Proposal of PhD Dissertation Prospectus.
Reducing Updates in Cellular Networks

Georgios Varsamopoulos
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287
{Georgios.Varsamopoulos}@asu.edu

Abstract

This is the proposal of my PhD dissertation prospectus. It shows my research on *location management* (LM) in cellular-type networks. The purpose of my research is to study the fundamental problem of *update cost minimization* in cellular networks and to propose solutions to it.

The number of mobile subscribers is bigger than ever; and it is going to increase. It is expected that the mobile subscribers will reach 2 billion by year 2006 ([32]). Due to the large subscriber population and the reduction of cell size in cellular networks, the rate of registrations is skyrocketing. In contrast of the traditional *update/search* trade-off problem, a new problem arises: the problem of *minimizing the update (registration) cost* without affecting the search (paging) cost.

This report gives overview of the optimization sub-problems that have been identified with respect to the problem of update cost reduction, and presents the various solutions that have been proposed. Lastly a prospectus of the work planned to be performed during the dissertation that is being proposed.

1 Introduction

”Stay connected”. It’s today’s way of saying ”keep in touch”, a language that millions of wireless customers speak, in a world of text messaging and voice mail. Staying connected is what millions of wireless customers try to do everyday, and keeping them connected is the objective of the wireless providers. *Cellular networks* are a type of networks that support wireless connection to *mobile units* (MUs). The main characteristic of cellular networks is that the area where wireless connection is provided is divided into smaller areas, called *cells*, each of the latter covered by an array of wireless directional transceivers (antennas); the antenna of one cell are connected to the same control unit referred to as a *base station*.

The first cellular networks appeared in the 1980's, for servicing mobile telephony subscribers. They featured a Fixed Channel Allocation among cells and a TDMA MAC protocol per cell for the wireless part of the connection. At the wired part, the connections were based on circuit switching. These were the first generation cellular networks. They supported low bandwidth analog-encoding voice channels. As technology matured, dynamic channel allocation and hybrid FDMA/TDMA protocols replaced the older protocols. Uninterrupted hand-offs were possible and higher bandwidths were available. That was the *second generation* of cellular networks. Soon after, the *second-and-a-half generation* (2.5G) of cellular networks came, which featured packet switching and packet data communications. Today we are shifting to the *third generation* (3G) of cellular networks which feature high bandwidths, a plethora of consumer services and a full integration into the Internet. Cellular networks started by providing *mobile telephony*; now they provide *mobile computing and networking*.

1.1 Basics of Location Management in Cellular Networks

Location management is, in essence, the maintenance of location information of mobile network users, in terms of network topology, so that messages can be routed to the appropriate locations. Cellular networks are divided into two major parts:

The wired part : the network infrastructure that is made of the base stations, the wired links and routers among them, and the various control agents, location databases etc.

The wireless part : the part of the network that consists all the mobile units that are serviced by the wired portion over wireless connections.

When a mobile unit transitions from a cell to another cell, i.e. it switches base stations, an operation called *hand-off* or *hand-over* (in Europe) takes place which essentially allocates a communications channel and other network resources for the mobile unit in the new cell and frees the resources allocated for it in the previous cell. Hand-off is a very important operation, as in most times it has to be performed in time-critical manner in order not to interrupt open sessions between the mobile unit and the wired network. In order to make hand-offs feasible, neighboring cells *overlap* at their edges so that a mobile unit that approaches the edge of a cell can start talking to the next base station before it loses connection from the current base station. In the early 90's, the technology enabled what is called *soft hand-offs*, which allow the mobile units to keep using their channels as they traverse among cells. Soft hand-offs do not re-allocate as many resources and can be performed much faster and with lower chance of service interruption.

The location information of a mobile unit, as perceived in location management, is not geographic information, but information of what point of the wired portion the mobile unit is connected to, i.e. it is topology-based location information. Depending upon the routing scheme, the location information may vary from a simple identification number of the cell in which the mobile unit is, to a more complex naming scheme that also provides routing information. In most cellular networks, there is a location maintenance entity often abstractly referred to as *location database*. The location database can be viewed as the common ”acquaintance” of the mobile units to which they report their location and which they ask about other mobile units’ locations.

