Novel Mobility Management Algorithms

Szalay Máté

Scientific supervisor

Dr. Imre Sándor

Budapest, 2007
Abstract

My thesis presents my new scientific results: a classification system for location management algorithms and two novel location management algorithms.

After the introduction, in Chapter 2 the related work is presented very briefly. Mobility and location management are explained in general, and some mathematical modelling techniques, such as Markov chains are introduced. A brief survey is given on state-of-the-art location management methods which are related to or can be compared to my own algorithms.

In Chapter 3 I introduce my own classification system of mobility management algorithms. This is a general classification method that can be used to classify existing and new algorithms and which has led to the finding of Hierarchical Paging.

LTRACK, my first mobility management algorithm is introduced in Chapter 4. First the network architecture and mechanisms are described, then qualitative analysis, simulations and quantitative analysis are presented. My algorithm is compared to other location management algorithms, and it is examined how changes in various parameters affect the performance of LTRACK.

My second mobility management algorithm – called Hierarchical Paging – is presented in Chapter 5. After qualitative considerations an analytical model is presented for it, which is compared to my simulations of Hierarchical Paging. The analytical results and simulation results fit very well. Simulations are also used to compare Hierarchical Paging to other location management algorithms, and the parameter ranges are determined where Hierarchical Paging is most efficient.

Finally, in Chapter 6 the conclusions are drawn, possible applications of my result and possible future directions in this research area are presented.

1 Background

Over the past few years there has been extreme growth in wireless communications. While mobile telephony remains one of the most widespread mobile application, mobile web browsing, sending and receiving email or even mobile banking are becoming more and more widely used. A mobile provider that can use its infrastructure in a more efficient way has a competitive advantage over the others.

Any mobility protocol has to solve two separate problems: location management (sometimes called reachability) and session continuity (sometimes referred to as handover management). Location management means keeping track of the positions of the mobile nodes in the mobile network, session continuity means to make it possible for the mobile node to continue its sessions (e.g. phone calls) when the mobile node moves to another cell and changes its
service access point. My dissertation concentrates on the problem of location management. Location management has to address the following questions[3]:

- When should the mobile terminal update its location to the network?
- When a call arrives, how should the exact location of the called mobile equipment be determined?
- How should user location information be stored and disseminated throughout the network?

These problems are usually solved in two stages: *location registration (or update or tracking) and call delivery (or searching)*[1].

There are several solutions to this problem [3, 5, 9, 10]. These solution usually use a hierarchical approach.

Several studies have addressed paging, various paging methods are presented, usually considering cost minimizations. Almost all of these papers focus on the paging algorithm itself, but the paging is always just at a single (usually the bottom) level of the network (*flat paging*, see Chapter 3). Ramjee et al. examine and compare three different paging architectures and protocols [6], all flat. Hajek, Mitzel and Yang [2] show algorithms to optimize registration and paging together. They examine serial and parallel paging, but none of their solutions introduces a hierarchy in the paging itself as my solution (*Hierarchical Paging* does).

Usually, paging is used at the bottom level of a hierarchical mobility management solution, as in case of a GSM network [4, 8].

In their paper[11], Woo et al. optimize location management for a special types of networks, namely *Two-Way Messaging* networks.

The paper of Akyildiz et al.[1] is a good general survey on various existing and proposed mobility management algorithms.

None of the above solutions mention *Location Tracking* or *Hierarchical Paging* as an alternative solution. An approach similar to LTRACK is used by RoamIP [10] but only for session continuity and not location management.

## 2 Terminology

In a wireless network service access points are usually called *base stations*. The node that is moving around in the mobile network is called *mobile node* or *mobile equipment*. *Mobile nodes* are connected to the network via *base stations*, each of the base stations covers one *cell*, and there are wireless links between the base stations and the mobile equipments. Base
stations are interconnected with routers to form a network; this network usually uses fixed
and sometimes microwave links.

As the mobile node is moving around, it changes its point of connection to the network
from time to time. The event, when the mobile equipment moves to a new service access
point is called handover or handoff.

