COMPLEX MOBILITY MANAGEMENT ALGORITHMS
AND APPLICATIONS

Péter Fülöp
Summary of Ph.D. Dissertation

Supervised by
Dr. Sándor Imre

Budapest, Hungary
2015.
1 Introduction

Information mobility has become one of the most important achievements of the modern world. The number of subscribers and also the network traffic is increasing in the mobile cellular networks. The popularity of the mobile communication is a result of the combination of different factors. Smartphones with more network interfaces keep spreading, while on the other hand the network operators implement technological improvements and standards (HSXPA-High Speed Packet Data, LTE (Long Term Evolution) to be able to cope with increased data transfer needs and also introduce sophisticated services in the market competition. As a result the operators are facing even more complex operational and management tasks:

1. The cell sizes keep getting smaller and change dynamically; the reuse of the available radio channels is increasing also. In this changing environment efficient and adaptive user localization and call transfer is needed to solve the mobility management tasks.

2. Thanks to the new technology improvements the bandwidth increases continuously, however some multimedia applications remain sensitive to sudden degradation of the QoS (Quality of Service). To avoid such problems the appropriate, dynamic dimensioning of the resources and application of call admission control CAC is inevitable.

3. It became part of our daily life to do bank transfers, handle purchases with our mobile devices. Although there are several available protocols that guarantee our safety in the wireless environment, it is still a challenging task to create a completely safe mobile environment due to the open radio interface and the vulnerability of the mobile devices. Lately the creation of user profiles and investigation of user behavior became an important research area. This is quite a complex task, however could significantly increase the safety of the mobile networks.

4. Nowadays there are multiple access networks available at the same time in one geographical location. Besides there can be overlapping between the cells with the concrete access technology. Hierarchical cells and the usage of “umbrella cell” are good examples, when covering more smaller cells with a greater one. The reason behind is serving the fast users and to avoid faults resulting from frequent call transfers. The movement models support the process of possible vertical call transfers at these locations.

These were just a couple examples from the numerous challenging task in the mobile networks and to be able to cope the operators need appropriate solutions and highly qualified, experienced engineers. A prospective, alternative possibility to introduce such procedures and mechanisms that enables the systems to reconfigure, heal or install themselves according to the changing environment. However the initial research area is far from being exhausted it can be seen that well-structured mobility management algorithms can support for example the call admission control, reconfiguration of the cell coverage, detection of the fault of the radio interface or a change in the localization strategy.

The topic of my dissertation is the modeling of the user movement behavior and the investigation of complex location management strategies, which can efficiently aid the above mentioned self-configuration processes.
2 Terminology

Upon setting up a wireless network service access points are located over the targeted geographical locations. These service access points are called base stations. The base stations’ radio interfaces cover a specific area, which is called a cell. The moving user end devices, called cell phones, can communicate with the network – and with each other - through the base stations. The mobile devices are usually moving and therefore change their access points. If during the movement they have an active connection/call the process is called a handover. The previously described functionality has to be managed by the mobility management service; one of this service’s main functions is the location management. The location management means the recording/registering the mobile’s location in the network with a predefined precision/punctuality. This is necessary so that the network is able to route any incoming call or data traffic towards the mobile device.

In my dissertation, during the modeling the mobility is treated generally, with no respect to the exact, applied technology. Accordingly I use the following network entities:

- Mobile Node (MN): moving node, which is communicating with another mobile or fixed base station,
- Mobility Access Point (MAP): the only type of fixed entity in the network to which a mobile node can connect and communicate through. (It does not necessary match an access point in the real network, it can mean/be a network or a set of access points if a mobile can communicate/can be in connection with/ more base stations),
- Mobile Agent (MA): fixed node, runs the mobility management algorithm, but no mobile can connect to it. A hierarchy node is such for example,
- Central or Home Agent (HA): a special, central MA, which always has an exact, or rough/approximate information on the whereabouts/location of an MN,
- Other node (N): such node in the network, that has no function related to mobility management. Fix communication partner, routers and other network elements belong here,
- Core Network (CN): the communication channel (wired or wireless) connecting the entities, which is represented by a graph. The junctions of the graphs represent the above mentioned network entities, the edges of the graphs represent the connections between the entities.
3 Research Objectives

In cellular mobile network the number of users and the amount of transferred data increased significantly. Due to this fact the network operators are facing heavy problems. My research objective with user movement modeling and mobility management strategy analyzing is to support designing and maintaining the next generation mobility networks and to introduce new approach of adaptive, location management algorithm.

My goals are:

- to investigate and compare location management, a part of the mobility management, strategies with respect to their need of signaling and processing resources on the backbone network, load/utilization of the air interface and the cost of the security.

- to classify and investigate Markov movement models applied in cellular mobile networks. Based on my research to create an optimal model and develop general Markov Movement Model Creator System (MMCS), which generates an optimal (in the number of states and the prediction accuracy) Markov model using the network and mobility input parameters.

- to create a self-organizing location management environment, where every mobile node manages their location update strategies for itself instead of the network (Client-driven Mobility Frame System – CMFS). The network only provides basic services for mobile entities: connectivity and administration.

Accomplishing my research goals we get one step closer to a self-organizing, intelligent mobile network, where the mobile node always uses the lowest-cost mobility management strategy and can be tracked/followed by an optimal movement model.

4 Research Methodology

In order to prove the efficiency of the management strategies and movement models I used two classical approaches: analytical considerations and simulations.

First the mathematical model had to be constructed for the analytical consideration. I used graphs to model the network and Markov chains for position management solutions and movement models. In the course of comparing the location management strategies I applied a mobility model created from an existing model in the literature according to my research goals. The extension of this model is used during movement model classification.