The operation of informing the location database of a mobile unit’s location is called an *update*, while the operation of asking the location database about a mobile unit’s location is called a *search*. Update and search may be as simple as a message sent between two network machines, or it can be as complex as an multi-party protocol that, depending on the conditions and the layout of the location database’s implementation, informs different parts of the location database.

Although, in concept, the location database can be viewed as a single entity, in most applied forms it has a distributed and hierarchical nature. The main reasons behind that are *scalability*, *fault tolerance* and *load balancing*. Distribution comes in play when the set of cells is divided into clusters of cells and assigning a separate location database to be responsible for the location tracking in each cell cluster. A hierarchy is also needed in order to keep track of which location database is tracking each mobile unit.

There are many location management schemes that have been proposed in the literature. Some of them can be found in [6, 7, 9, 18, 24, 39]. The most dominant and most studied location management scheme is the VLR/HLR architecture, described in the next subsection.

1.2 VLR/HLR architecture: registration and paging

Most of the 2G *mobile telephony* and PCS networks use a distributed and hierarchical layout commonly called the *VLR/HLR architecture*. Examples of this architecture are the IS-41/IS-95 [39, 40] networks in North America and GSM [41] and GPRS [42] in Europe. In this architecture, the cells are logically organized into *location areas*(LAs). Each location area has a ”local” location database called *location register* (LR) that acts as the part of the abstract location database for its location area. The location register and the cells in the location area need a lot of bandwidth among them, mainly for performing hand-offs and delivering calls, therefore cells not only cover a contiguous geographical area but also make up well connected subnet that includes the local database and router. Figure 1 visualizes the division of cells (circles) into location areas (squares). The methodologies and techniques for building a cellular network from the ground up fall under the topic of *network*

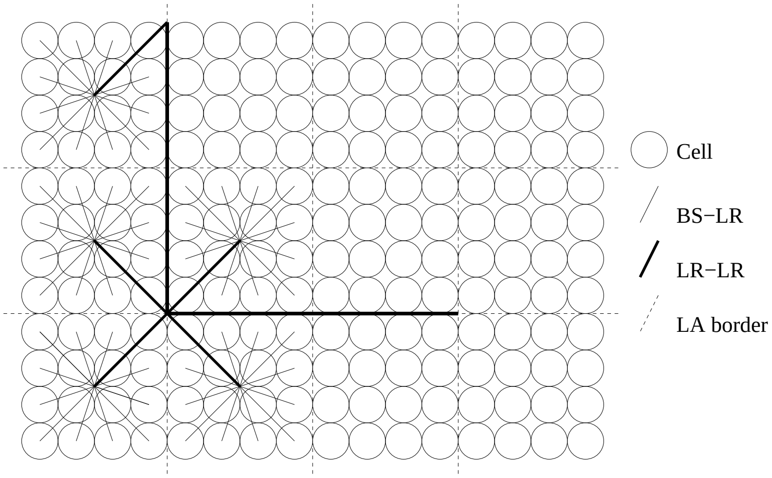


Figure 1: **The cells of the coverage area are divided into location areas. Cells in a location area are usually connected via star topology.**

planning and design [4, 16, 19, 26, 28, 48] and is of as big importance as the optimization of the location management protocols.

In the VLR/HLR architecture, mobile units are assigned to location registers, so that every mobile unit has a designated location register called the unit's *home location register* (HLR). Accordingly, the location area of the HLR is called the *home location area* (HLA). The HLR of each mobile unit can be obtained by a universally known mapping function that gives the HLR id with the mobile unit's id as input.

As long as a mobile unit resides within its HLA, it updates its location (i.e. what cells it is in) to the HLR only at fixed time intervals. This is referred to as *time-based dynamic update* [7]. When a call arrives for that mobile unit, the HLR is queried and, since it knows the last location reported by the unit, it signals all the cells around the reported cell. This signaling operation is called *paging*. The set of cells that are paged is called *paging set* or *paging area*. The size of the paging area depends upon the time interval of the updates and upon the maximum speed of the mobile unit. There are various paging schemes that optimize paging time and paging cost (some known methods, among others, are *selective paging* [1], *profile-based paging* [31, 51], *predictive paging* [15, 49], *adaptive paging* [13] and *sequential paging* [33]). When the unit receives the paging signal, it responds to the paging and the call is established.

