far from optimal and there are many proposals to make it more effective. (One can think about an MN near to the CN but roamed to a network far form its HN.) Updating *Mobility Bindings* by providing the CN with the CoA of the MN would shorten the path for the datagrams to be delivered. This is called *route optimization* and unfortunately requiring many security issues to be solved. It is also clear that the CN has to be capable to this protocol that is evidently a disadvantage of such an extension. (Also there is the problem of smooth handoffs that has to be solved and additional bindings from the FA to the MN has to be applied.)

Tunnelling the packets from the HA to the CoA also provides basic security for the communicating entities since the real destination address (the Home Address of the MN) is hidden from any intervening routers. This is useful because the location of the MN is hidden from many unauthorized networks elements. However, security issues still need further considerations since a corrupt FA can still easily access to possibly confident information.

2.3. Mobile IPv6 [8,23]

The Mobile IPv6 provides IP mobility over IPv6. It uses the advantages of the newer IPv6 protocol and also the experiences gained from the Mobile IPv4 are taken into consideration.

2.3.1. Advantages of Mobile IPv6 over Mobile IPv4

- First of all the available number of possible IP addresses to be used are large enough. In Mobile IPv4 the address shortage could lead to many problems, when assigning a *collocated care-of address*. This is discussed above in Section 2.2 and also in [11].
- Secondly, it is easier to find solution for *route optimization* because the security abilities of IPv6 are much more sophisticated than that of IPv4. Route optimization is a part of Mobile IPv6, while it is just an extension of Mobile IPv4.) The usage of IPv6 Destination Options [**?**] are used to exchange additional information.
- Thirdly, the usage of IPv6 header instead of tunneling the packets is reducing the header size that increases performance.
- Finally, IPv6 does not need Address Resolution Protocol (ARP) Neighbor Discovery since it is included.

2.3.2. Terms

- *binding*: The association between the Home Address and the Care-of Address of the Mobile Node.
- *registration*: When the Mobile Node sends Binding Update to the Home Agent or a Correspondent Node.
- *binding authorization*: Correspondent registration needs to be authorized.

2.3.3. The basics of Mobile IPv6

As it is described earlier, in case of Mobile IPv4, when the Mobile Node is away from home, a Care-of Address is assigned to it, and can be reached transparently on his Home Address.

When the Mobile Node gets new CoA, it *registers* a Binding with a router in his home network. This association is recorded in the *Binding Cache* of the router. The MN now can be reached through the HA with IPv6 encapsulated, tunneled packets. This association has an expiration time.

The MN can also send *Binding Update* to the Correspondent node and with this additional *binding* the packets sent can be delivered directly to the MN. During this operation authorization of the MN and the CN is done using the IPv6 header.

A loss of these binding updates may occur when the CN is moving as well. If we take a short look at such a scenario, it can be seen that the *Binding Update* sent to a moving node might arrive to the former location of it and the communication between the two nodes will be lost [14].

2.4. Mobility layers

Hierarchy can also be introduced into the mobility scheme. In this Section the difference between Micro and Macro-mobility (in some works even Global-mobility) is described mostly theoretically and without getting into details.

If we consider the Internet as a huge globe-size network, one can agree that local management systems are essential to make IP mobility work. This is also the case with the Mobile IP kind of protocols. However, both types presented above, can work at large-scale without any further limitations but there are some problems. They do not scale very well, handovers can take very long (up to a few seconds), so they are not suited to always-on scenarios. These problems led to the introduction of Mobility-layers. These layers are called Micro-, Macro- and even Pico- and Global-mobility layers. There can be many reasonable ways to differentiate between them.

In some works Micro-mobility is referred to as the intra-network mobility and Macro-mobility as inter network-mobility in one region while Global-mobility then refers to the movements between these geographical regions [20]. In other works, Micro-mobility network means the part of the Internet that is locally close to the node [3]. Another approach is also possible when Micro-mobility area can be considered as a virtual subnetwork of the node that can be large geographically.

2.4.1. Micro-mobility protocols

Micro-mobility is also very important to provide real-time services to mobile nodes. The delay and delay variation (jitter) gets evidently worse as the network grows. Micro-mobility protocols try to reduce this effect with local management of the packet delivery. These parameters (delay and jitter) are mainly used to determine the Quality of Service (QoS) in IP networks especially in Voice over IP (VoIP) applications.

There are two well-known solutions that address the same problem. The Cellular IP (CellIP) [1] and the Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) [19] protocols are introduced to provide not just better QoS but, as an essential requirement, also a fast and effective handover (or handoff). When examining the protocols, it is clear that they can be used locally in regional systems. They are domain based Micro-mobility supporting protocols [3].

The advantages and disadvantages of such protocols are discussed in [2] via examining their efficiency, simulating and comparing them to each other.

2.4.2. Macro-mobility, global-mobility protocols

The classical mobility management protocols such as Mobile IPv4, Mobile IPv6 can be considered as the Macro–mobility or Global–mobility protocols.

In this structure they handle a kind of inter domain mobility between the Micro-mobility domains. The whole network now can be considered as having a hierarchical structure.

2.4.3. Other protocols

There are many mobility management extension protocols to Mobile IPv4 and Mobile IPv6. Some of these solutions can easily be classified to the layer structure given above, and there are also others that can be used in several layers. In this paper it is also our aim to find and show the layer(s) or layer structure where LTRACK can be used the most efficiently.

In our discussion the use of multiple mobility layers will result n a tree-like approach when modelling the backbone network where the levels of the tree might be assigned to a specific mobility layer.

3. Existing solutions to improve Mobile IP

In the MIP structure we have an IP host which always sends the request for location update to the Home Agent, and the Home Agent maintains an up-to-date database that contains the location of the Mobile Node. Since the HA has to know the exact location of the MN in order to deliver the request, in the case of MIP basic solution messages are sent after each handover to maintain the database of the routers and the HA. This mechanism has the advantage of simplicity and has an obvious model and a simple cost function as we will see.

Although this handover management is simple, it is not cost-effective. The application of such a protocol puts extraordinary high signalling load on the bearer network(s). As we will see, this cost can be drastically decreased by using a different algorithm for location updates.

In this Section, mobility management algorithms, that were introduced by others are presented and organized in three categories. The three categories are: the hierarchical-like, the cellular-like and the location tracking-like solutions. Solutions belonging to these categories are all targeted at reducing the cost of Mobile IP, but they follow a different approach, as there can be many types of reductions since it strongly depends on the network model used.

3.1. Hierarchical-like solutions

There are regional location management system proposals to reduce the signalling traffic by maintaining the MN Care-of Address locally [5,9,20]. To do this, there is a special node called Mobility Agent (MA) acting as a local HA for the node and managing it in a Micro-mobility approach.

In our terminology "Hierarchical Mobile IP" covers HMIP [9] and a whole group of other protocols that have similar attributes. They all use mobility layers, so we can consider the HMIP network structure as a tree of Mobility Agents (MA) with the HA in the root of the tree.

3.1.1. Hierarchical mobile IP protocol

Now we discuss how we will examine the Hierarchical Mobile IP (HMIP) [9]. The main idea in the Hierarchical Mobile IP is that the *Location Update* information is sent only to the nearest MA (router GFA) on the network if the MN moves within the subnetwork managed by this entity (Fig. 1).

The basic operation is that a new PCoA (Personal Care-of Address) is assigned to the node within the subnetwork and changes when the node changes its point of attachment (FA). Its original CoA – in the HMIP scheme it is called the Virtual Care-of Address (VCoA), – remains unchanged unless the MN moves out from the management area of the given MA, see [9].

Let us now take a look at the network tree of MAs of the hierarchical mobility. One path from the HA leads to the old FA of the MN and another one leads to the new one. It is clear that it is enough to send the location update message to the nearest common router with the old path and the HA can allocate the MN properly. However, typically there are FAs between the MAs. Now we take the best case for the signalling optimization and consider every router as an MA. This is how we will examine the HMIP handover protocol.

3.1.2. Dynamic hierarchical mobile IP protocol

This protocol, introduced in [12,13,24], will be discussed similarly to the HMIP in this paper. The reason is that all the nodes in the network graph are treated as MAs, more specifically GFAs, because we count only the messages sent between them.

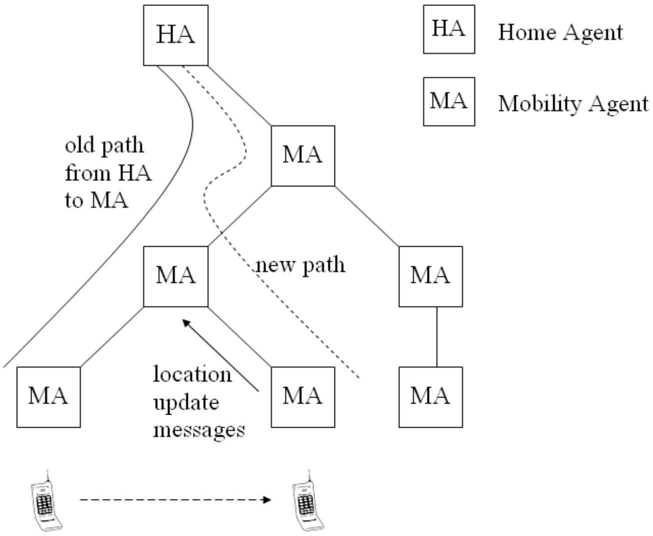


Fig. 1. Basic operation of HMIP.

The HMIP approach does not exploit the advantage of possible direct links between its possible MAs. The Dynamical Hierarchical Mobile IP protocol has dynamically created hierarchical mobility domains. That means when an optimal set of FAs have been visited by the MN under the same MA, it changes its VCoA and sends Location Update to a new MA. Still there is signalling on the backbone network in every case the MA is changed for an MN not like – as we will see later – in the case of LTRACK.

3.1.3. *Telecommunication enhanced mobile IP* (*TeleMIP*)

This is a version of HMIP. It uses the same multiple CoA solution to provide Micro-mobility services for the Mobile Node. From our point of view, it has the same cost functions as HMIP. The differences of TeleMIP from HMIP can be handled the same way as it is described in the former (HMIP vs. DHMIP case).

3.2. *Cellular-like solutions – Cellular IP*

There are other, cellular-like solutions for this problem. One example is Cellular IP (CIP) [1] that uses the cellular-like Location Area and paging technique to locate the Mobile Nodes that are in idle mode. This saves a lot of signalling traffic but on the other hand, the exact location of the MN has to be determined by paging when there is a communication request. Paging is usually an expensive method in the sense of signalling cost.

The idea behind this kind of approach comes from the GSM protocol. Basically CIP is a Micro-mobility protocol and it is rather difficult to enhance it to other layers. It is compared to a location tracking like solution called the Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) in [2] and turns out to be an efficient solution to handle mobility within an access network.

Since this protocol is not yet discussed within higher mobility layers it cannot be compared to LTRACK directly. One direction of our further research is to examine how the cellular-like solutions could be enhanced to and optimized in case of multiple mobility layers.