I prepared a Mathematica [8] application to confirm the analytical considerations and developed a simulation environment. The simulations were written in the open source, widely known, discrete time OMNet++ [cmfs27] using C++ language. Using these the results are comparable to the approaches in the literature and the effect of different network configurations can be measured.
5 New Results

5.1 Enhanced framework for analyzing mobility management strategies

The function of mobility management is to secure that the mobile device is continuously able to access the network services, receive and initiate calls or data traffic during its movement. Location management is a sub component of mobility management with the aim to be able to determine the location of the mobile device with certain accuracy even when the mobile device is moving. There are a number of position management solutions in the literature for the different mobile network technologies. Generally all of them divide position management into two phases: position update and call delivery. Independent of the solution their usage generates certain cost in the network. This cost could include the amount of traffic transferred during signaling, resource and processing cost occurred in the network nodes, costs of using the radio interface or the extra steps needed to setup a secure connection.

If the mobile updates its current position more often the network can have more accurate location information hence the call delivery is simpler and faster. In this case the position update cost is higher, but the cost of call delivery is lower. With less frequent position updates, the costs are the other way around. Finding a good balance is a typical optimization problem.

The position management strategies operate with different efficiency in various network structures. Therefore it is important to compare the cost of different position management strategies during the planning of a concrete network to be able to choose the optimal solution. I have not found such a wide comparison model in the literature.

**Thesis 1.1** I have created a Mobility Management Model (MMM), that is able to compare the main mobility management strategies in the literature from signaling, processing in the nodes, air interface usage and security point of view. I have defined cost functions for the different aspects [J7,J9,J11,C7,C8,C11,C14] (Chapter 2.1).

The mobility management model (MMM) is an extension and generalization of the simple mobility model in the literature [8]; my first step during its creation was the preparation of the network’s and the mobile’s entity model. One possible realization of the network modeling is using a concrete, specific network representation to describe a protocol. The obvious disadvantage of the above is that it cannot be used for other protocols and in other contexts. Therefore in many dissertations/papers the network is described by a simple, general parameter, like the mean/average distance between the nodes. Any network can be described this way, but the power of the model is rather weak if one would like to compare the behavior of different protocols on certain network structures. Summarizing the experiences I combined the two solutions, defined the general characteristics (\(n\)), which are independent of the technology and also introduced a special, defined parameters needed to properly describe the protocol investigated (\(\varepsilon\)). Details can be found in Chapter 2.1 of my dissertation. Table 1 and Figure 1 introduces the parameters.
Table 1: The parameters of the MMM

<table>
<thead>
<tr>
<th>General network parameters ($\mathbf{g}$)</th>
<th>Technology and cost constants ($\mathbf{c}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$ – weighted adjacency matrix</td>
<td>$c_u, c_d$ – wired signaling cost constants ex.: the unit cost of one update on a link, etc.</td>
</tr>
<tr>
<td>$B_\Pi$ – transition matrix</td>
<td>$c_o, c_6, c_{co}, c_{dis}, c_c$ – processing cost constants, ex.: registration cost, forwarding cost, etc.</td>
</tr>
<tr>
<td>$m$ – the average distance from the central agent</td>
<td>$c_{au}, c_{ad}$ – air interface cost constants, ex.: the cost of uplink message, etc.</td>
</tr>
<tr>
<td>$g_H$ – the average distance from the nearest hierarchical junction</td>
<td>$c_{su}, c_{sdt}, c_{sdu}$ – Security cost constants.</td>
</tr>
<tr>
<td>$g_T$ – the average distance between two MAPs</td>
<td>$g_T$ – the average distance between two MAPs</td>
</tr>
<tr>
<td>$g_C$ – the average distance of MAPs from the main MA of a Location Area</td>
<td>$g_C$ – the average distance of MAPs from the main MA of a Location Area</td>
</tr>
</tbody>
</table>

Using the introduced parameters I defined cost functions for the mobility management strategies, which can be categorized into four bigger groups: signaling cost ($C_{\text{SIGN}}$), processing cost ($C_{\text{PROC}}$), cost of radio/air interface ($C_{\text{AIR}}$) and security cost ($C_{\text{SEC}}$). I divided the strategies into five bigger groups: centralized (CENT, e.g. MobileIP), hierarchical (HIER, e.g. Hierarchical MIP), wireline/wire tracking (WIREDT, e.g. LTRACK), wireless tracking (WLESST, e.g. HAWAII) and paging type (CELL, e.g. GSM, CIP).

The cost functions can be determined in general form, shown in equation (1). The general form is a utility function of the network and general parameters ($\mathbf{g}$) and the technological constants ($\mathbf{c}$), differentiated by the mobility parameters $\lambda(t)$, which expresses the handover intensity and $\mu(t)$, which represents the incoming call intensity. Based on the above the general equation is the following:
\[ C = E\left[ \int_{-\infty}^{\infty} f_h(n,c)dP(\lambda(t)) + \int_{-\infty}^{\infty} f_p(n,c)dP(\mu(t)) \right], \quad (1) \]

where \( \lambda, \mu \) are the intensities of a Poisson process, functions \( f_h, f_p \) are unique for each mobility management strategy and are linear functions of the network parameters \( n \) and the technology constants. \( f_h \) represents the handover cost and \( f_p \) the transport cost of one paging/call. Since the integral is taken with respect to a discrete Poisson process it can be substituted by a simple sum, introducing mobility rate \( \rho \). The mobility rate, \( \rho \) is the handover process conditioned on no paging. According this the above cost function simplifies to:

\[ C = \rho \cdot f_h + (1 - \rho) \cdot f_p, \text{ where } \rho = \frac{\lambda}{\lambda + \mu}. \]

All cost functions are detailed in Chapter 2.2 of my dissertation.