When the unit leaves the HLA, then a signaling operation called *registration* takes place. The registration informs the local LR that a new unit has entered its area. The local LR is responsible for performing the paging for the newly entered mobile unit. The local location area is called the *visitor*

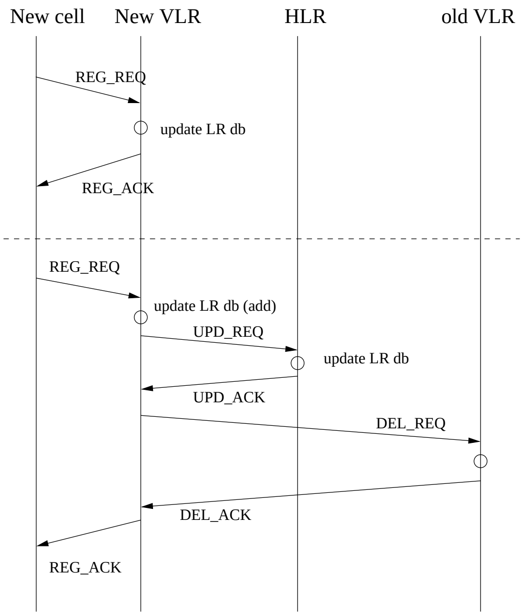


Figure 2: **Registration message sequence in a typical VLR/HLR system.**

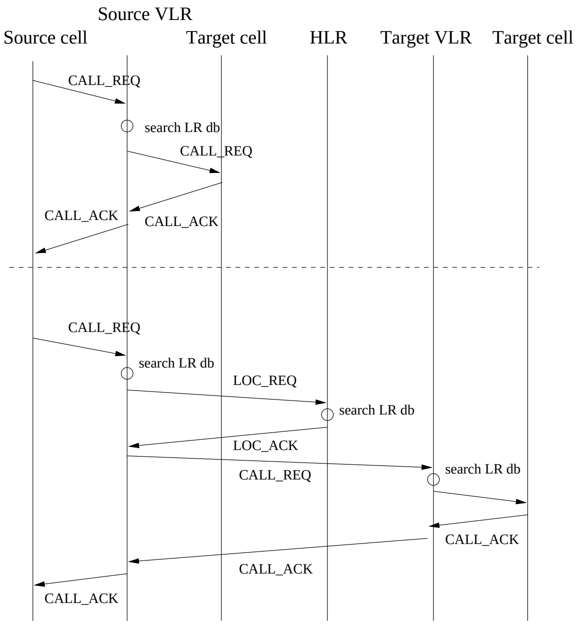


Figure 3: **Call delivery messaging sequence when both parties are in the same Location Area (upper), and when the parties are in separate Location Areas (lower).**

location area (VLA), and the corresponding location register is called the *visitor location register* (VLR). The registration operation also informs the mobile unit's HLR of the VLA which the mobile unit is in. Every time the mobile unit leaves a location area and enters another, a registration is performed so that the associated VLRs as well as HLR are informed of the transition. A registration operation is also performed every time the mobile unit powers up (assuming it powers up within the coverage area of the network).

A mobile unit (caller) that needs to call another unit (callee) sends a call request. The system sends a query to the callee's HLR which knows of the callee's latest registration and gives back the id of the VLR it is registered with. The call is then directed to the VLR of the callee which pages the VLA for the callee. Figures 2 and 3 show the messaging sequence of a typical VLR/HLR architecture for call delivery and registration.

2 Challenges in Location Management

The objective of location management is to maintain correct location information and to have this location information available when mobile data needs to be transfered among mobile units. An efficient location management scheme should satisfy the basic requirements of information *accuracy*, information *availability*, *scalability*, and meet performance requirements like low access *latency* and low messaging/signaling *overhead*.

Information accuracy is essential to allowing the network to route the data to the correct location, i.e. the location of where the data recipient resides. Information accuracy in a hierarchical location management architecture goes hand-in-hand with granularity of location information. To understand this concept, let's consider the hierarchical location management architecture of GSM: when a mobile unit is outside the HLA, its HLR does not know the last reported cell of the mobile unit; it only knows the VLA that it is in. This is a *coarser granularity* of location information. The VLR has a finer granularity in its information about the location of the mobile unit, as it knows the last reported cell; yet this information is not accurate enough for routing the data. Paging needs to be performed in order to find the exact location (cell) of the mobile unit. Only then data can be successfully routed to the mobile unit and back.