3.3. Location tracking-like solutions

The idea behind our Location tracking-like algorithms is to find an optimal compromise between the hierarchical-like and the cellular-like ones. However, HAWAII works with a same kind of location tracking method in Micro-mobility. This means that when there is a handover, the packets are neither broadcasted nor duplicated but sent from the old point of attachment of the MN to its new one, thus the location of the node is “tracked”.

One other example for such a method is the Three-location area (TrLA) [17] management algorithm but it was developed for Personal Communications Systems (PCS) thus this protocol is slightly different from the Mobile IP-like ones and it would be inappropriate to discuss it further here.

4. LTRACK - Location Tracking

Our proposal, LTRACK is a mobility management solution introduced in 2003 [21,22]. The LTRACK protocol works over a network of LTRACK nodes. An LTRACK node is a logical entity, it can be a base station of the network, or it can be a small subnetwork where mobility management is handled locally, see Section 2.4.

In the LTRACK mobility management scheme the mobile node is always connected to one LTRACK node. When it changes its point of connection, a handover takes place.

Each mobile node of the LTRACK network has an entry in a register called the Home LTRACK Register (HLTR). This register is similar to the Home Location Register of GSM networks or the Home Agent of the Mobile IP scheme. In the GSM mobility management scheme, if the mobile node is in *idle* state, then only the Location Area is known, which the mobile node is staying in, but no exact location information is stored. In the Mobile IP scheme the exact location information is stored in the Home Agent, and is updated frequently. The LTRACK scheme is somewhere between these two solutions.

For each of the mobile nodes, the HLTR stores the address where it received location update message from. It is the address of an LTRACK node, and is a next-hop towards the mobile node. The mobile node is either still connected to that LTRACK node, or that LTRACK node knows another next-hop LTRACK node towards the mobile. Finally, the mobile node can be found at the end of a chain of LTRACK nodes.

When the mobile node moves from one LTRACK node to another, handover takes place. The LTRACK node that the mobile node moves away from is called the old LTRACK node, the one it moves to is called the new LTRACK node.

There are two different kinds of handover in LTRACK: normal handover and tracking handover. In a normal handover the mobile equipment updates its entry in the HLTR by sending the address of the new LTRACK node to it. In case of a tracking handover the mobile sends the address of the new LTRACK node to the old LTRACK node over the air interface that is an important difference from the previous location tracking-like algorithms.

If the mobile node can only communicate to one LTRACK node (hard handover), the address of the new LTRACK node has to be sent to the old LTRACK node just before takes place. On the other hand, if the LTRACK node is capable of communicating to more than one LTRACK node simultaneously (soft handover), then the address can be sent any time during the handover. The handover protocols in most of the modern technologies would allow the MN to give information to its former point of attachment without any more significant traffic through the air interface. However if this can not be done the LTRACK node should make a normal handover when it moves to the new station.

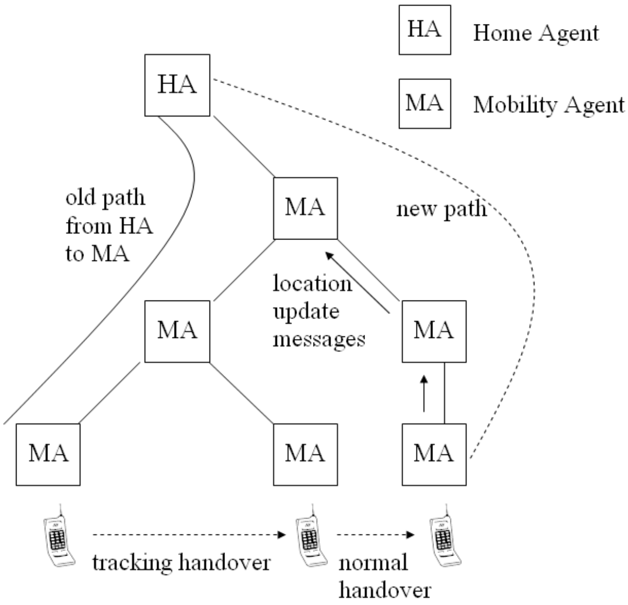


Fig. 2. LTRACK normal and tracking handovers.

The normal handover is similar to a Mobile IP handover: it generates a lot of signalling traffic. A tracking handover does not put any signalling load on the network, but if an incoming call (or incoming packet) arrives to the mobile node, we have to find it in a hop-by-hop manner which is expensive. Every handover can be either a normal handover or a tracking handover, see Fig. 2.

If only normal handovers are used, the location management scheme becomes very similar to the Mobile IP scheme. In the LTRACK scheme a normal handover can be followed by some tracking handovers before another normal handover takes place. If the mobile node does not receive a call (or packet) between two normal handovers then less signalling is used, if a call (or packet) is received after some tracking handovers, then more signalling is used compared to a normal-handover-only scheme.

Thus the most important decision of LTRACK, as it is also for all the *location tracking*-like algorithms, is when to make a normal handover and when to make a tracking handover. In the previous works [15,18, 24] the number of handovers to be take was derived from the signalling cost parameters of the network and the mobile movement models. We propose a method using Markov chains that is an alternative way to describe the behavior of these protocols. Our network model will also improve these examinations.

5. Modelling the IP backbone and the behavior of the MN

In this Section our model of the network is described. This preambles the model used for examining the handover algorithms.

In many works the network topology is fixed; for example the GSM cell structure is used for the analysis in [17]. The disadvantage of such an approach is that it uses a network structure which is too specific thus unapplicable for the IP backbone. Other works use an average cost values for a network and define them as a constant. This approach can handle a more general network setup but hidden relations between the cost of different tasks are neglected.

We suggest that one should use at least two parameters which are derived from the two kinds of approaches above and those together could provide a more exact way to describe the different networks.

The network will be considered as a simple graph without loop edges or multiple edges. The edges are the network links, vertices are the network nodes. Either all or a special subset of these can function as an Access Point, Foreign Agent, a simple router or even a Home Agent. It is also possible that the network is modelled on different layers. For example a node in the given graph can represent a simple router while in another model it might cover a whole subnetwork. With our approach, different network structures and parameters can be used to differentiate between these cases.

We have a specific graph describing the structure of the network on a given layer. The parameters describing the graph will be discussed generally and will also be computed for specific examples. Since most mobility protocols use a tree topology network, we will model it, and we will also model the fully linked (or meshed) topology.

It is important to point out that one advantage of the LTRACK cost model presented later in this paper is that it takes the network topology as a parameter. This enables us to calculate the costs using different network parameters which may be derived from the model of different networks. Since – as it will be addressed later – the handover cost is heavily dependent on the network structure, it is impossible to calculate it generally although most of the papers discuss how to determine the cost for location registration or packet delivery network independently [4,12,13,24]. As it will be shown, discussions in the related works are somehow similar to our fully linked topology case.

In our terminology the distance between two nodes in the graph is the length of the shortest path leading from one to another. We define the distance of the MN from the HA as the distance of its access point (graph node) from the HA.

5.1. Parameters describing the network

We say that having possible access points, FAs and the network, it is a reasonable assumption that the probability of a MN moving from one access point to another is the same in both directions. Thus:

$$Prob(\mathrm{FA}_A \to \mathrm{FA}_B) = Prob(\mathrm{FA}_B \to \mathrm{FA}_A), \tag{1}$$

where → stands for the movement from one FA to another. (This rule might not hold in a very few special cases. To discuss them is out of our scope.)

This assumption makes us able to have a constant parameter describing the distance of the MN from the HA because moving close has the same probability of moving further away from it. (It will be shown that if this parameter varies over time or position, the parameters of LTRACK will rely upon this dependency but the model can still be applied.)

After accepting these assumptions we only have to deal with movements on the same average depth level that will be denoted by $m \in \mathbb{R}$. The second graph parameter g is obtained as follows:

Let us consider the shortest path leading from the old FA_A and the new FA_B of the MN, to its HA. Let $g \in \mathbb{R}$ denote the length of the average shortest path leading from FA_B to the nearest node in the old path. (One will realize that it is equal to the number of signalling messages sent via the links by an optimal HMIP-like protocol. This will be described in an upcoming section.) Note that because of the Eq. (1) from our assumption above: $\|\mathrm{FA}_A, \mathrm{HA}\| = \|\mathrm{FA}_B, \mathrm{HA}\| = m$.

It is obvious that $1 \leqslant g \leqslant m$. The reason for choosing such a parameter is the mechanism of the LTRACK algorithm. When the call is routed “hop-by-hop” on a tree-like topology, $2g$ edges are involved since the signalling has to go “up” to the nearest common router just like when there is a handover in the HMIP case and then "down" to the new FA. Since the network structure is static there is no need to update the routing tables of the routers because the destination of the packet is the "next-hop" FA. According to this fact there is no need for additional processing when there is a "tracking handover". This makes the LTRACK different from protocols like introduced in: [18,19,24].

5.1.1. *Tree and tree-like network topologies*

The tree topology is one of the most widely used one when considering mobility networks, because most mobility management algorithms use a physical or logical tree topology network. After restricting the discussion to trees we will also generalize our model.

While using an average depth level m, we also introduce an average value of the degree of the network nodes: $\delta \in \mathbb{R}$. Here, we still can have many different models for the network. One is that the FAs are located only at the leaves of the tree. We can say that the MN moves to each neighboring cell with the same probability and each cell has δ neighbors so we suppose that MN moves in a δ range in one step, so we consider that it can move to δ neighboring FA's area with the same probability.

The aim now is to calculate the average number of new graph edges involved in the signalling in case of a handover. This parameter is denoted by g.

To calculate g we differentiate the possible movements upon their costs. We say that that the MN has δ^i ways to move between two leaves under different layer $i-1$ nodes, of which:

- 1 new edge is involved: δ^m (Circularly between the leaves under an $m-1$ layer node.)
- 2 new edges are involved: δ^{m-1} (Circularly between the $m-1$ layer nodes under an $m-2$ layer node)
- ...
- i new edges are involved: $\delta^{m-(i+1)}$ (Circularly between the $m-(i+1)$ layer nodes under an $m-i$ layer node)
- ...
- m new edges are involved: δ (Circularly between the 2 layer nodes under an 1 layer node)

$$g = \frac{1\delta^m + 2\delta^{m-1} + 3\delta^{m-2} + ... + m\delta}{\delta^1 + \delta^2 + \delta^3 + ... + \delta^m} = \frac{\delta}{\delta - 1} - \frac{m}{\delta^m - 1}, \tag{2}$$

and the whole graph $G(h)$ involved in an h movement of the MN: $G(h) = gh + m$.

To compare our achievements to others' [12,13,24] we have to mention that our variable g is related to some given constants in previous works, for example m_{ff_u} "transmission cost of location update between FAs" and m_{fh_u} The "transmission cost of location update between an FA and the HA". Comparing the different approaches we should say that $g = 2m_{ff\_u}$ and $m = 2m_{fh\_u} = m \cdot 2m_{ff\_u}$ from [24]. The performance of the algorithm using g and the values above as additional constants will be discussed in Section 7.4.