**Thesis 1.2** The MMM is able to choose the optimal mobility strategy or to compare existing management protocols for an \( N \times A > \Pi \times C > \) network and \( c \) cost constant vector \([J7,J9,J11,C7,C8,C11,C14]\) (Chapter 2.3).

There are two approaches for using the MMM framework. The first one is if the network description \( (n) \), the supposed/hypothetical cost constant vector \( (c) \) is available and the mobility management strategy with the lowest cost is determined. The other field of application is for comparing concrete, existing management protocols on one or more specific network configurations. In this case the strategy if the protocol is given, the cost constants specify the exact management method.

The description of the network is equivalent of defining matrices \( A \) és \( B_\Pi \). Afterwards the previously introduced parameters (Table 1) can be determined. The mobile entity’s handover frequency between two MAPs can be modeled by a Poisson process. Let \( B_\Omega \) denote the rate matrix of the continuous Poisson chain describing the mobile’s characteristics. Transition matrix \( B_\Pi \) can be deducted/reduced from \( B_\Omega \). The network topology, connections between MAs are represented in matrix \( A \). I compared the cost functions of the five main strategies in different aspects (Figure and 3) using Mathematica [8] application.

![Figure 2](image-url)  

Figure 2: The summed cost functions of centralized-like (one-dot-dash), hierarchical-like (two-dot-dash), wired (dashed) and wireless (solid) tracking-like, cellular-like (dotted) approaches as a function of the mobility ratio: \( \rho \). The two figures show the costs on two different simulated networks, \( g_T, g_C \) are significantly less (meshed) on the right side figure.
I analyzed the full cost of mobility management approaches in three different network scenarios. Three different, typical network topologies are investigated with high and low handover intensities. Table 2 shows the result, with bold type the lowest-cost, in fact the optimal, with gray background the highest cost strategy in given configuration.

Table 2: The full costs of approaches in different network scenarios

<table>
<thead>
<tr>
<th></th>
<th>More meshed</th>
<th>Long-Tree topology</th>
<th>Wide-Tree topology</th>
<th>More meshed</th>
<th>Long-Tree topology</th>
<th>Wide-Tree topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>CENT</td>
<td>4</td>
<td>16.32</td>
<td>6.72</td>
<td>8.02</td>
<td>24.96</td>
<td>11.76</td>
</tr>
<tr>
<td>HIER</td>
<td>3.86</td>
<td>14.94</td>
<td>6.36</td>
<td>7.60</td>
<td>20.82</td>
<td>10.72</td>
</tr>
<tr>
<td>WLESST</td>
<td>3.47</td>
<td>10.81</td>
<td>9.38</td>
<td>7.41</td>
<td>11.7</td>
<td>10.85</td>
</tr>
<tr>
<td>WIREDT</td>
<td>3.93</td>
<td>13.75</td>
<td>22.64</td>
<td>7.61</td>
<td>20.65</td>
<td>16.64</td>
</tr>
<tr>
<td>CELL</td>
<td>7.04</td>
<td>103.2</td>
<td>18.48</td>
<td><strong>4.54</strong></td>
<td>19.8</td>
<td><strong>6.54</strong></td>
</tr>
</tbody>
</table>

Figure 2 shows that the total costs of the centralized, hierarchical and wired tracking approaches increases with the handover frequency rising as all handovers generate extra costs to the network. The wireless tracking solution rather spares the cost of handover with the mobility rate increasing. It is trivial for the cell types protocols that the lower the number of the incoming calls is the lower signaling traffic is generated on the network. Table 2 shows similar results for analyzing Figure 2, quantified.
5.2 General mobility modeling and location prediction based on Markovian approach – MMCF-Markov Movement model Creator Framework

In wireless networks one of the most important tasks is modeling the users’ movement. A precise model based user movement prediction supports network planning, self-configuring, self-healing networks, fraud detection an admission control improving the quality of service (QoS).

The different movement models in the literature can be divided into two major groups, individual [11][12] and group [13][14] movement models. There are diversified mathematical solutions in both groups, from which the most popular is the Markov model.

The Markov model is a stochastic, mathematical model; its simplest representation is the Markov chain. The Markov chain denotes a stochastic process meeting the Markov characteristics. The Markov characteristics mean that the actual present and future state of the system is independent of the past states. Markov models are widely used in solution of telecommunication problems [10] throughout the world.

Thesis 2.1 I have created a classification method for the discrete-time Markov movement models (MMCS – Markov Movement-model Creator System). Every M model which states express the staying in an area is definable exactly in \( M^{<L,R,O>^L} \) form, where \( L \) is ‘level’, \( R \) is ‘resolution’ and \( Q \) is the ‘depth’ of the model. The solution is able to compare and to analyze the efficiency of Markov movement models in different network scenarios [J1,J2] (Chapter 3.2).

I have investigated the Markov models that can be found in the literature and based on these I have extended the model introduced in the previous chapter with specific parameters describing the user movement.

I have created my classification system, where two main type models have been distinguished, User-Centralized and Access Point-Centralized. The latter one is further separated into two subtypes. Figure 4 depicts this main classification.

![Figure 4: The main classifying of the Markov mobility models](image-url)

---

10
Besides the simple classification I have defined attributes describing the main characteristics of the models. These are the following:

- **Level** (L): models can treat users or user groups separately introducing a new dimension, new *level* into the given movement model. This is justified when certain users’ movement significantly differs from the usual or we would like to distinguish them with for example a multi-level movement model created for a different admission control strategy.