The information availability requirement makes sense in a distributed location management architecture. Information that is available by one part of the location management system, should be available, through a protocol, to any other part of the system that requests that location information. The location management should be efficient enough that the discovery of location information does not consume excessive resources and time. This leads to the performance requirements.

The main reason for achieving low latency and overhead in a location management scheme is the need for minimizing the chance for service interruption or call drop. For example, during a hand-over of a mobile unit in a call, the location and routing information has to be updated in a time critical manner, so that the call is not interrupted, and of course incoming calls are re-routed to the new cell. In a circuit-switched network, it means that the switching center has to respond to the change within a minimum time requirement, and in packet-switched network, the routing information has to be updated on the fly.

2.1 The update/search trade-off

The traditional task of keeping the performance of location management within the requirements, is to devise management schemes that distribute the load posed to the location management system, so that overall the LM system meets the requirements.

Search and update pose a load to the network, in terms of messages generated and sent through the network. These basic operations are related in the following formula:

$$L = f_u c_u + f_s c_s \tag{1}$$

which states that the load posed to the network (L) is the number of updates (f_u) into the cost of one update operation (c_u), plus the number of searches (f_s) into the cost of one search operation (c_s). These values may not be true for every given instance of a data connection, but they express the cost of a data connection in an average fashion.

Analyzing the formula given above, we can relate the various factors to the requirements of a location management scheme. f_u affects the accuracy of location information: the more frequent the updates are, the ”fresher” the location information is, therefore more accurate. c_u affects the availability of location information: the more available the location information is made (i.e. more parts of the distributed location database system are updated) the more cost is needed to perform that operation. f_s is mostly dependent on external factors (e.g. human interaction), thus location management schemes do not deal with altering that factor. Nevertheless, depending upon the nature of the application that creates the traffic, *data caching*, either in the form of *lazy caching* [12], or in the form of *eager caching* [22], and *data broadcasting* [3, 36] schemes may reduce f_s, if the application can benefit from such schemes. c_s is affected by both accuracy and availability, it is the remaining cost to obtain an adequately accurate location information, and which is not paid by the updating part of the scheme. We can intuitively perceive c_s as

$$c_s \propto \frac{1}{f_u c_u}$$

A location management scheme usually defines the cost of update and cost of search. As we mentioned previously, cost merely expresses the load of the messages generated on the network to perform the requested operation.

The big challenge in location management research is the *search/update trade-off*: generally, the more effort one puts in updating the location the less effort is needed to be searched, and vice versa. In terms of factors, this means that by decreasing c_s we most likely have to increase c_u; on the other hand, if we decrease c_u we most likely have to increase c_s.

For example, assume that in the VLR/HLR architecture the mobile units inform their location to the LR every time they make a hand-off. This practice eliminates the need for paging. On the other hand, this practice increases the ”updating” process. As a separate example, assume that upon each registration the VLR informs all LRs, not only the HLR, of the registration. This second practice eliminates the need for an HLR query every time a call is placed. These examples demonstrate the update/search trade-off.

Most of the work on location management has been on enhancements and improvements over basic location update and paging schemes, which either reduce the update cost or the search cost. These enhancement techniques include *location caching* [20], *profile replication* [21, 35, 51], and *location prediction* or *selective paging* [1, 15, 31, 33, 49] for search cost reduction. For update cost reduction, some techniques are *forwarding pointers* and *location anchors* [17, 23, 37], *delayed registration* and *predictive registration* [7, 10, 30, 34, 44].

The choice of the right optimizing scheme depends on the value of f_u and f_s. If the frequency of updates is much larger than the frequency of call arrivals, so that $f_u c_u > f_s c_s$, then then we should choose to employ a scheme that decreases update cost ($f_u c_u$) and increases search cost ($f_s c_s$), so that there is an overall reduction. Similarly, we can choose a suitable optimizing scheme when $f_u c_u < f_s c_s$. It is widely known that the ratio $\frac{f_s}{f_u}$ is called *call-to-mobility ratio*. Each of the techniques mentioned in the previous paragraph has a range of CMR that it is suitable for. It is not uncommon for a cellular network to employ various optimizing schemes that are suitable for various CMRs and to choose and activate them according the individual conditions (CMR) of each mobile unit.

3 The Problem of Update Cost Reduction

In order to provide good accuracy and availability of location information, there is need for more frequent updates. Updating more frequently reduces the error in the data accuracy, increases the granularity and helps in the availability.