In a mobile network the terms location and position usually do not mean geographical
position, but they refer the location of the mobile node within the network, the service access
point (base station) it is connected to at a specific moment of time.

3 An Engineering Problem

In a mobile network both tracking and searching have a cost. Here, cost doesn’t necessarily
mean a cost paid in money, it refers to a general “price” that is to be payed in money, time,
algorithm complexity, wasting of resources, shorter equipment battery lifetime, lower user
satisfaction, etc.

Consider one scalar parameter of a mobility management solution. If an increase in the
parameter value increases some cost factors and does not change others on the whole range,
then there is no conflict: the lower parameter value, the better. If changing the parameter
might increase some cost factors and decrease others then there is a “conflict” which might
lead to an optimization problem.

In this sense, there is a general “conflict” between tracking and searching. The scalar
parameter is the frequency (and perhaps accuracy) of location updates. If more update
messages are sent, the network will have more accurate position information of the mobile
node most of the time. On one hand, update messages have a cost, but on the other hand, if
we have more accurate position information, the (expected) cost of the searching procedure
will be smaller.

The price of sending updates more frequently is paid in higher signallng load on the
network, higher communication channel usage, shorter battery life for the mobile node. The
price of searching is paid in more signalling, more communication channel usage, longer search
procedures (lower user satisfaction), and perhaps higher algorithm complexity.

Thus, updating more frequently makes the total cost of updates higher, but the total cost
of searching lower; and updating less frequently makes the total cost of updates lower, but
the total cost of searching higher.

Of course this is just an oversimplified model of the problem of location management,
but it is easy to see that depending on how various costs (time, money, signallng load) are
compared to each other, this can lead to optimization problems. There can be optimum
update “frequencies” which lead to the lowest total (update and searching together) cost
possible.

My thesis addresses this problem.

4 Research Goals

4.1 Universal Location Management Model

The first research goal was to provide a universal model or classification system for location management. The model should be general enough that it covers most (if not all) of today’s location management algorithm, and should enable us to classify and compare various algorithms.

4.2 Novel Mobility Management Algorithms

To make mobility management more efficient, we can either tune the parameters of an existing algorithm to suit the environment better or completely new algorithms can be designed. The most important research goal was to find novel mobility management mechanisms and build algorithms that use them. The efficiency of an algorithm depends on several factors, so we usually can’t say that an algorithm is better than another one in every scenario and at every parameter setting, this was not my goal. My goal was to find new location management algorithms that outperform most of today’s solutions at reasonable parameter settings of reasonable scenarios.

5 Research Methodology

To find out if my new location management methods are indeed more efficient than others, I had three approaches: analytical considerations, simulations and measurements. When developing novel algorithms, in the early phase usually simulations and analytical methods are used; those are the approaches that I have used in my work. To use the third approach – measurements – my algorithms would have had to be implemented, which is usually a time (and money) consuming process, which should normally follow the previous two.

To use an analytical method, first a mathematical model had to be constructed. I have defined the mathematical model for both of my new algorithms. Both of the models make use of Markov-chains extensively. For the mobile network I used a simple graph model, the mobility model was taken from literature with some minor extensions.

I have also created several simulations. I haven’t used any specific simulation tools, but built my own simulation framework in C++. Some of the simulations are created using
MATLAB\[7\], others are built upon my own simulation framework.

Location Management Classification
I have built a general, structured, classification system for location management methods. \[C14, C15\]

My classification system is presented in Chapter 3 of my dissertation. It is not a thesis itself, but contains new elements, and helped to find Hierarchical Paging.

My system is general and technology independent in the sense that it covers most of the current technologies and solutions, and hopefully future ones as well.

First, single-level solutions are classified, then multi-level solutions are considered.

By using my system, the different mobility solutions can be classified. After classifying the mobility solutions of today, we may find that some classes are empty. A class may be empty for two reasons:

- It is impossible or senseless to implement a mobility algorithm that belongs to that class.
  
  For example, an algorithm belonging to another (specific) class obviously outperforms it in every sense.

- No mobility algorithm has been found yet that belongs to that class.