- **Resolution** (R): the model’s *resolution* describes if we join two or more MAPs and represent them with one state in the Markov chain. This usually happens when the given MAPs shouldn’t be handled separately due to the fact that users behave similarly within these or simply a infinitesimal number of users are connected.

- **Depth** (Q): in case of controlled movement the users’ actual location can highly depend on the previously visited access point. How much the given model takes it into account, how many previous steps in the users’ movement are used is represented by the depth. The Markov property is not harmed by this as we are not introducing the theory of memory into the Markov model, only to the user movement situation represented by this. In case of MMCS we always use first class Markov chains.

Using these attributes much more complex and sophisticated models can be determined. With the help of MMCS I have carried out an analytic cost evaluation using some Markov models found in the literature with different network configurations (Chapter 3.5.2). I have investigated the models in different network environments (Figure 5). The cost of the models were determined as a function of two parameters, the number of states \( \text{n}_{\text{states}} \), and the theoretical error \( E_T \):

\[
C_{\text{MM}} = \text{log} \text{n}_{\text{states}} + E_T . \quad (2)
\]

The theoretical error \( E_T \) is the summarized and standardized wrong predictions by MAPs. This is implied by handling certain MAPs joined in the model or not taking into account previous steps with sufficient depth (RM, M3 [3], o-th [15]) and during the prediction we calculate the users in the MAPs with linear distribution.

I have used special user environment to highlight the advantages and disadvantages of certain models. The network scenarios are shown in Figure 5. These are the following:

- Scenario a. : 'A park, uniform user distribution'
- Scenario b. : 'Simple road'
- Scenario c. : 'Highway to city'
- Scenario d. : 'Directional motion'
- Scenario e. : 'Differentiated users'
Figure 5: The main network scenarios in performance analysis with MMCS

The result of the analysis presented in Table 3, details are shown in chapter 3.2.8 of my dissertation.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RM (n_{states}=3)</td>
<td>0.47</td>
<td>1.81</td>
<td>1.28</td>
<td>2.11</td>
<td>2.11</td>
</tr>
<tr>
<td>M3 (n_{states}=4)</td>
<td>0.6</td>
<td>1.93</td>
<td>0.6</td>
<td>1.93</td>
<td>1.93</td>
</tr>
<tr>
<td>M7 (n_{states}=8)</td>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
<td>1.9</td>
<td>1.9</td>
</tr>
<tr>
<td>2nd (n_{states}=49)</td>
<td>1.69</td>
<td>1.69</td>
<td>1.69</td>
<td>1.69</td>
<td>2.94</td>
</tr>
<tr>
<td>3D (n_{states}=16)</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>2.1</td>
<td>1.2</td>
</tr>
</tbody>
</table>

One can see which models perform well is which special situations. This implies that the certain model has to be carefully chosen or constructed for the given network location.

If we would like to create an own Markov movement model, take into account all historical data (depth) and treat all different user behaviors distinguished it will result in a model with high number of states. There for it is practical to allow a minimal error and find the Markov model with the least number of states.

**Thesis 2.2** I have developed a general, discrete-time Markov Movement model Creator Framework (MMCF) with states representing the user's location. Using MMCF an optimal number of states \( M(\mathcal{L}, \mathcal{R}, \Omega) \) Markov model can be determined for every \( N(A, D, B) \) network and for \( \varepsilon \) error vector \([J1, J2]\) (Chapter 3.3).
The idea behind MMCS is to determine a Markov movement model with minimal number of states for a given network that meets the initial conditions. These conditions are determined by the fault vector \( \varepsilon \). In the model creator framework I have created recommendations for determining the parameters \( L, R, \) and \( O \) (chapter 3.3 of my thesis).

The complexity of the given model could be denoted with the number of states that is the following:

\[
    n_{\text{states}} = \sum_{l \in L} n_m / n'_M \cdot w'_{nm} w'_{O},
\]

where \( n_m \) is the number of MAPs in the examined network, \( n'_M \) is the number of new MAPs after merging for the resolution \( (R) \), \( w'_{O} \) is the weighted average depth on level \( O \), and \( w'_{nm} \) is the average number of neighboring MAs on level \( l \).

The main point of the proof is that the attributes are defined by functions \( L = f_l(A, D, B, \varepsilon) \), \( R = f_r(A, D, B, \varepsilon) \) and \( O = f_o(A, D, B, \varepsilon) \) and \( n_{\text{states}} = f_{\text{states}}(L, R, O) \) can be derived from these.

Function \( f_{\text{states}} \) indirectly depend on the error vector by \( f_l, f_r, \) and \( f_o \) and as a function of this the number of states is strictly monotonically decreasing. Hence the proof is indirect, if there would be a Markov model for parameters \( A, D, B, \varepsilon \) with a lower number of states, the corresponding \( \varepsilon^* \) would exceed the error vector \( \varepsilon \) given as the original condition (\( \varepsilon^* > \varepsilon \)).

With this step I have created the possibility for creating arbitrary discrete time Markov models. Below I have investigated the movement models that can be created by the MMCF and improved one of the known models, M3 [3].

As we could see while investigating the cost of the movement models, the M3 model properly fitted/adjusted to the access points is perfectly capable/suitable for modeling one or two directional movements. Although it shows relevant error/inaccuracy in a highly populated city with irregular road system or a park, where the movements have multiple directions. As a mitigation I represented all \( n \) adjacent/neighboring cells in the access points with a Markov state instead of merging the left and right side cells used in the M3 [3] model. I created a much more precise, so called Mn modell. The number of states in the Mn modell is \( n+2 \).