5.1.2. *Fully linked network graph approach*

After discussing the effectiveness of a tree topology, we examine another widely used one. This is the *fully linked* graph. It is a topology where the nodes that are geographically close to each other are connected. If all the neighboring nodes of a tree topology mobility network are connected, the result is a fully linked graph.

It is straightforward that in a fully linked graph g is close to 1. Since we do not use *absolute* cost values, m is close to g and δ loses its meaning. It means that we are always likely to have a link between the former and the new point of attachment.

A model, similar to this is used to examine the performance of the DHMIP algorithm [12,13,24] because the costs are constant between each pair of nodes (except for the Home Agent). However, it will be shown in Section 7.4 that it is crucial to make g dependent of m to discuss LTRACK correctly.

5.2. Modelling the MN movement and packet delivery

Here, our model for the MN movement and the packet delivery will be presented. We say that in a given time interval the MN changes its point of attachment with a constant rate. This will be modelled with a homogenous Poisson process as in other works [24,27]. Let λ denote the parameter of the Poisson process and so denote the rate of handovers of an MN.

The other parameter that can be introduced in a similar manner is the rate of receiving a call: μ. This can also be time- or location-dependent but we assume it is constant for the examined time interval.

Let us introduce ρ as the "mobility ratio" meaning the probability that the MN changes its FA before a call arrives. Here $\rho = \frac{\lambda}{\lambda+\mu}$. In [27] rho is defined like $\rho = 1 - \alpha(0)$. (Note that the ρ in the two works are not the same! The ρ in [27] would be $\rho = \frac{\mu}{\lambda}$ with our notation.)

In real systems it might be not true that the mobility ratio is homogenous. Parameters μ and λ might depend on the position. Our model does not count with this scenario. However, we can say that these constant values are some average values and valid for a given network region in a given time interval.

5.3. Theoretical background of the cost function

In this paper the cost will be derived from the weighted number of messages sent between the nodes in the network. Basically our cost function is the following:

$$C_{\text{ALGORITHM}} = p_h \cdot e_u \cdot C_u + p_c \cdot e_d \cdot C_d,$$

where

- p_h: Prob("Making a handover");
- p_c: Prob("Receiving a call");
- c_u: Cost of update on one edge;
- c_d: Cost of delivery on one edge;
- e_u: Number of edges involved for the update;
- e_d: Number of edges involved for the delivery.

One can see that the cost function we use is a simple expected value calculation. We have the obvious probability field (Ω, A, P) of two possible events: "Making a handover" and "Receiving a call", where $p_h + p_c = 1$. We say that a discrete probability field can be used to determine the optimal number of handovers for our proposal and using it, we are able to compare our LTRACK algorithm to others. The reason is that there is no signalling when the MN is neither paged nor a handover takes place.

The number of edges involved will be derived from the network models above using the parameters g and m while the probabilities will be derived using the model of the algorithm. The mobility ratio ρ and the handover/call rate λ/μ will be used to determine the probabilities of receiving a call or a packet.

Additional parameters are introduced such as the cost of update and delivery on an edge. These will be homogenous all over the network:

- c_u: The cost of update per edge between hops,
- c_d: The cost of delivery per edge between hops.

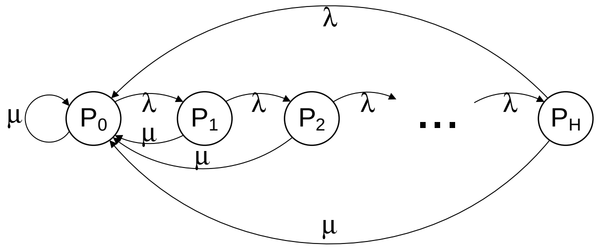


Fig. 3. Markov chain model for LTRACK.

There are two main reasons for the introduction of these:

- One is that the cost of the two different processes might be different.
- The other is that the cost of these operations might be different for each protocol for the same kind of network structures with the same g and m parameters.

6. Handover model of LTRACK

Our model is based on Markov Chains which is a widely used technique for mathematical representation of problems in different fields of telecommunication. Unlike in [17] where a semi-Markov process is used to evaluate a Personal Communications Service (PCS) location-tracking algorithm: Three-location area (TrLA) and where each state of the Markov Chain modelled a cell in the network, in our work the modelling will only focus on the state of the MN according to the LTRACK algorithm.

6.1. Basic model of the algorithm

In this Subsection we present an essential, simplified model of our algorithm. We are considering the mobile station with its two possible events (see Fig. 3):

1. One is a handover. It is when the MN moves and changes its FA. When there is a "tracking handover", the next-hop path of LTRACK becomes longer and there is a change in the state of the MN (from P_i to P_{i+1}). Variable λ is the parameter of a Poisson process. After reaching a threshold (at the $(H+1)$th movement), LTRACK makes a "normal handover" so returns to the P_0 state with a Location Update.
2. The second event is when the MN receives a call (or incoming packet). As it was defined in Section 5.2, let μ denote the rate of receiving a packet or call according to a Poisson process. When a call is received a Location update is made according to the LTRACK algorithm. This means that the MN returns to the P_0 state.

Figure 3 depicts the states of the MN assuming the fact that the rate of moving to a new cell and receiving a packet are both homogeneous Poisson processes. (Otherwise there are different λ and μ values for each change. We will discuss a slightly similar case in Section 6.3 where the *loop removal* is added to the model.)

One can say that this assumption of homogenity might not hold because a MN usually does not move or receive a call at night as frequently as during the day. However, in our case we never said that the λ and μ parameters depend only on the MN. They can also depend on the time of the day or on some special property of the subscriber, but they usually can be considered constant for a period of time which is longer than the time between two normal handovers.

While the MN is in P_i state it means that there was no call received and the MN stayed in the area of the same FA. The problem of the ongoing calls, while the location area is changed, is handled similarly to the HAWAII protocol. We say that both algorithms examined should work as HAWAII when there are ongoing calls [2,19].

6.2. The stationary distribution of the MN states

The matrix representation of our Markov Chain can be derived from Fig. 3.

$$\mathbf{Q} = \begin{pmatrix} -\lambda+\mu-\mu & \lambda & 0 & 0 \cdots & 0 \\ \mu & -(\lambda+\mu) & \lambda & 0 \cdots & 0 \\ \mu & 0 & -(\lambda+\mu) & \lambda \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots \ddots & \vdots \\ \mu+\lambda & 0 & 0 & \cdots\cdots & -(\lambda+\mu) \end{pmatrix} \tag{3}$$

The cost of the LTRACK algorithm can be computed by summing up the probability of each movement between states in the Markov Chain multiplied by its rate. (In this paper the cost is basically the number of edges where signalling was transmitted in the network graph.)

Our finite Markov Chain is irreducible and aperiodic so there is a stationary state $\mathbf{D}_\infty$. This state is reached from any initial $\mathbf{D_0}$ distribution. Theoretically, the probability of the states can be derived from the following equations:

$$\Pi \cdot \mathbf{D}_\infty = \mathbf{D}_\infty, \sum_{i=0}^{H} P_i = 1, \tag{4}$$

where $\mathbf{D}_\infty$ is a vector of P_is. The following substitution can be used: $\rho = \frac{\lambda}{\lambda+\mu}$. Recall, that ρ, the "mobility ratio" means the probability that the MN changes its FA before a call arrives.

It is not useful for us to use the equation above for computing $\mathbf{D}_\infty$. However, we can also use the rate matrix (Q) description of the network. We compute the stationary distribution ($\mathbf{D}_\infty$) as a function of H.

$$P_0 = \frac{\rho-1}{\rho^{H+1}-1}, P_i = P_0\rho^i, P_H = \rho^H \frac{\rho-1}{\rho^{H+1}-1}. \tag{5}$$

The mean value of the point of return h_r with a call arrived under a given H should be determined as:

$$M[h_r] = \sum_{i=0}^{H} P_i \cdot i = \frac{\rho}{(1-\rho)^2}(1-(H+1)\rho^H + H\rho^{H+1}). \tag{6}$$

This denotes the expected number of "tracking handovers" made until a call arrives as long as no LTRACK "normal handover" has occurred. (If we have a mean value of the cost of each movement in the Markov Chain, the mean cost of receiving a call can also be calculated.)

6.3. Extended handover model with loop removal

In this Section a special phenomenon of LTRACK will be added to our model. In the works related to the DHMIP there is *loop removal* modelling, however, it cannot be used here since our model is different from what is used in [24].

A *loop* occurs when a MN returns to its former point of attachment (FA) without making a normal handover and without an incoming call or packet arriving. It is important to take this into account when determining the parameters of the DHMIP or the LTRACK protocol since they are making decisions on the number of Access Points or Subnetworks the MN has visited. When the MN returns to a former point of attachment before it has changed its MA or executed Location Update it can remove this *loop* and decrease the counter of the visited points of attachment.

Basically when there is a *loop*, the node returns to a former state in the model of LTRACK. (It can be seen that the number of visited FAs should not necessarily be equal to the number of states jumped backwards in the model. This might be a strategic decision in the implementation of LTRACK.) It is also an an important point that when these extra backward links are put into the model, – even with jumping back one state at once, – it can happen that the state of the MN falls back to the P_0 state so the model should be extended carefully.

6.3.1. Basic concepts of loop removal

For the complete discussion of *loop removals*, it is clear that the movement of the MN should be modelled not only from the point of view of the LTRACK protocol but also from a network point of view. Basically a movement model (another Markov chain model) of the MN is needed on the specific network or even the specific network at a certain time of the day. Road topology information can also be used to generate a more exact model for these movement-related issues [10].

This model, – that should be a model based on Markov chains, – is independent of our LTRACK model and uses different states. Combining the two models requires the generation of the Descartes product of the states of the two different Markov chains and the result would be extremely complex.

The graph of the extended Markov chain, can be drawn along two perpendicular axes with respect to the two types of states. One horizontal movement is a movement in the original LTRACK model and the vertical ones belong to *loop removal* modelling.

The probabilities of the transitions between the states are computed in the adequate combination of the probabilities from the two models that can be divided into horizontal rows and vertical columns. (It is a reasonable assumption that the Markov chain model of the *loop removal* has a stationary distribution as well since we can say that there are finite number of access points between the normal handovers of LTRACK and the MN can return to wherever it likes to.) Summing up the state probabilities of the stationary distribution of the extended model in a column will give the state probability of a new, extended LTRACK model. Also the state transition probabilities can be summed up between the columns and used in the new state model.

Since the location of the MN (the vertical “level”) is not important for LTRACK this one dimensional model will be enough to model the algorithm with avoiding the unnecessarily complex computations once the model is determined. With the usage of multi-states (state sets) the extended model is reduced back to a more simple one.

6.4. An example of loop removal strategy: exponential approximation

In this strategy we say that there is little chance for huge loops to be removed at one step. We say that the probability of the number of removed loops decrease exponentially. It is a reasonable assumption as the MN is less likely to return to a cell it has visited much earlier without returning to a cell it has visited later. That rare exception would be when it moves along a huge "circle".