In addition to the design benefits of frequent updating, there is the technology and consumer impact to the update rates. The bandwidth of each wireless channel has increased from a few Kbps to several tenths of Kbps. This requires higher energy consumption and lower error rates in the channel. In order to compensate for the energy consumption and to achieve lower error rates for the increased bandwidth, the sizes of the cells have been reduced. Most of the urban cellular networks now employ microcells instead of macrocells.

The need for smaller cells is also mandated by the ever-increasing numbers of cellular customers. Current trends in the usage of PCS networks show that the number of subscribers is expected to increase exponentially for the next several years. In [32], Rappaport states that in 2001 there were over 600 million cellular/PCS subscribers and they are expected to reach 2 billion by year 2006. In most countries the yearly subscriber increase is at least 40%. Each cell has a maximum number of channels that it can provide. This poses a limit to the users that can be serviced by a base station. Higher cell density can afford higher population density and in addition it allows higher frequency reuse.

As the cell sizes decrease and subscriber population increases, the frequency of updates (hand-offs and registrations) increase dramatically. The increasing number of subscribers and the increasing number of update operations multiply the effect on the performance of the location database to point that the latter is becoming a bottleneck in location management. The need for efficient and scalable update schemes is greater than ever. This growing rate of updates demands for some control and reduction over the update rates.

3.1 Related work: Overlapping in Cellular Networks

The problem of minimizing (or, at least, reducing) the updates beyond the update/search trade-off hasn't been thoroughly studied by the research community. Nevertheless, a technique known as *location area overlapping* has shown promise for reducing the frequency of updates.

Location area overlapping is a feature that enables location areas to share cells among them. It is not strictly considered an update/search optimization technique, but it has a direct effect on the update and search costs. Overlapping can reduce number of updates [9, 47], and it can also provide smooth updates in the system [28, 29, 50], similar to the reason for which overlapping was introduced at the cell level to enable smooth user hand-offs between cells.

The concept of overlapped location areas was first used in the early 1990's to solve the problem of frequent HLR updates caused by movements on the boundaries of location areas [29]. In particular, overlapped LAs can achieve the following: (a) increase the area handled by each LR, effectively reducing the number of inter-LA hand-offs, and (b) keep the number of users (mobile

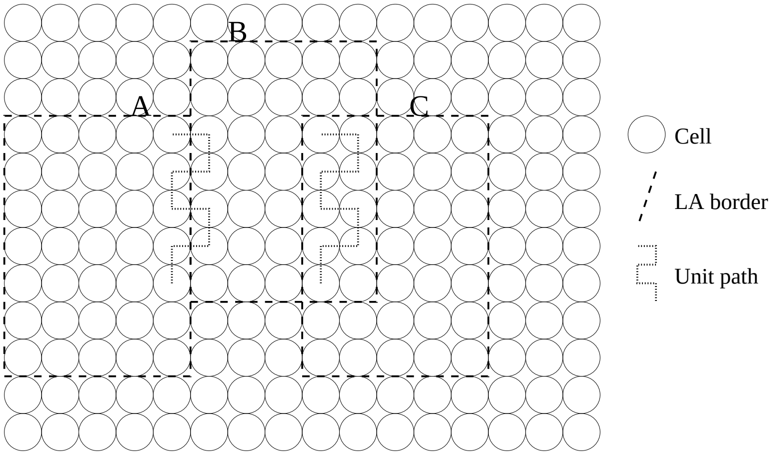


Figure 4: **The ping-pong of a mobile unit between two non-overlapping location areas (A,B) can be contained with overlapping (B,C), thus dramatically reducing the number of registrations.**

hosts) handled by each LR similar to those of non-overlapped LAs. The practicality of overlapping when used in solving mobility-based problems was first pointed out in [29], which proposed two overlapping layers of location areas to reduce location updates. The benefits of overlapping when used in planning personal communication networks have also been given in [28]. Many researchers have found the overlapping of registration or paging areas to be useful and there have been several works that use overlapping ([9, 13, 15, 18, 29, 52, 53]).