**Thesis 2.3** I have created the \( n+2 \) state Mn model, which is an enhancement of the access point-centralized M3. The Mn mobility model is more accurate than the 3-state Markov model (M3). The accuracy difference in percentage is:

\[
    \sum_{j \in \text{MAP}_{a} \cup S_{\text{adj}}} \left[ \frac{1}{3} \left( \frac{N_{k'}}{N_{k}} \right) N_{k} \cdot \lambda \right] \% \text{, where } L, R \text{ the right and left area MAP groups}
\]

in M3 model, \( N_{a} \) the number of users in \( \text{MAP}_{a} \), \( S_{\text{adj}}^{i} \) is the index set of neighbour MAPs of \( \text{MAP}_{i} \) and \( \lambda \) is the average handover rate. The calculation complexity of the model is \( o(n^3 + n + 1/n) \) [J3,J5,C2,C4,C5,C6] (Chapter 3.4).

In the theoretical model the cells are represented with a hexagon covering the plane, so in a cluster each cell has 6 regular neighbors. However in reality this does not exist, the cells do not cover such regular areas and usually do not have 6 neighbors. Therefore it is practical to calculate with changing, \( n \) number of neighbors in case of a given MAP.
MAPs belonging to one cluster are mapped into the states of a Markov chain in the Mn model, that is the \( n \) neighboring MAPs and the central MAP, plus all MAPs outside the cluster are represented by an 'outside' state. This results in \( n+2 \) states. Two special type of the Mn model are M3 and M7 [J1, J3]. The naming of these models was created during a previous interpretation hence M3 contains 4, M7 contains 9 states.

I have calculated the inaccuracy of M3 compared to Mn by weighting the two models prediction difference for the next moment with the number of actual users. This is shown in the following equation:

\[
\frac{N_{i}^{t+1}_{M3} - N_{i}^{t+1}_{Mn}}{N_{i}^{t+1}_{Mn}},
\]

where \( N_{i,A}^{t} \) is the predicted number of users in cell \( i \) at time \( t \) by model \( A \) and \( N_{i}^{t} \) is the actual number of users in cell \( i \) at time \( t \). Using this for all cells and creating an average gave the result mentioned in my thesis. The detailed proof is contained in Chapter 3.4.2 of the dissertation.

The neighboring cells in the cluster are divided into two groups in the M3 model. However any other distribution is possible. Therefore the statement regarding the accuracy can be generalized the following way:

\[
\sum_{k, j \in G} \sum_{i, \text{MAP} \in S} \left\{ \frac{1}{n_G + 1} \left( \frac{N_{i,j}^{t}}{N_{k}^{t}} \right) \right\} N_{k} \cdot \lambda,
\]

where \( G \) is the set of the grouped areas, and \( n_G \) is the number of groups in the model.

The calculation complexity is determined by three main factors: processing capacity needed for the determination of the crossing probabilities, calculation of a steady state of the gauss elimination and the prediction. Taking these into account the complexity can be determined as \( o(n^3 + n + 1/n) \) (Chapter 3.4.2).

I have investigated the accuracy of the Mn model using the simulation as well. The simulation was prepared using the OMNet++ framework (Figure 6).

![Figure 6: The TAEV values in RW, ExtRW, M3 and M7 models in case of one directional movements](image)
I have investigated the accuracy of the MMCF as well. The MMCF generated Markovian approach holds the average error rate, follows the changes in user motion appropriately and is able to learn the directional motion patterns, which proves the strength of the Markov Model Creator Framework (Figure 7).

Figure 7: The TAEV values in RW, M3, M7 and optimal MMCF model

The network operators monitor and log the mobiles’ movement and could try to predict their future location. To be able to determine the users’ future distribution we need to deduce the steady state probabilities of the Markov chain and for this we need to define the transition probabilities. The transition probabilities can be defined as the relative frequencies calculated from the network logging. This process is not accurate, the determination contain errors depending on the number of samples. I have named this error as *empirical* error and denoted it with $E_P$.

**Thesis 2.4** I have introduced a method to process the network traces and determined the transition probabilities of $M_n$ model. Using this method I have determined the empirical error ($E_P$) of the user distribution determined by the $M_n$ model:

$$E_P = H_S + nH_{N_k} + H_{NO},$$

where

$$H_S = n(1-q-3v)H_{N_k} + (1-np)H_S - \frac{4}{\sqrt{2}}(nH_{N_k} + 1 + H_S),$$

$$H_{N_k} = H_S p + H_{N_k}(2v + q) + H_{NO}v + \frac{1}{\sqrt{2}}(H_S + 3H_{N_k} + H_{NO} + 1),$$

$$H_{NO} = nH_{N_k}v + H_{NO}q + \frac{n}{\sqrt{2}}(H_{N_k} - H_{NO}),$$

and $p,q,v$ are the averages of the transition probabilities of the $M_n$ Markov model [J3,C2,C4] (Chapter 3.5).