We take a node moving in a GSM-like cellular network where every cell has δ neighboring cells. (The notation (δ) is intentionally the same that was used for network modelling in Section 5.1.1.) If the node moves randomly with $1/\delta$ probability to each neighboring cell the probabilities of the type of loops will be the following: Loop_2: Now the mobil moves from a cell and then returns with no calls or Normal Handover and without other cells visited:

$$\mathrm{Prob}(\mathrm{Loop}_2) = 1/\delta.$$

Loop_3: The mobile node returns with the same conditions but visits two extra cells instead of one:

$$Prob(\mathrm{Loop}_3) == \frac{\delta-1}{\delta} \cdot \frac{2}{\delta} \cdot \frac{1}{\delta};$$

Loop_4: The mobile node returns with the same conditions but in the 4th step (we suppose that $\delta \geqslant 3$ and we have hexagonal-like structure, there are always 3 cells bordering an intersection):

$$P(\mathrm{Loop}_4) = P(\mathrm{Loop}_2|\mathrm{Loop}_2) + P(\overline{\mathrm{Loop}_2}) = \frac{\delta^2}{\delta^4} + \frac{4\delta}{\delta^4}.$$

If we take a look at the values of each *loop removal* probability than it is easy to see that:

$$P(\mathrm{Loop}_i) > (1/\delta)^{i-1}.$$

We will use the following approximation:

$$P(\mathrm{Loop}_i) \approx (1/\delta)^{i-1},$$

from each state backwards.

The modified Markov chain will be the following:

$$\mathbf{Q_E} = \begin{pmatrix} -\lambda+\mu-\mu & \lambda & 0 & \cdots & 0 \\ \mu+\frac{\lambda}{\delta} & -\lambda-\mu & \lambda\frac{\delta-1}{\delta} & \cdots & 0 \\ \mu+\frac{\lambda}{\delta^2} & \frac{\lambda}{\delta} & -\lambda-\mu & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mu+(1+\frac{\lambda}{\delta^H}-\frac{\lambda}{\delta^{H-1}}-\ldots-\frac{\lambda}{\delta}) & \frac{\lambda}{\delta^{H-1}} & \frac{\lambda}{\delta^{H-2}} & \cdots & -(\frac{\lambda}{\delta^H}+\mu) \end{pmatrix}.$$

It is clear that the Markov chain is still finite, irreducible and aperiodic. This implies that the state probabilities can be computed together with the expected point of return which are needed for further examinations.

6.5. *Another example of loop removal Strategy*

In this example we present a similar method for modelling the behavior of the mobile as it was presented in [25]. The basic idea is that the MN is moving between hexagonal cells. If we look at the MN at a certain position the probability of return can be computed using a discrete Markov chain model.

Let us model the network with the same graph used in Section 5 with a little modification. We remove all the edges between cells that are not neighboring each other geographically or vertically. (One can se that our model will have no restrictions and can be applied in systems where vertical handovers are used as well.) Now each node in the graph is a domain handled by one particular Access Point. The neighbors of this node are domains where to the MN can make a handover.

Initially the MN is attached to the one starting node in the graph (it is in state K_0 in the Markov chain). Now it can make a handover to an average of δ neighboring domains. When the MN is in these domains it is in state K_1 in the Markov chain. With this structure the MN can move in a 2D model, to a domain in the same domain group (K_i) or to the next (K_{i+1}) or previous (K_{i-1}) ones. (If the MN could move from a domain in domain group K_j to K_{j+m} where $m > 1$ then we have contradiction since this new cell should be in K_{j+1}.)

The next step is to calculate the number of nodes in each group:

- $r_0 = 1$, obviously;
- $r_1 = \delta$, since δ is the average number of neighboring domains.
- $r_i = \delta r_{i-1} - v2r_{i-1} - (r_{i-1} + r_{i-2}) - r_{i-1}$, since: there are δ neighbors of each cell in K_{i-1}; but each domain in K_{i-1} has 2 neighbors in the same domain group; we assume that without vertical handovers from each neighboring nodes in K_{i-2} we can move to the same nodes in K_{i-1} which means that $r_{i-1} - r_{i-2}$ will have one and $2r_{i-2}$ nodes in K_{i-1} will have two neighbors in K_{i-2}; similarly r_{i-1} nodes will have 2 neighbor in K_i so the MN can move there on 2 edges ⇒ r_{i-1} should be taken.

Factor $v > 1$ is the average number of possible vertical handovers in each K_i. It is clear that for a higher v, $K_i^v < K_i$ so the possibility of staying closer to the starting node is higher so the rate of *loop removals* raises.

The movement probabilities can be derived:

- $P(K_0 \to K_1) = 1$;
- $P(K_1 \to K_0) = 1/\delta$;
- $P(K_i \to K_{i-1}) = \frac{r_i + r_{i-1}}{r_i}/\delta$;
- $P(K_i \to K_i) = v2/\delta$;
- $P(K_i \to K_{i+1}) = \frac{r_i + r_{i+1}}{r_i}/\delta$.

The Markov chain obtained by using these probabilities is slightly different from the Markov chain that was presented in Section 6.4.

6.6. Modelling the existing solutions

According to our modelling system presented above we can examine the existing protocols as well. It is not too hard to see that the model for the Mobile IP protocol will be the following:

$$C_{MIP} = \rho m \cdot c_u + (1 - \rho) m \cdot c_d, \tag{7}$$

since the MN has to make updates using m edges from its point of attachment to the HA and when there is communication request towards the MN, the data or the call is routed directly to it via m steps.

The model for the Hierarchical-like solutions is like this:

$$C_{HMIP} = \rho g \cdot c_u + (1 - \rho) m \cdot c_d, \tag{8}$$

since the location of the node has to be updated always to the nearest common router which is g steps away from the MN.

The model for the Cellular-like approaches is rather difficult to construct using this method since the optimal location areas are network dependent. This is a scope of further works.

7. Optimal LTRACK and comparison

In this section $\hat{H}$, the optimal number of LTRACK "tracking handovers" is computed that should be made before a "normal handover" takes place. We will have ρ as an input value and determine the cost under different $\hat{H}$s. Hierarchical-like solutions will also be compared to our algorithm here. Examinations are made under sightly different conditions in terms of network structure and signalling costs with *Mathematica* [26] to study the behavior of the LTRACK algorithm.

7.1. The cost function(s) of LTRACK

The cost function ($C_{algorithm}$) gives the total signaling cost for the specified *algorithm* per call received or handover made according to the properties of g, m, c_u, c_d and other parameters described earlier. The cost function for the original LTRACK without modelling *loop removal* is:

$$C_{OLDLTRACK}(H) = \rho P_H m c_u + (1-\rho)(m c_d + M[h_r] e_{tr} c_d + (1-P_0) m c_u), \tag{9}$$

where e_{tr} stands for the number of edges along which the packets have to be rerouted if the MN is is not under the given FA. This value was proven to be $2g$ for a tree structure and obviously 1 for the fully linked topology.

We have seen in Section 4 when a "normal handover" occurs in LTRACK, there is a signalling message sent to the HA. It is obvious that a hierarchical mobility layer structure could be used in the way that it is used in every handover of the HMIP protocol. This makes the cost of LTRACK lower than or equal to the cost of HMIP since using $H = 0$ for LTRACK results in the same behavior of the protocols.

The cost function for the examined Hierarchical LTRACK is

$$C_{LTRACK}(H) = \rho P_H g c_u + (1-\rho)(m c_d + M[h_r] e_{tr} c_d + (1-P_0) g c_u). \tag{10}$$

Obviously the optimal handover rate can be computed as

$$\hat{H} = \{f(H) : \min(C_{LTRACK}(H))\}^{-1}. \tag{11}$$

For the basic model the cost function $C_{LTRACK}(H)$ can be treated as a continuous one and can be derived since Eq. (5). This provides a fast and easy solution for computing the optimal value of $\hat{H}$ for LTRACK.

If we extend our model with the effect of *loop removal*, it is clear that the cost function for LTRACK remains the same but values of the state probabilities (P_H,P_0), and the expected point of return ($M[h_r]$) will be different.

7.1.1. The gain of LTRACK

To make the examination clearer LTRACK is mostly compared to the HMIP-like protocols since MIP is obviously much worse in the sense of signalling traffic load. Our *gain* function is the difference of the cost functions:

$$C_{GAIN}(\cdot) = C_{HMIP}(\cdot) - C_{LTRACK}(\hat{H})(\cdot), \tag{12}$$

where $(\cdot)$ can be one of several parameters related to the graph (m, g, δ), mobility rate (ρ) or the cost difference between Location Update and Packet Delivery per edge (c_u, c_d). Function $C(\cdot)$ gives the benefit of LTRACK when any handover or call/packet arrival occurs.

As we do not use absolute costs, we can not calculate the cost of a handover using an algorithm or another. Instead we will compare two different algorithms and find out which performs better.

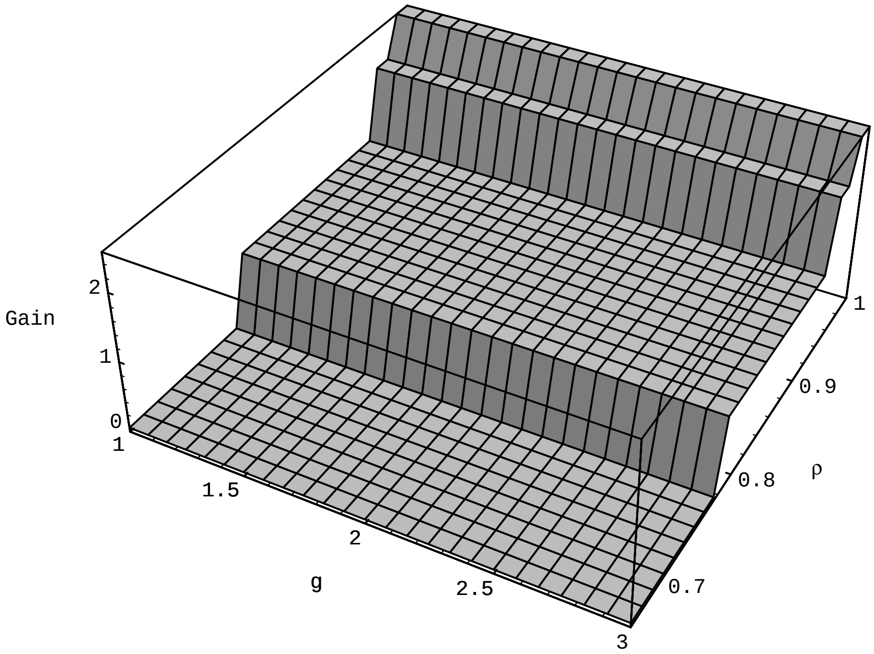


Fig. 4. $\hat{H}$:Optimal value for H.