The latter case may be exciting; this is the case where our general and solution-independent mobility management classification system has actually enabled us to find something new that might be rather specific. Actually, this is the way how Hierarchical Paging was found.

6 New Results

1. Set of Theses: LTRACK

LTRACK is the name of the first of my two novel location management algorithms. LTRACK is presented in Chapter 4 of my dissertation.

Thesis 1.1 I have introduced a new location management algorithm called LTRACK. LTRACK is the mosaic for “Location Tracking”. The idea is that the exact location of the mobile node is not stored in a central database, but the information is shared among the network nodes that the mobile node had visited recently. The searching procedure “runs” through the chain of these nodes. \[J1, J3, C9\]

An LTRACK network is built up from LTRACK nodes, and a mobile node is connected to one of the LTRACK nodes in the network. There is a special node in the network: the home LTRACK register (HLR), every mobile node has an entry in the HLR.
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 HLR. It sends the address of the new LTRACK node to the HLR. In case of a tracking handover the mobile sends the address of the new LTRACK node to the old LTRACK node. Figure 1 depicts the two different handover types of LTRACK.

For each of the mobile nodes, the HLR stores the last address where it received location update message from. It is a "next-hop" towards the node. The mobile node is either connected to that LTRACK node, or that LTRACK node knows a "next-hop" LTRACK node towards the mobile.

Once an incoming call arrives, there is a series of LTRACK nodes pointing from the HLR to the mobile node, see Figure 2.

Sections 4.1. to 4.4. describe the various mechanisms of LTRACK in detail.

**Thesis 1.2** I have created simulations of LTRACK. These simulations show the efficiency of my algorithm and has provided a way to examine how changes in various parameters affect the performance of my algorithm. [C9, C11]

The simulations were written in MATLAB. First, LTRACK is compared to MobileIP and Hierarchical MobileIP and is proven to be more efficient then them, see Figure 3. Then, it was examined how LTRACK can be fine-tuned for optimal performance.

Section 4.5 of my dissertation presents my simulation results.
Figure 2: LTRACK locating the mobile node

Figure 3: Signalling requirements of various protocols

**Thesis 1.3** I have provided a sophisticated analytical model for LTRACK. This model enables the determination of optimal parameter values of LTRACK, and lets us compute the cost gain of LTRACK over a Hierarchical MobileIP-like solution. [J1, C13, C11]

First, the network model is introduced, and various network topologies are considered. A Markov-chain model is used to model the two kinds of LTRACK handover mechanisms, and to find optimum values for LTRACK parameters. Then, a special LTRACK-related phenomenon called loop removal is examined using the analytical model.

Then, a similar model is presented for MobileIP and Hierarchical MobileIP, and their efficiency is compared to LTRACK. Using my techniques, the most efficient algorithm can be selected for a specific scenario.
The analytical considerations of LTRACK are presented in Section 4.6 of my dissertation.

2. Set of Theses: Hierarchical Paging

Hierarchical Paging is the name of the second of my two novel location management algorithms. It is presented in Chapter 5 of my dissertation.

**Thesis 2.1** *I have introduced a new location management algorithm called Hierarchical Paging. The idea is to use the well-known paging at not just the bottom level of a hierarchical solution. Hierarchical Paging can either be considered as using paging at two (or more) neighboring levels, or using a hierarchical version of paging at a single level. [C14, C15]*

The network where Hierarchical Paging is used is divided into subnetworks, and there is an network node in each subnetwork that is responsible for local mobility management. This node is going to be referred to as the root node. If a mobile is staying in a subnetwork, then the root node of the subnetwork either knows the exact location of the mobile node within the subnetwork, or it can determine it by using a (single-layer) paging algorithm.

Thus, while the mobile node is moving around within a subnetwork, no location updates are sent. There is no handover at all in the top level, so when the mobile node has to be found the top level knows which subnetwork the mobile is currently in, and within that subnetwork paging can be used.

When the mobile node moves to a new subnetwork (or paging area), the root nodes of both the old and the new subnetworks have to be notified, but inter-subnetwork mobility can still be handled locally (no network-wide central node had to be notified). So the old root node learns that the mobile node is not in its subnetwork any more, and the new root node learns that the mobile node is in the subnetwork it is responsible for.