Network operators log the events in their network, amongst them the users’ cell transitions between MAPs. I have developed a procedure for summing up these latter events. Using this based on relative frequency the transition probabilities of $M_n$ model can be determined. However the relative frequency will never be completely precise/accurate hence the user distribution resulting from these transition probabilities will have empirical error $E_P$ as well.
I carried through the error resulting from the determination from relative frequency throughout the calculation of the steady state probabilities. We only approximate the Mn model’s transition probabilities \((p_i, q_i, v_i, \lambda)\) during the determination with the calculated values \((\hat{p}_i, \hat{q}_i, \hat{v}_i, \lambda)\). We solve the equation \(P = P\Pi\) using the transition matrix constructed from the calculated values, that is \(\hat{\Pi}\) is used instead of \(\Pi \). In the solution the steady state probability containing errors, \(\hat{P}\), is determined instead of \(P\). Summing up the relations:

\[
\hat{P} = P + H, \quad \hat{\Pi} = \Pi + E
\]
\[
\hat{P} = \hat{P}\hat{\Pi} \rightarrow P + H = (P + H)(\Pi + E), \quad (6)
\]
\[
H = PE + H\Pi + HE
\]
\[
H = H\Pi + \hat{P}E
\]

where matrix \(E\) contains the error of the transition matrix, vector \(H = H_{in}\) means the error of the steady-state transition probability; the determination of \(H\) was my goal. Detailed deduction is included in Chapter 3.4.2.1 of my dissertation.

Using the plotting abilities of *Mathematica* [6] I have investigated the error of model Mn as a function of \(p, q, v\) and \(n\). The aggregated transition probabilities \(p, q, v\) describe the basic characteristics of the user movements, I have analyzed some characteristics of these. \(p\) always describe movement from the central cell toward a neighboring cell, \(q\) mean staying in the cell, \(v\) corresponds to transitions between neighboring cells. Using these the empirical error of Mn models prediction can be determined for different user movement patterns and behaviors.

![Figure 8: Empirical error as a function of the number of states. a: \(p=0.1, q=0.9, v=0.1\) (slow motion), b: \(p=0.1, q=0.1, v=0.1\) (heading to the central MAP), c: \(p=0.9, q=0.1, v=0.9\) (fast motion)]
Figure 9. Empirical error with $p$ as a function of the $p$ transition probability. a: $q=0.9, v=0.3$ (slow motion), b: $q=0.1, v=0.9$ (fast motion)

One can see in figure 8 and 9 that the empirical error decreases until a certain point and in all cases will increase by increasing the number of states. That this increase occurs at a higher number of states in case of slow motion and a lower number of states in case of fast motion and their steepness is different can be explained by the fact that lower in case of slow motion the number of different movement patterns is lower, hence the calculation error is lower. In case of fast motion the number of different samples is significantly higher.

It can be clearly seen as well that in an situation a point is given where the empirical error is 0. In my dissertation I have analyzed by what $p$, $q$ and $v$ aggregated transition possibilities does this ideal situation exist (figure 10 shows this).

Figure 10. Eliminated empirical error ($E_r=0$) as a function of $p, n$, and $q, v=0.02$. 
5.3 Client-driven Mobility Frame System – Mobility from a brand new point of view

I have compared the different position management strategies using more aspects during the creation of the mobility management model (MMM, Chapter 2.1). It can be simply seen that different mobility management strategy results the lowest total cost \( C \) for certain networks. Typically they use a given strategy in a network, which supposedly results in optimal cost for the majority of the mobile users, but surely not for all. The ideal situation would be if all of the mobiles would follow such strategy that results in the lowest total network cost for each and therefore they would only use the minimally needed resources to realize their own position management.

The third greater research area of my dissertation is creation of a self-organizing location management procedure to be able to realize the ideal situation introduced above.

**Thesis 3.1** I have introduced a new, in mobile network used self-organizer location management framework, the CMFS (Client-driven Mobility Frame System). In CMFS every mobiles manage the mobility for itself and network provides only basic features, the connectivity and administration. Using CMFS every mobile or group of mobiles is able to select the mobility strategy which cause the minimum signaling cost from the network point of view [J4,J6,J9,C3] (Chapter 4.1).

The Client-driven Mobility Frame System that I have developed is fundamentally different from the classical approaches. It is inspired (but not dependent) by the fact that the mobile users typically move within a range of access points and rarely leave their environment. For a mobile device to manage its movement properly it is necessary to know and store the access points (MAPs), mobile agents (MAs) that have been contacted. I refer to these and their connections as logical network (LN). The logical network has to be updated by the mobile continuously to be able to make a proper decision. Its size is dependent on the algorithm run by the MN.

Besides the logical network the mobile devices continuously keeps and updates internally the following-like matrix:

\[
Q_{\text{cost}} = \begin{bmatrix}
\infty & C_{\text{strat1} \rightarrow \text{strat2}} & C_{\text{strat1} \rightarrow \text{strat3}} & \cdots & C_{\text{strat1} \rightarrow \text{stratn}} \\
C_{\text{strat2} \rightarrow \text{strat1}} & \infty & C_{\text{strat2} \rightarrow \text{strat3}} & \cdots & C_{\text{strat2} \rightarrow \text{stratn}} \\
C_{\text{strat3} \rightarrow \text{strat1}} & C_{\text{strat3} \rightarrow \text{strat2}} & \infty & \cdots & \cdots \\
\cdots & \cdots & \cdots & \cdots & \infty \\
C_{\text{stratn} \rightarrow \text{strat1}} & C_{\text{stratn} \rightarrow \text{strat2}} & \cdots & C_{\text{stratn} \rightarrow \text{strat(n-1)}} & \infty
\end{bmatrix}, \quad (7)
\]

\[
Q_{\text{change}} = \begin{bmatrix}
C_{\text{strat1}} & C_{\text{strat2}} & C_{\text{strat3}} & \cdots & C_{\text{stratn}}
\end{bmatrix}
\]

where \( C_{\text{strat1} \rightarrow \text{strat2}} \) is the cost of changing from \( \text{strat1} \) to \( \text{strat2} \), \( C_{\text{strat}} \) is the summarized, possible cost of management strategy \( \text{strat} \). During the movement/motion of \( MN \) upon connecting to a new MAP, it will update its matrix \( Q_{\text{change}} \) and vector \( Q_{\text{cost}} \). The \( Q_{\text{change}} \) matrix and \( Q_{\text{cost}} \) vector and investigates if it is worth to change from the current strategy or not:
If the mobile finds a cheaper strategy it will perform the technological handover minimizing the possible costs.