Overlapping has the ability to reduce the frequency of updates (namely registrations) in two ways. First way is the elimination of the so-called *ping-pong effect*: as we can see in Figure 4, a mobile unit's path is to oscillate at the border of the two location areas A and B. This path causes a lot of updates. When LAs overlap, like in the example of B and C, the same path can be completely contained in one of the two overlapping location areas. The second way is on a larger scale effect, that includes the elimination of the ping-pong effect as a sub-case, the containment of unit mobility. Larger location areas have higher probability of containing more of a mobile units mobility, thus reducing the number of updates.

Ho and Akyildiz in [18], propose a way to statically determine the overlapping of location areas. A year later, Bejerano and Cidon in [9] proposed a hierarchical, overlaid and overlapping location area (LA) organization of the coverage area in which the communication cost between two units is linearly bound by the geographical (Euclidean) distance of their respective lowest-level location registers. The LAs of one level have twice the radius of the LAs of the level below, which makes the system have a logarithmic number of layers with respect to the size of the coverage area. The scheme performs very well on average, showing optimal average case search and update times, however it performs poorly in the worst case. The scheme uses a default-LA registration policy, that is, a mobile unit in a region has to register with a pre-determined default LA.

3.2 Problem statement and approach

As we have seen in the previous section, the main topic of location management research is providing solutions under the update/search trade-off constraint. Nevertheless, as explained in the motivation at the beginning of this section, the load increase caused by the combined increasing number of subscribers and the increasing number of cells demands for something beyond the algorithms that deal with the trade-off. Using algorithms based on the update/search trade-off to deal with the increasing load of updates would look like hiding the dirt under the carpet.

The impact of the update increase requires a decrease of the $f_u c_u$ component in the equation(1). The update/search trade-off argues that optimizations can only done under the rule that $c_s \propto \frac{1}{f_u c_u}$, which means that reducing $f_u c_u$ increases the search cost (c_s). The problem we are facing is that we need to investigate reduction of $f_u c_u$ without affecting c_s. The goal should be to devise methods, improvements or optimizations to the update cost, as expressed in the load equation (1), without affecting the search cost. A similar research effort was done in optimizing the search cost without affecting the update cost. Paging schemes are known to optimize cost of one of the two operations, specifically the search operation, without unbalancing the update/search trade-off – and the explanation is easy – they manage that by trading-off between average case and worst case in the time axis. Efficient paging schemes assume some kind of knowledge on the mobility and call pattern of the units in order to achieve optimal states. We can use similar assumptions on mobility and call patterns to achieve a reduction on the update cost.

As we will see in the problems that have been studied, knowledge of mobility pattern plays an important role into achieving an optimal state. The goal of my research is **given mobility and call pattern information and knowing the topology of the wired network, devise optimizations to the update cost without affecting the search cost**. A universal algorithm/solution to this problem is not expected to be found, but we can identify aspects of location management, either protocol-wise, or topology-wise, or even decision-wise, in which aspects we can reduce the update cost without affecting the search cost.

We use Equation 1 to give a methodology of our approach to the problem. The load posed to the network by location management scheme is expressed by the equation 1:

$$L = f_u c_u + f_s c_s$$

The study can focus either on keeping the c_u constant (i.e. maintain the update network protocol) and decrease the f_u (i.e. change the update criteria/policy), or on keeping the f_u constant and improving the c_u.

Analyzing the reduction of $f_u c_u$ in the VLR/HLR architecture, we adapt the load formula to the architecture specifics:

$$L = f_u(\alpha c_u^{vlr} + \beta c_u^{hlr}) + f_s c_s$$

where α is the probability that an update (hand-off) will be reported to the VLR, while β is the probability that an update (hand-off) will be reported to the HLR. If $c_u^{hlr} \gg c_u^{vlr}$, then minimizing the registrations is a desirable goal. Generally the inequality is true. Under this assumption, overlapping of location areas, as described previously, increases that α and reduces the β, thus managing to reduce the overall update cost $f_u c_u$.

The study of overlapping and its potential to reduce the update cost has lead my focus on the problem of *dynamic overlapping* of location areas. elaborated in section 4.2. In dynamic overlapping, the extent of overlapping and its layout is determined dynamically by the location registers. In concept, an dynamic/adaptive approach is better than static allocation, as the former can change to accommodate changes in the input traffic and mobility patterns. Dynamic overlapping tries to find the best trade-off between (αc_u^{vlr}) and (βc_u^{hlr}) in order to minimize the overall update cost.