7.2. The optimal number of tracking handovers ($\hat{H}$) of LTRACK

The optimal "normal handover" rate of LTRACK can be determined by finding the global minimum of function 10. We will use $C_{LTRACK}(h)$ as the continuous version of the cost function. (For the case without *loop removal*, the closed formula can be used with the substitution $(H \in \mathbb{N}_0) \rightarrow (h \in \mathbb{R})$.)

In Fig. 4 one can see the number of "tracking handovers" should be made between normal ones to obtain the best performance. (This is a case without modelling *loop removal*.) It is clear that $\hat{H}$ must be a positive integer.

Figure 5 shows how the Exponential Approximation based *loop removal* affects the optimal value of H: the optimal number of tracking handovers has to be made before a normal one is higher when the parameter ρ is less if the effect of *loop removal* is also considered.

The figures above were calculated on different tree-like network structures. Considering a graph where there is always a link between the old and the new FA of the MN (fully linked case: $e_{tr} = 1$, $g = 1$) the optimum number of "tracking handovers" is higher. In Fig. 6 the four curves from down to top denote: 1. no links and no *loop removal*, 3. no links with *loop removal*, 4. with links but without *loop removal*, 5. with links and with *loop removal*.

Although LTRACK with the optimal "normal handover" rate performs at least as good as the best HMIP algorithm, it is reasonable to ask which network and MN parameters affect our location management method. It will be shown that the performance depends strongly on the difference between location update and packet delivery costs, the value of g and m and the behavior of the MN in terms of ρ. From now the cost of LTRACK is computed with $\hat{H} \in \mathbb{N}_0$ so the cost save is optimal.

The integer value is determined as follows: the $C_{LTRACK}(H)$ function is numerically evaluated for $\forall H \in \mathbb{N}_0$ and $\hat{H}$ with the minimum cost is taken as the result. One can see that only finite Markov

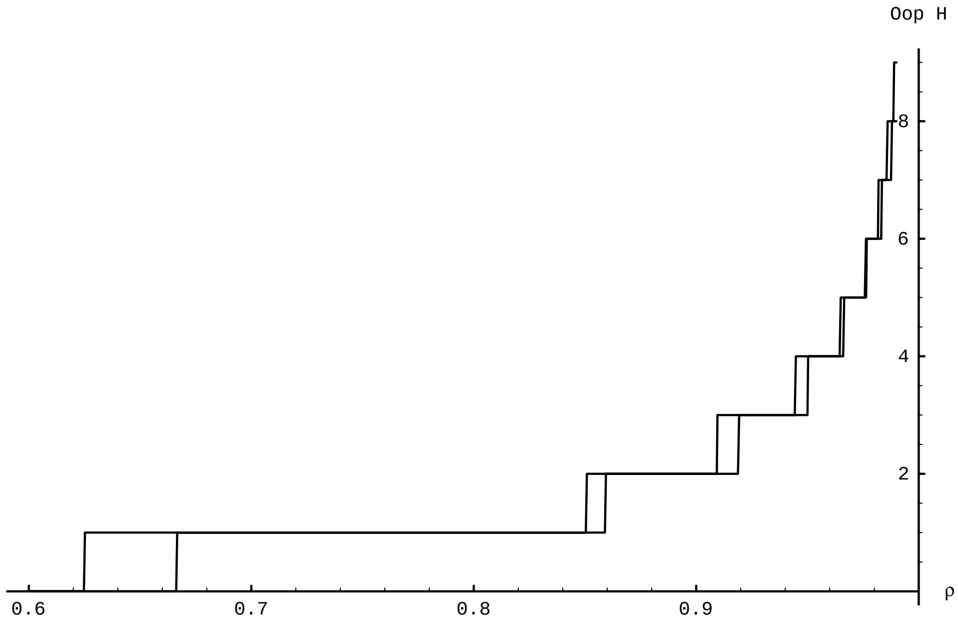


Fig. 5. The effect of *loop removal* on $\hat{H}$.

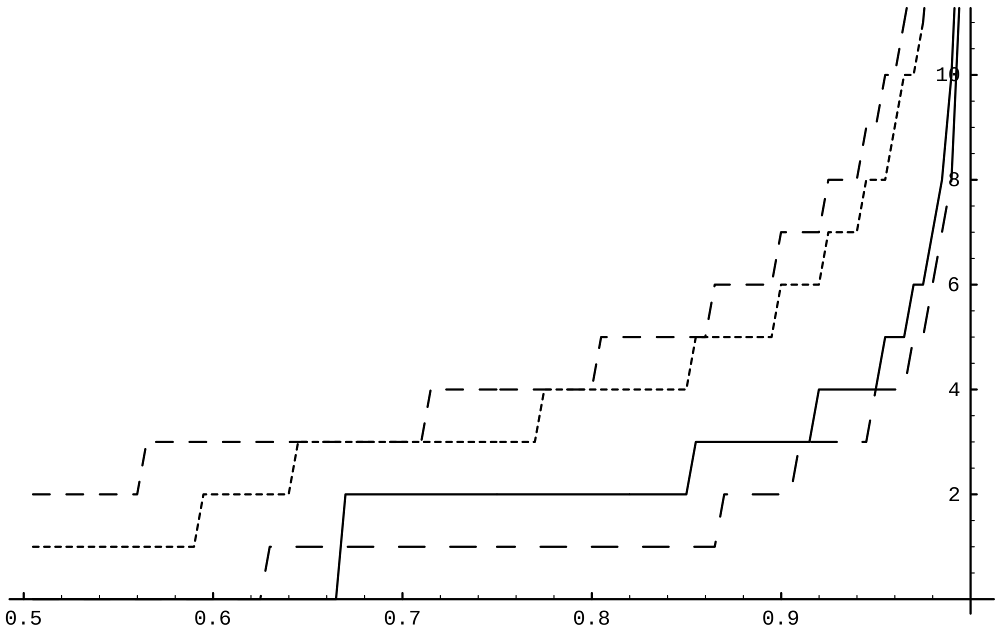


Fig. 6. The effect of links between the FAs on $\hat{H}$.

chains have to be examined since an infinite one would model a scenario when the MN never updates its location to anyone which is obviously unrealistic.

For practical purposes, we evaluate the cost function for a finite number of Hs to obtain the minima.

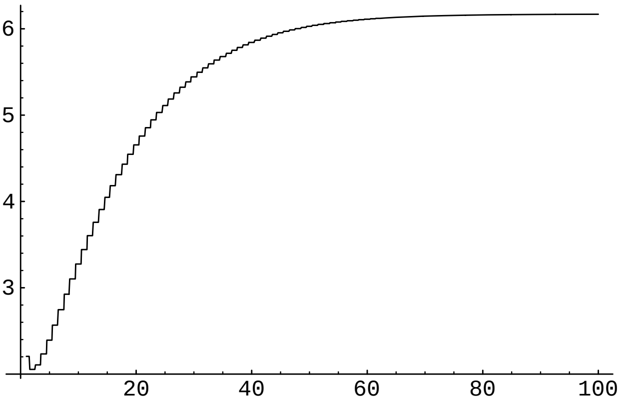


Fig. 7. The effect of links between the FAs on $\hat{H}$.

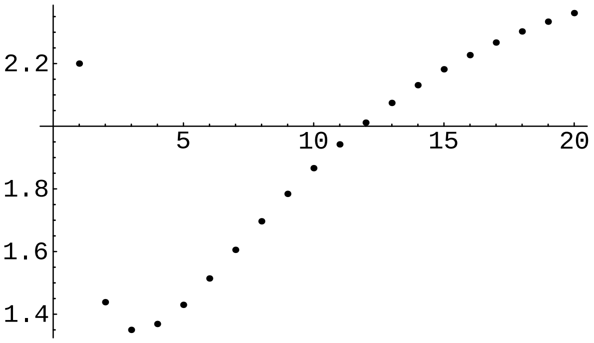


Fig. 8. The effect of links between the FAs on $\hat{H}$.

One can argue that function C_{LTRACK} might have multiple number of local minimum values and the global one might not be in the first H_0. However, one can see on Figs 7 and 8 that this function increases after reaching its first "minima". (The difference between the two functions is that the second one was calculated with the closed formula, neglecting the *loop-removal* and taking a continuous function.)

There is the following theorem for the case without *loop removal* to backup the idea above:

Theorem 1. The cost function for LTRACK is monotony increasing ($C_{LTRACK}(h) < C_{LTRACK}(h+\varepsilon)$) if $h > \frac{1}{1-\rho}$.

Proof. The derivative for the basic model of the LTRACK (the one without *loop-removal*) is the following (using Eq. (10) with Eq. (5)):

$$\begin{aligned}\frac{dC_{LTRACK}(h)}{dh} = {} & -\frac{g\rho^{1+H}e_{tr}c_d(-1+\rho)(-1+\rho^{1+H})^2}{(-1+\rho)(-1+\rho^{1+H})^2} \\ & +\frac{g\rho^{1+H}c_u(-1+\rho)^2\rho + e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2\ln\rho}{(-1+\rho)(-1+\rho^{1+H})^2}\end{aligned} \tag{13}$$

where $\rho < 1$. We say that there is a $H_0 = [\frac{1}{1-\rho}] + 1$ constant from where the equation above is always greater than zero so the function is monotony increasing if $h > H_0$:

$$-\frac{g\rho^{1+H}e_{tr}c_d(-1+\rho)(-1+\rho^{1+H})^2}{(-1+\rho)(-1+\rho^{1+H})^2} \tag{14}$$

$$+\frac{g\rho^{1+H}c_u(-1+\rho)^2\rho + e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2\ln\rho}{(-1+\rho)(-1+\rho^{1+H})^2} > 0 \tag{15}$$

Since $0 < \rho < 1, (\Rightarrow \ln\rho < 0) 0 < g, 0 < c_u, 0 < c_d, 1 \leqslant e_{tr} \leqslant 2$:

$$\begin{aligned}
&e_{tr}c_d(-1+\rho)(-1+\rho^{1+H})^2 \\
&\quad+(c_u(-1+\rho)^2\rho + e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2)\ln\rho > 0 \\
&\quad(c_u(-1+\rho)^2\rho + e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2)\ln\rho < 0 \\
&\quad c_u(-1+\rho)^2\rho + e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2 < 0 \\
&\quad e_{tr}c_d(-1+H(-1+\rho))(-1+\rho^{1+H})^2 < 0 \\
&\quad -1+H(-1+\rho) < 0 \\
&\quad \frac{1}{\rho-1} < H
\end{aligned} \tag{16}$$

Since we used only equivalent transformations between the equations above, the theorem is proved. □

For the *loop-removal* case, we present some figures and it can be seen that the behavior is similar. *Loop removal* modifies the quantitative values of the function but the behavior respect to its monotony is similar. At first, continuous function will be introduced the same way in Eq. (13) and the derived function will be analyzed.