The MAs in the network has to meet certain criteria to be able to serve the MN and its selected algorithm. If the nodes have this capability the mobiles can use any mobility management algorithm, independently of each other.

The MA has to know at all times where to route packets to the mobile. The link between the mobile’s general and actual address is stored in a database. (The address can be an IP address, but usage of a different protocol is also possible). If MA receives a packet addressed to the mobile, its check amongst the records towards which MA or MAP to forward it. If it cannot be found in the records, the request is discarded. If the MN connects to a new MAP after a successful handover, it registers there, but can also decide to continue the registration procedure on a MA on a higher level. I have developed a message structure (C-MS, CMFS Message Structure) to support this signaling process. The detailed introduction and the discussion of related requests can be found in Chapter 4.1.3 of my dissertation. Using CMFS both the classical and the new, more complex mobility algorithms can be realized by using the control of the mobile device.

The main advantage of this solution compared to the traditional protocols, GSM, Mobile IP (IPv4 and IPv6) is that no functionality related to the protocol to be used has to be implemented in the network elements and the network structure. The network can remain simple, the nodes only have to interpret and execute simple commands, registration, message forwarding, and deletion of registration. The logic and the control, hence the complete management algorithm runs in the mobile. The main advantage of the system results from this, all mobiles can chose for itself the optimal, most cost effective mobility strategy with the least network cost and even change the strategy during operation.

5.3.1 Application of CMFS

CMFS introduced in the previous section is only a framework and a tool to reduce the costs occurring in a network. To be able to reach the mentioned ideal state the appropriate algorithms have to be created in the framework. Using these the mobile devices are able to realize their position management with the least cost for themselves and the network as well.

5.3.1.1 Tracking-like approaches

As the first step I have investigated the tracking type position management strategy. The basic question with the tracking solutions is when the MN shall perform a tracking handover and a normal handover (Figure 11). In case of normal handover the mobile entity refreshes it record at the central agent (HA), while in case of tracking handover only at the previous connection point.
The classic LTRACK solution utilizes a Markov chain to determine the optimal number of *tracking handovers* that are always followed by a *normal handover* [7]. However, this is a global number, all mobiles apply it in a given network. It would be more efficient if all mobiles or a group of mobiles would determine the optimal *tracking handover* number. To compare the two theories I have outlined a simple handover situation, where the MN is at MAP\(_{i-1}\) and moves towards MAP\(_i\) (Figure 11). I have used the following parameters in the investigation:

- \(u_{i-1,i}\) – the cost of tracking handover from MAP\(_{i-1}\) to MAP\(_i\), generally \(u_t\)
- \(d_{i-1,i}\) – the cost of call delivery from MAP\(_{i-1}\) to MAP\(_i\), generally \(d\)
- \(u_{i-1}\) – the cost of normal handover from MAP\(_{i-1}\) to HA
- \(d_{i-1}\) – the cost of call delivery from HA to MAP\(_{i-1}\)
- \(u_i\) – the cost of normal handover from MAP\(_i\) to HA
- \(d_i\) – the cost of call delivery from HA to MAP\(_i\)
- \(p\) – the probability of call delivery to mobile node at MAP\(_i\)
- \(r\) – the relative frequency of tracking handover

Let me assume that the distance between MAP\(_{i-1}\) and HA and MAP\(_i\) and HA is equal, therefore \(u_i = u_{i-1} = u\), and \(d_i = d_{i-1} = d\).

![Figure 11. Tracking handover decision](image)

As a result of analysis I have created an alternative, tracking type solution the PTMIP (Personal Tracking Mobile IP).
I have created the PTMIP (Personal Tracking Mobile IP) tracking-like position management protocol in CMFS. The cost of PTMIP is lower than the cost of LTRACK with $C_{\text{diff}}^{\text{LT,PT}}$, $C_{\text{diff}}^{\text{LT,PT}} = \min\{(1-r)(u_i + d_i + p - u) + u \cdot r(u - u_i - p d_i - p u)\}$ [J4,J6,J7] (Chapter 4.3).

The main idea of PTMIP is that the MN itself decides on performing a tracking handover or a normal handover based on the measured costs. In the classic LTRACK solution, as mentioned previously, the network determines a global tracking handover number valid for all mobiles. Based on the handover simulated (Figure 12) and the parameters used in the investigation we can assume that the number of tracking handovers in the LTRACK solution is $r/(1-r)$. Based on these the cost of LTRACK is:

$$K_{\text{LT}} := r(u_i + p(d_{i-1} + d_i + u_i)) + (1-r)(u_i + p d_i).$$  \hspace{1cm} (9)

When using the PTMIP strategy the MN measures the cost of a possible tracking handover and normal handover and chooses the optimal step accordingly. As a function of this the cost of PTMIP:

$$K_{\text{PT}} := \min(K_T := u_i + p(d_{i-1} + d_i + u_i), K_N := u_i + p d_i).$$  \hspace{1cm} (10)

The difference between the two calculated cost $K_{\text{PT}}$ and $K_{\text{LT}}$ gives the $C_{\text{diff}}^{\text{LT,PT}}$ value mentioned in the thesis. $C_{\text{diff}}^{\text{LT,PT}}$ only depends on the cost of the update $u$, it is independent of the call delivery cost, $d$. The abstract location of the cost functions as a vary of $u$ is shown in Figure 13.