The objective of reducing the frequency of updates (f_u) has lead my focus on the problem of *optimal registration sequence* (ORSP): In an environment of overlapping location areas, there is the element of choice of registration in the overlapping sections, which didn't exist in environments with non overlapping location areas. The problem of best choice then arises. As it is shown in section 4.1, solving the ORS problem we can optimize the number of registrations in an environment of overlapping LAs, without affecting the search cost. ORS problem has some interesting variations that extend beyond the VLR/HLR architecture and they are to be studied in the planned future work.

4 Research Already Accomplished

Two problems are studied with respect to the goal of *reducing the update cost without affecting the search cost*. The problem of *optimal registration sequence* and the problem of *dynamic overlapping of location areas*. The first problem is defined on a per-user basis. The objective is to minimize unnecessary registrations by making a smart choice of registration when the element of choice exists. The results are very encouraging as they show that reduction of updates is possible by simply making smart registration choices. In on-line fashion, it is shown that under some constraints on the topology, a competitive algorithm is achievable.

The second problem is on aggregate basis and tries to find the best size of overlapping location areas with respect to given (aggregate) mobility and call patterns. As the analysis of this problem shows that overlapping has considerable effect on search cost; it increases the average paging cost. Nevertheless, the cost increase is caused by the underlying network topology which is not suitable for overlapping. For this reason, research on suitable network topologies has been planned for study during the dissertation credits.

The sections that follow elaborate on the two problems mentioned above and present the research conducted and the results achieved. So far, theoretical results on both problems have been obtained that support the goal of the research, while future work will be to study the practicality and implementation issues under different system models, one model being cellular networks and the other model being ad-hoc sensor networks, by analysis and simulation.

4.1 The Optimal Registration Sequence Problem

Overlapping of location areas gives the ability of choice of registration at overlapping sections. The problem of best choice has been studied under a uniform cost model: the cost of any registration equals a constant. Under this cost model, optimizing the cost equals minimizing the number of registrations. There are two versions of the problem that have been studied. The off-line version and the on-line version. Background on On-line Computation can be found in [5, 11]. Although only the on-line version is of practical importance, the off-line version was studied for two main reasons: (a) to find the computational limits of this problem and (b) to use the off-line solutions to measure the performance of the on-line solution. The results of this research have been published in [45, 46], and a journal paper is to be submitted to IEEE TMC. With respect to the *update rate minimization* problem, the ORSP manages to reduce updates without affecting the search cost, thus being a step closer to a solution.

4.1.1 Off-line version

The off-line problem can be described as follows: the coverage area is covered by statically overlapping location areas. For simplicity, overlapping sections are called ”regions”. A pre-determined path of a unit is given, which traverses the coverage area through the regions. Each region has a set of available location areas, i.e. those location areas that overlap on it. In each region the mobile unit has to be registered with one of the available location areas. These availability sets change from region to region. The choice of a registration in a region that is not available in the next region will cause a new registration. The Optimal Registration Sequence problem asks for the sequence of registrations of minimal count over the given path.

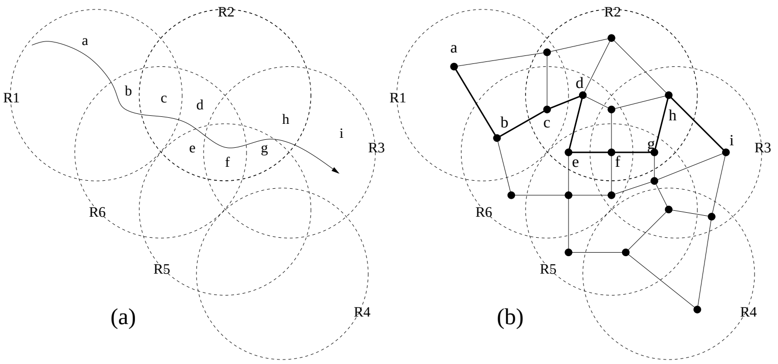


Figure 5: **Demonstration of the ORS problem. (a) Example of the system model and a user traversing the regions formed by the overlapping LAs. (b) The adjacency graph that is built based on the system model. The circles $R1 - R5$ are Location Areas; a–i are the regions that are traversed by the user.**

Figure 5 illustrates the off-line problem. A mobile unit traverses the area that is covered by overlapping location areas (R1–R6). The mobile has to perform some registrations, since it doesn't stay under one location area. Every time the mobile leaves a location area, it must register with another location area that is available in its current region. The mobile starts at region "a" and ends in region "i". One possible registration sequence would be (a,b,c) in R_1 (i.e. mobile remains registered with LA R_1 during its trajectory through regions a, b, and c), then (d,e,f) in R_6, (g,h) in R_2 and lastly (i) in R_3. Another possible registration sequence is (a,b,c) in R_1, (d,e,f,g) in R_2, and (h,i) in R_3. Notice that the number of registrations (including the initial one) is 4 in the first example and 3 in the second example. Is there a better, i.e. shorter, registration sequence than that? For the given example the shortest registration sequence that can exist is with length 3.