7.3. The importance of the graph model

As it was outlined, an easier discussion of the problem can be made if we assume that g does not depend on m. If we examine our gain function, it can be seen that it became independent of m for the tree topology ($e_{tr} = 2g$):

$$C_{GAIN}(g) = g(\rho c_u - \rho P_H c_u - (1-\rho)(M[h_r]2c_d + (1-P_0)c_u)). \tag{17}$$

It seems that the gain of using LTRACK does not depend on the network model used: If all other parameters are fixed, the larger g is, the bigger gain we have, as long as $0 < C_{GAIN}(g)$. (For this please see Theorem 2.) On the other hand, as it was shown in Section 7.2, $\hat{H}$ depends on g that implies, since P_H and P_0 depends on $\hat{H}$, C_{GAIN} with the optimal H will depend on g in all sense.

Theorem 2. With an optimal H value it can be shown that $0 \leqslant C_{GAIN}(\cdot)$.

Proof. This can be proved indirectly: If $C_{HMIP} - C_{LTRACK}(\hat{H}) < 0$ then $C_{HMIP} < C_{LTRACK}(\hat{H})$. This is impossible since the optimal value for H can be chosen to 0 and then $C_{LTRACK}(0) = C_{HMIP}$. □

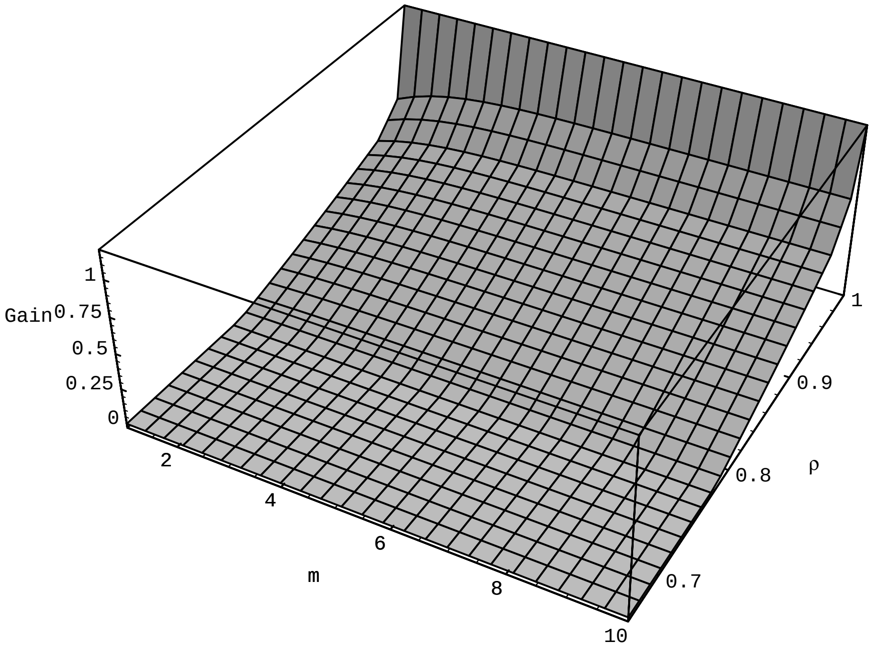


Fig. 9. How ρ and g effects on LTRACK having $m = 5$.

7.4. *Analysis of the results*

One obvious achievement of these calculations is that they provide a numerical method to determine the optimal parameter for LTRACK and also to determine the cost. Since these functions are difficult to solve we use *Mathematica* [26] to calculate the optimal handover rate as well as to analyze the results.

7.4.1. *The parameter ρ*

It is not hard to see that the higher ρ is, the better the performance of LTRACK is (Fig. 9). Also it can be easily proven that if $\rho \leqslant 0.5$ then $\hat{H} = 0$ when there are no direct links between the old and the new FAs of the MN.

Example scenarios where ρ can be high enough (even close to 1):

Highway: The most obvious scenario is a highway especially in a domestic area when there are many access points and the mobile node moves at a relatively high speed. The handover rate (ρ) can reach a really high level in this situation. On the other hand the rate of packets or calls received can also be low.

Vertical handover: The expression *Vertical handover* refers to the handovers when the MN is switching between different wireless access technologies. In these cases the node switches its MA in every realistic scenario since it is moving to another network.

There can be many reasons for this kind of handover. One is that the former access technology is no longer available. Another reason can be that the MN tries to find the most suitable one from the available technologies. For example there can be expensive and inexpensive services and also the available bandwidth and QoS can be important parameters.

The MN can switch from a good service to a poorer one for a short period because the better is temporary unavailable or its quality decreases drastically. These situations imply not just a high handover rate but a higher chance for *loops* that, as we will se, also makes our algorithm perform better.

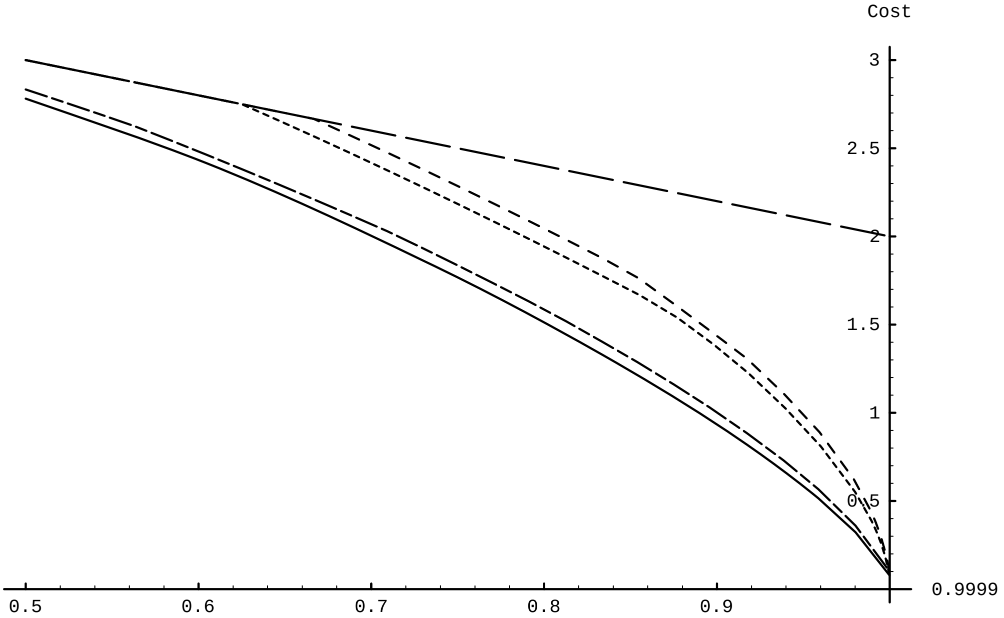


Fig. 10. How ρ and g effects on LTRACK having $m = 5$.

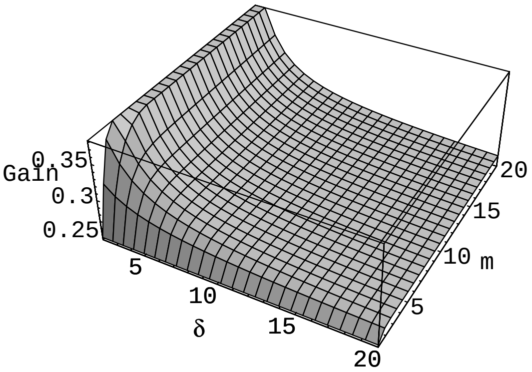


Fig. 11. How δ, m affects on the cost of LTRACK having $\rho = 0.9$.

We can examine the different scenarios for the tree-like and the fully linked backbone network cases with the *loop removal*. On Fig. 10 the curves from top (most expensive) to bottom (less expensive) are: 1. (dotted) the cost of the hierarchical solution, 2. (two-dotted) no links ad no *loop removal*, 3. (dashed) no links with *loop removal*, 4. (one-dotted) with links but without *loop removal*, 5. (solid) with links and with *loop removal*.

7.4.2. The parameter $g(m, \delta)$

We examine the effect of $g(m, \delta)$ in a tree topology network. Figure 11 shows that we gain more for smaller δs when g becomes relatively large compared to m. The conclusion is that if the number of nodes and edges are fixed, it is better to have a deeper network.

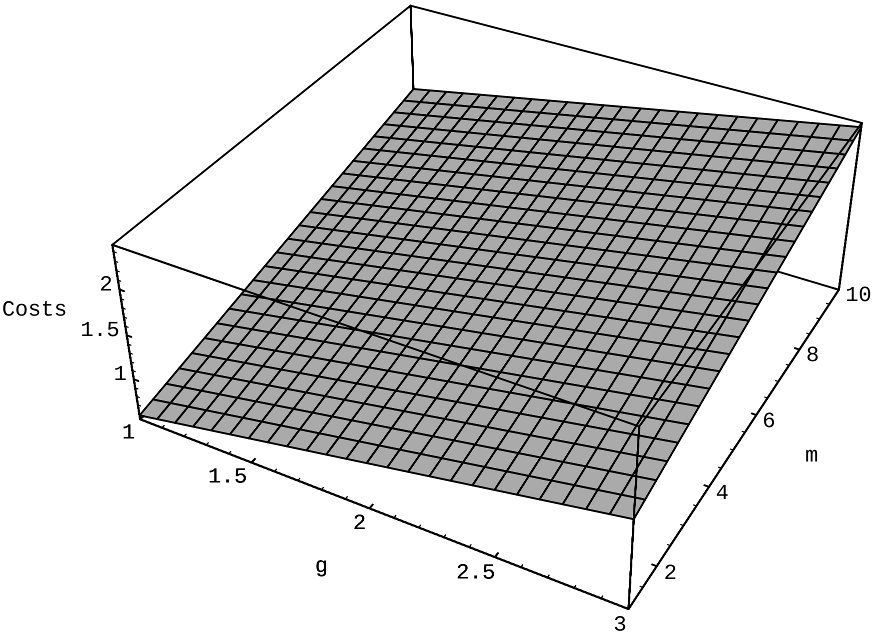


Fig. 12. How g and m effects on LTRACK having $\rho = 0.9$.

7.4.3. Tree topology vs. linked graph

The different types of network models are presented earlier in Section 5. There is the tree topology and the fully linked one. We say that the tree topology examination is appropriate if there is some kind of hierarchy introduced even with mobility layers. However, the fully linked graph approach can also be used since it is likely that the neighboring FAs are linked.

With the adequate choice of values for the g and m parameters, other kind of network topologies can be modelled as well. For example let us consider fully linked networks for each mobility technology. We can assume that all the FAs are interconnected. However if vertical handovers (handover between access technologies) are also considered, it is likely that we have to use g parameter so that $g \geqslant 1$.

Figure 12 shows the cost of LTRACK with independent values of g and m and with a fixed mobility ratio ($\rho = 0.9$ and the worst topology type $e_{tr} = 2g$). The same scenario with the gain of LTRACK is depicted in Fig. 13.

7.4.4. The parameter $\psi = c_u/c_d$

The cost difference between packet delivery (c_d) and location update (c_u) is examined. We introduce $\psi = c_u/c_d$. It can be seen in Fig. 14 that it is better when the delivery cost is lower compared to the cost of the update.

7.4.5. The effect and behavior of loop removal

We have already shown that *loop removal* makes our algorithm cheaper (see Fig. 15 for example). Here, we focus on the effect of the number of neighboring cells used with the Exponential Approximation.