![Figure 13. The cost of tracking-like solution in vary of $u$.](image)

Using the situation shown in Figure 13 we can assume that MN moves from MAP$_{i-1}$ to MAP$_i$, only knows $u_{i-1}$ in its current position and visited MAP$_i$ first. Using these for the calculation the proper decision is the tracking handover, because at the steady point $u_{i-1}$ is greater than $u^*$. The cost of PTMIP will only be greater than the cost of LTRACK, if when recognizing $u_i, u_i < u^*$ and the decision is tracking handover. However we had our base assumption, that $u_{i-1} = u_i = u$. Furthermore if MN has previously visited access point MAP$_i$, than $u_i$ is known and calculating with that the decision is obvious.
5.3.1.2 Cellular-like approaches

With the widespread use of the modern PLMN technologies the cell-type strategies became more popular. The main point of this solution is that the neighboring cells are organized into a higher level structure, so called paging area (PA), (Location Area in case of GSM, Routing Area in case of UMTS, Tracking Area in case of LTE, Paging Area in case of CIP). The inactive mobiles do not update their location information in case of crossing cell borders, only at paging areas borders. In case of call delivery, as the exact position is unknown, the discovery of the destination station/device is done using broadcast. The idea behind the solution is that the MN moves in a certain set of cells, rarely leaves this area, hence the cost of signaling can be minimized. Grouping and joining the cells like this is an NP-complete problem; more scientific work is dedicated to its solution. However this problem can also be perceived on the level of individuality, and different PA coverage can be created for each MN.

Thesis 3.3 I have created the PCMIP (Personal Cellular Mobile IP) a cellular-like position management strategy in CMFS, which provides optimal paging area coverage for mobile nodes [J4,J6,J7] (Chapter 4.3).

The MN is able to create its own paging coverage using PTMIP and therefore can minimize its signaling cost. The disadvantage of this solution is that the mobile entity has to do extra calculations, maintain the visited access points and their corresponding data, has to calculate probabilities and intensities.

The algorithm works the following way: if the mobile connects to a new access point that is not registered in its own database, it has to decide to add the new access point to the actual PA, create a new PA or reorganize all access points handled by the mobile. The cost of reorganization also has to be taken into account at all steps.

Using the signaling system introduced previously the building blocks of the algorithm are the following:

- The specific cost of maintaining a MAP_i:
  \[ c_i = \mu_i d_i + \sum_{j \not= i} \lambda_{i,j} u_i = \mu_i d_i + \lambda_{i,j} u_i \]  \hspace{1cm} (11)

- The specific cost of maintaining a new PA, which is the incorporation of MAP_j into MAP_i:
  \[ c_{i,j} = \mu_j (d_i + d_j) + (\lambda_{j,i} - \lambda_{i,j}) (u_j + u_i) - \lambda_{i,j} u_i \]  \hspace{1cm} (12)

- The specific cost of maintaining a new PA, which is the incorporation of \( M \) set into MAP_i:
  \[ c_{i,M} = \sum_{m \in M} \mu_m (d_i + d_{i,m}) + \sum_{m \in M} \lambda_{m,i} (u_m + u_i) - \sum_{m \in M} \lambda_{m,i} u_i \]  \hspace{1cm} (13)

where \( \mu_i \) the number of delivered calls at MAP_i, \( \lambda_{i,j} \) the number of handovers between MAP_i and MAP_j, \( \lambda_{j,i} \) number of handovers to MAP_j.

The algorithm, detailed in Chapter 4.3.2 of my dissertation uses these building blocks.
6 Application of the results

The comparison of the mobility protocols can be different. Their performance depends on the implementation and the network, a real comparing analysis can be used in special cases. The Mobility Management Model (MMM) I have introduced is capable of showing which approaches result in a better solution in case of specific network parameters, what can be the bottleneck of the mobility strategy. This can be a guideline to design the optimal protocol for a specific environment and function. MMM is useful for both the network operators and the researchers/ones doing scientific work as well. The operators can use it to design their networks and fine tune the parameters, while the researchers can take advantage of comparing mobility protocols developed by them.

The creation of MMCS and MMCF are the topmost results of my research in the mobility modeling. Using MMCS the discrete time, Markov movement models representing steady states can be compared; using MMCF new models with minimal number of states can be created based on the network parameters. The resulted model can be used as a basic building block for self-configuration and self-healing networks. The network can reconfigure itself based on the predicted user distribution as a preparation for serving a coming busy hour traffic. In case of self-healing networks a major deviation from the predicted distribution of the users from the movement model could imply the failure of the radio interface. Mobility models play a major part in creation of user profiles also and certain frauds can be detected by their usage.

Finally I have created such a mobility protocol using CMFS, which completely breaks away from the previous ideas and architectures. Both the mobility management and algorithm is determined by the moving entity. The most important result is that my method is capable of realizing more, classical and new mobility protocols without the reconfiguration of the network. As all mobiles can maintain an adapted system for their own movement that using a proper algorithm the signaling traffic can be minimized more efficiently as before. Starting from my work, in the future several alternative position management solutions can be created in the CMFS framework, customized to the particularities of the mobile networks and the needs of their operators.
7 References


8 Publications


[J7]. P. Fülöp, B. Kovács, “Numerical analysis of mobility management algorithms”, Info-Communications-Technology, Volume LxII., pp. 32-38, 2007/7,


[C1]. P. Fülöp, B. Kovács and S. Imre, Mobility Management Algorithms for the Client-driven Mobility Frame System – Mobility from a Brand New Point of View, Proc. 6th International Conference on Advances in Mobile Computing & Multimedia (MoMM2008), Linz, Austria, November 24-26, 2008.