When an incoming call or packet arrives destined to the mobile node, the old root node is contacted first, because top level mobility management still “thinks” that the mobile node is in the old subnetwork. But the old root node knows that the mobile node is not there, so a top level paging takes place. This top level paging means that the gateway of the network (or the top-level root node) “pages” all the (bottom level) root nodes of the subnetworks, and the root node of the new subnetwork is going to give a positive reply. Then, possibly another (bottom level) paging round is needed to determine the exact position of the mobile.

Sections 5.1. and 5.2. of my thesis describe the various mechanisms of Hierarchical Paging in detail.

**Thesis 2.2** *I have provided a sophisticated analytical model for Hierarchical Paging. This model considers whether the exact position is known at top and bottom levels, and enables the*
determination of optimal parameter values (such as subnetwork size). Further, it proves the efficiency of Hierarchical Paging over single layer (flat) paging in some scenarios. [C15]

First, the network model is introduced. The mobility-model is independent of the network structure and the algorithm used, it is not my own mobility model, but one that is widely used in literature. This model was extended to cover inter- and intra-subnetwork handovers which are important factors of Hierarchical Paging.

Then, a Markov-chain model is introduced which handles the different states of the mobility management algorithms. A two-level Hierarchical Paging solution is modelled.

Costs are assigned to the different transitions of the Markov-chain, the stable state and the average cost of an event is determined. This method enables us to examine how changes in various parameters affect the performance (cost) of Hierarchical Paging.

Then the same Markov-chain model is constructed for the single-level (flat) paging algorithm. Using this method, the two algorithms can be compared to each other at specific environment parameters.

The analytical considerations of Hierarchical Paging are presented in Sections 5.3 and 5.4 of my dissertation.

**Thesis 2.3** I have built simulations of Hierarchical Paging. First, some simple simulations were carried out to validate the results that my analytical model provided for my algorithm. Then, Hierarchical Paging was compared to two other location management algorithms, and the effect of changes in parameters was examined. [C14]

![Figure 4: Comparison of calculated and simulated results of Hierarchical Paging](image)

The simulations were written in C++. I have written a flexible simulation framework, which can be used to carry out simulations of various paging algorithms and can also be easily extended to handle different scenarios.
The simulation results fit the analytical results very well, see Figure 4.

After the validation, Hierarchical Paging was compared to its two "competitors": flat paging and the classical mobility management solution: centralized top-level combined with paging at the bottom level.

Figure 5 shows the simulation results. The subnetwork size (area size) is on the x axis running from 1 to 100 from left to right, mobility ratio is on the y axis running from 0 to 0.99 from top to bottom. Black pixels indicate where Hierarchical Paging was the most efficient.

Note that Hierarchical Paging is more efficient than the classical, paging bottom plus centralized top level solution at high mobility ratio values (which is usually the case in real-life scenarios) and reasonable subnetwork sizes. Using very small or very large subnetwork is highly inefficient no matter what kind of paging algorithm is used. Thus, my algorithm is efficient in the most reasonable parameter-ranges.

Sections 5.5 and 5.6 of my dissertation presents my simulation results.

7 Application of the Results

The mobility management classification system introduced in Chapter 3 could be used to find classes of mobility management algorithms that are empty and thus to find new mobility management algorithms. Note, that Hierarchical Paging was actually found this way.

My two location management algorithms, namely Hierarchical Paging (Chapter 5) and LTRACK (Chapter 6) can be used to handle mobility in networks more efficiently than existing solutions. None of my two algorithms is the silver bullet, but the if the ideas are incorporated in complex solutions, they can improve efficiency, thus reduce costs.
The simulation framework that was used to compare Hierarchical Paging to other location management algorithms is so general and flexible, that it can be used for further comparative simulations and for parameter optimizations of novel location management methods.

References


Publications

Foreign-language Journal Papers Published Abroad


Foreign-language Journal Papers Published in Hungary


Journal Papers in Hungarian Language Published in Hungary


Article in Edited Book


Foreign-language Paper Published in International Conference Proceedings