The higher the chance of return to a former point of attachment ($1/\delta$) is, the less the cost will be. Thus the less possible neighboring nodes a MN can make a handover to, the more cost-effective LTRACK is (see Fig. 16).

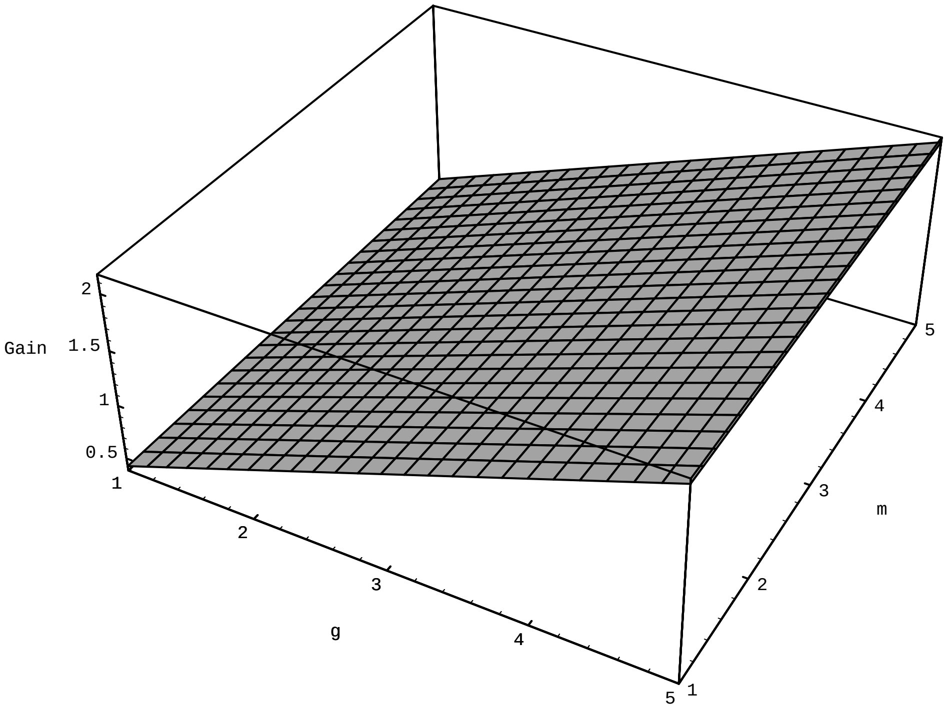


Fig. 13. How g and m effects on the gain of LTRACK contra HMIP having $\rho = 0.9$.

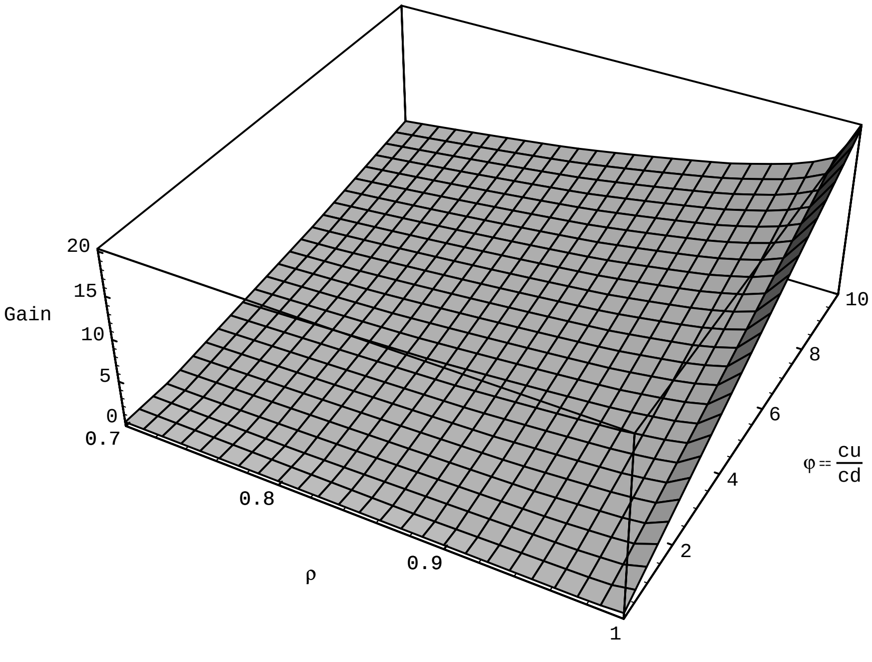


Fig. 14. How ψ and ρ effects on LTRACK having $m = 2, g = 5$.

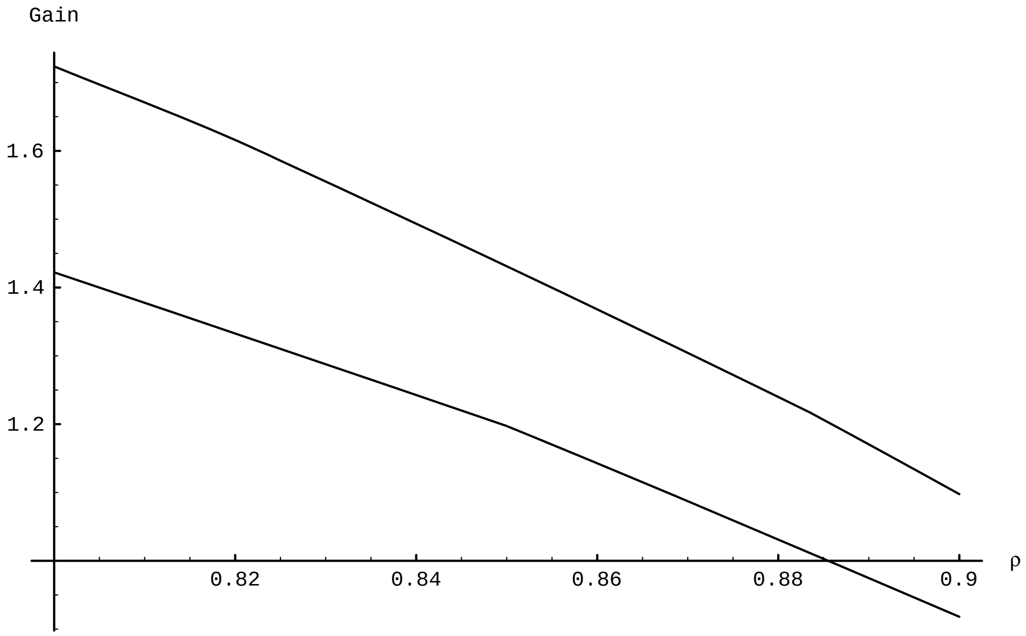


Fig. 15. The gain of *loop removal* compared to LTRACK.

7.5. Summarizing the results

It has been proven that LTRACK outperforms the regional registration-like solutions in many cases and is never worse than them. Applying a sophisticated *loop removal* model has shown the positive effect of *loop removal* on LTRACK. It was shown that LTRACK saves a lot of signalling load on a tree-like network topology and performs even better when there are additional links between FAs and the network is more *fully linked*-like.

These results above provide us a simple qualitative method to decide whether it is worth to implement and run the algorithm on a given network or not. Moreover, the network can also be designed in a way to obtain the best performance from LTRACK.

8. Conclusions and future work

8.1. Achievements in this work

After an introduction to mobility management, some mobility management algorithms were introduced. Amongst them was our very own mobility management algorithm: LTRACK. Then we have presented our own network model and mobility model, compared them to other models, and showed how these can be used to model the various mobility management algorithms.

LTRACK was optimized and compared to other handover methods and turned out to be a very efficient solution in many cases. It was shown that the network topology affects the cost efficiency of our algorithm and that it is a very important factor for other algorithms as well.

An enhanced Markov Model for LTRACK was introduced and simplified back to provide and effective model for the *loop removal* phenomenon. With slight modifications this enhancement can also be used for the modelling of other algorithms.

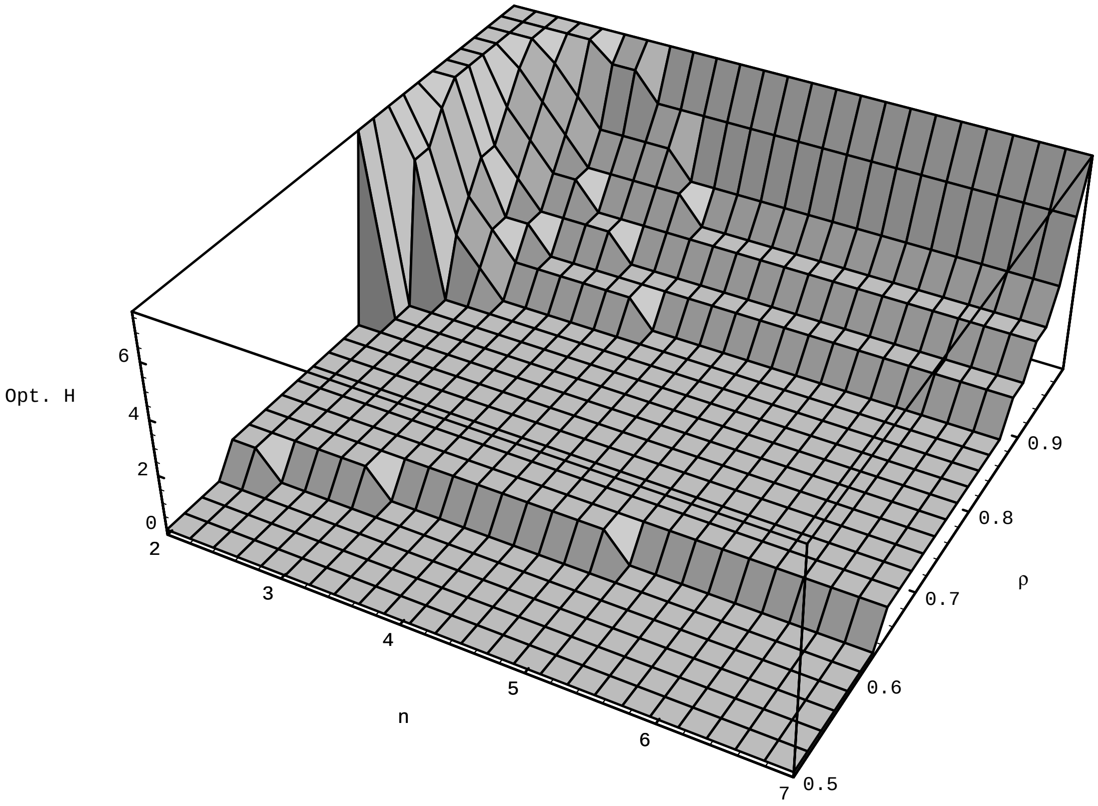


Fig. 16. The optimal H with *loop removal*.

Also it was shown that some properties of the network are very important, but these properties can be incorporated into the model by the use of some parameters (such as g, m, δ) that describe these properties.

8.2. Future work

A mathematical method was introduced to obtain $\hat{H}$, the optimum number of "tracking handovers" that should be made before a normal one takes place. A future goal is to improve cost-efficiency further with more sophisticated decisions between normal and tracking handovers. One can see that the performance of LTRACK depends strongly on the cost saving property of each "tracking handover". Instead of using a counter of "tracking handovers" (h) we will try to give a direct approximation of the expected value of the cost of making tracking handover instead of a normal one. When this expected value is negative, making a tracking handover is profitable. To enable such calculations in real systems, some extra information should be provided to the MN. This method is already under development, a "base station coloring" method will be introduced.

Our model is very general and covers a lot of different aspects, but *processing cost* is not taken into account. Incorporating processing costs into our model is within the scope of our future research. The cost of using the air interface together with the battery usage will also be added to our model. However, these parameters are the most important ones when we want to include the cellular-like solutions into our examination. There are many difficulties to do this especially because of the location area optimization for the cellular-like solutions but we are currently working on it.

References

[1] A.T. Campbell, J. Gomez and A.G. Valkó, An Overview of Cellural IP, *IEEE* (1999), 29–34.

[2] A.T. Campbell, J. Gomez, S. Kim, C. Wan, Z.R. Turányi and A.G. Valkó, *Comparison of IP Micro-Mobility Protocols,* comet.columbia.edu/micromobility, 2004, 29–34.

[3] B. Gloss and C. Hauser, *The IP Micro Mobility Approach*, EUNICE 2000, Enschelde, 2000.

[4] C. Abondo and S. Pierre, *Dynamic Location And Forwarding Pointers For Mobility Management,* Mobile Information Systems, IOS Press, 2005, 3–24.

[5] C. Castelluccia, A Hierarchical Mobile IP Proposal, Inria Technical Report, 1998.

[6] C.E. Perkins, Mobile IP, *IEEE Communications Magazine* (1997).

[7] C. Perkins (Editor), IP Mobility Support, IETF, Request for comments, 1996.

[8] D. Johnsson, C. Perkins and J. Arkko, *Mobility Support in IPv6,* IETF, Request for comments 3775, 2004.

[9] E. Gustafsson, A. Johnsson and C. Perkins, *Mobile IPv4 Regional Registration,* draft-ietf-mobileip-reg-tunnel-06.txt, work in progress, Mar, 2002.

[10] G. Sivaradj, R. Nakkerman and P. Dananjayan, *A Prediction Based Flexible Channel Assignment in Wireless Networks Using Road Topology Information,* MOMM2005, Third International Conference on Advances in Mobile Multimedia, 2005, 107–118.

[11] J. Redi and P. Bahl, *Mobile IP: A Solution for Transparent, Seamless Mobile Computer Communications,* Fuji-Kezai's Reports on Upcoming Trends in Mobile Computing and Communications, 1998, 1–10.

[12] J. Xie and F. Akyildiz, *A Distributed Dynamic Regional Location Management Scheme for Mobile IP,* Proc. IEEE INFOCOM, 2002, 1069–1078.

[13] J. Xie and F. Akyildiz, A Novel Distributed Location Management Scheme for Minimizing Signalling Costs in Mobile IP, *IEEE Transactions On Mobile Computing* (2002).

[14] K.D. Wong and W.W. Lee, *Likelihood of Lost Binding Updates when Mobile Nodes Move Simultaneously,* MOMM2005, Third International Conference on Advances in Mobile Multimedia, 2005, 9–19.

[15] M. Song, S. Kang and K. Park, *On the design of energy-efficient location tracking mechanism in location-aware computing,* Mobile Information Systems, IOS Press, 2005, 109–127.

[16] N.A. Fikouras and C. Gorg, *A complete comparison of algorithms for Mobile IP hand-offs with complex movement patterns for Internet audio,* Proceedings of the Fourth International Symposium on Wireless Multimedia Communications (WPMC), 2001, 9.

[17] P.G. Escalle, V.C. Giner and J.M. Oltra, Reducing Location Update and Paging Costs in a PCS Network, *IEEE Transactions on Wireless Communications* **2002**, 200–209.

[18] R. Jain and Y.-B. Lin, *An auxilary user location strategy employing forwarding pointers to reduce network impacts for PCS,* Wireless Networks I., 1995, 195–210.

[19] R. Ramjee, T.La Porta, S. Thuel, K. Varadhan and L. Salgarelli, A Hierarchical Mobile IP Proposal, Inria Technical Report, 1998.

[20] S. Das, A. Misra, P. Agraval and S.K. Das, TeleMIP: Telecommunications-Enhanced Mobile IP Architecture for Fast Intradomain Mobility, *IEEE Personal Communications* (2000), 50–58.

[21] S. Imre and M. Szalay, LTRACK – A Novel Location Management Method, EUNICE 2003, Budapest – Balatonfured, Hungary, 2003.

[22] S. Imre and M. Szalay, *Parameter Considerations of LTRACK,* EUNICE 2004, Tampere, Finland, 2004.

[23] W. Fritsche and F. Heissenhuber, *Mobile IPv6 – Mobility Support for the Next Generation Internet,* IABG mbH, 2000.

[24] W. Ma and Y. Fang, Dynamic Hierarchical Mobility Management Strategy for Mobile IP Networks, *IEEE Journal of Selected Areas In Communications* (2004).

[25] W. Wang and F. Akyildiz, A Dynamic Location Management Scheme for Next Generation Multitier PCS Systems, *IEEE Transactions On Wireless Communications* (2002), 178–189.

[26] Wolfram Research Inc., Mathematica, http://www.wolfram.com/.

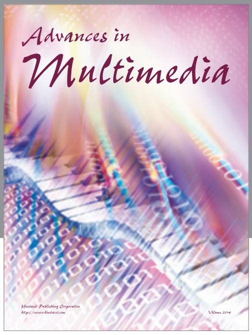
[27] Y. Fang and Y. Lin, Portable Movement Modelling for PCS Networks, *IEEE Transactions on Vehicular Technology* (2000), 1356–1362.

Benedek Kovács received the MSc degree in Computer Sciences from the Budapest University of Technology and Economics in 2006. His research areas include mobility management; parameter estimation.

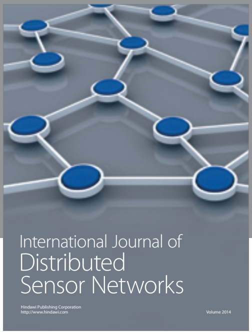
Máté Szalay received the MSc degree in Computer Sciences from the Budapest University of Technology and Economics in 2001. He is a PhD student at the same university. His research areas include mobility management, reliability, security.

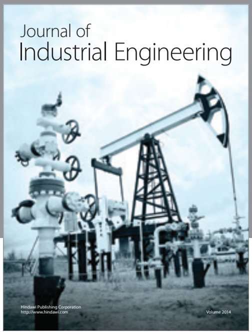
Sándor Imre received the MSc degree in Telecommunication Engineering from the Budapest University of Technology and Economics in 1993. In 1999 he received the PhD degree in Electrical Engineering. He is an Associate Professor at the Budapest

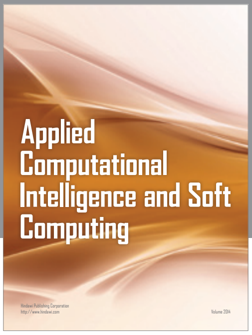
Universiy of Technology and Economics and the R&D Director of the Mobile Innovation Center, Hungary. His research areas include IP mobility, routing, reliability; Wireless LANs; Software Defined Radio; Quantum Computing.

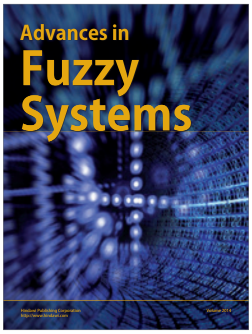

Advances in
Multimedia

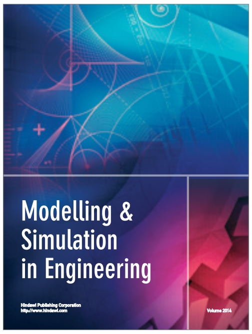

The Scientific
World Journal

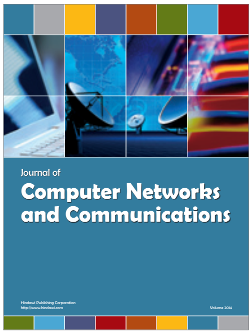

International Journal of
Distributed
Sensor Networks

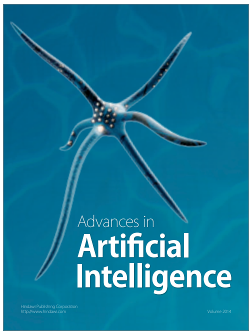

Journal of
Industrial Engineering

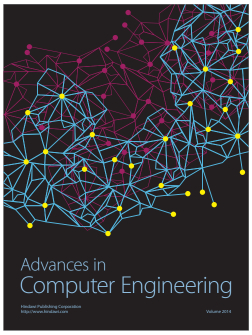

Applied
Computational
Intelligence and Soft
Computing


Advances in
Fuzzy
Systems

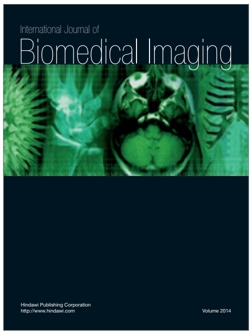

Modelling &
Simulation
in Engineering

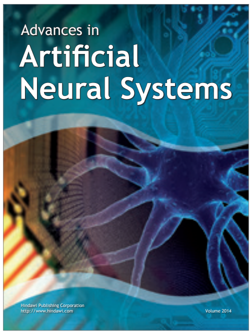

Journal of
Computer Networks
and Communications

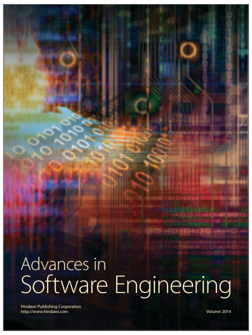

Advances in
Artificial
Intelligence


Advances in
Computer Engineering

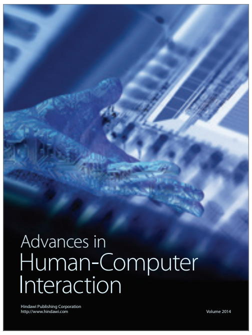

International Journal of
Computer Games
Technology

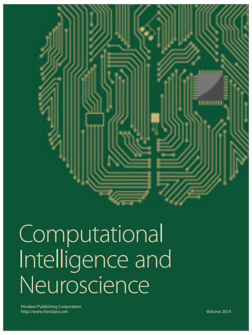

International Journal of
Biomedical Imaging


Advances in
Artificial
Neural Systems


Advances in
Software Engineering


Journal of
Robotics


Advances in
Human-Computer
Interaction


Computational
Intelligence and
Neuroscience


International Journal of
Reconfigurable
Computing


Journal of
Electrical and Computer
Engineering