The problem can be formally defined like this: Consider that the region path p is partitioned into disjoint sub-paths $p_1, p_2, \ldots, p_m$, $(p_i = g_{i_1} g_{i_2} \ldots g_{i_j})$, such that $p_1 p_2 \ldots p_m = p$. We say that a registration sequence $S = r_1 r_2 \ldots r_m$ is *consistent* with path p *iff* there exists a partitioning $p_1 p_2 \ldots p_m$ of path p such that $\forall g_i \in p_j,\ r_j \in A(g_i)$. Then, the **deterministic optimal registration problem** is: *given a sequence of regions $g_1 g_2 \ldots g_m$ and their availability sets $A(g_1), A(g_2), \ldots, A(g_m)$, find a consistent registration sequence S of minimum length.* A greedy solution exists for this problem. Before we present the algorithm, we need to introduce three concepts:

Availability set $A(g)$ of a region g is the set of location areas that overlap over region g, and of course they are available in g.

Maximal prefix $\pi(r, p)$ of location area r for path p is the maximal prefix of p such that every region in it contains location area r in its availability set.

Optimum prefix $\pi(p)$ of p is the longest of the maximal prefixes.

Optimum prefix function $\sigma(p)$ returns a 2-value vector $(r_\pi, \pi(p))$, where r_π is the location area of the optimum prefix.

The outline of the algorithm is:
Algorithm DORS(r, p):

1. compute $(r, \pi) = \sigma(p)$
2. add area r to the registration sequence S
3. remove the optimum prefix from p: $p := p - \pi$
4. if p is not null, go to step 1

It is not hard to see that the time complexity of the algorithm is linear to the product of the maximum number of registrations α into the length of the path $m = |p|$. For each availability set, the algorithm examines the elements exactly once: for each availability set, it increases the ”length” counter of each location area in the availability set. When it reaches a region in which none of the previously kept location areas is available, it fills up the optimal path up to that point and continues the same way. Since the maximum size of an availability set is α and the number of regions in the path is m, the algorithm takes $O(m\alpha)$ steps to finish.

There are two versions of the algorithms. The algorithm that takes an initial location area r, and is denoted as DORS(r, p), and the indefinite algorithm that doesn't take an initial registration r, and is denoted as DORS(p), and equals the best of all possible DORS(r, p) for that path p. Actually the following theorem has been proved true:

Lemma 1 *For any trajectory* $p = g_o g_1 \ldots g_m$*:* $\forall r \in A(g_o)$ $|DORS(p)| \leq |DORS(r,p)| \leq |DORS(p)| + 1$

As we mentioned above, solving the off-line version sets the foundations for the study of the on-line version of the problem, which follows.

4.1.2 On-line version

The on-line version of the problem states that the future path is not known, but probabilities are given for the next possible transitions. An example graph of the input mobility model is given in Figure 6. The graph of the random walk model will be referred as G_R. The on-line problem asks

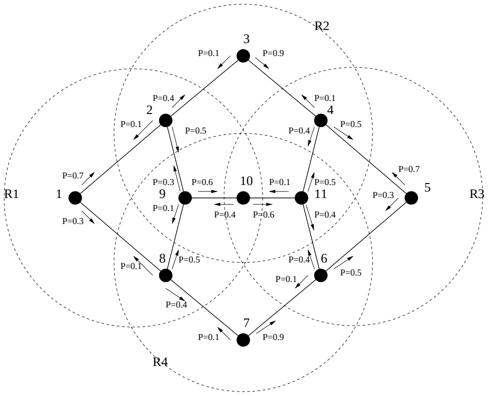


Figure 6: **In the on-line version of the ORS problem, we are given a mobility pattern described by a random-walk model.**

for the choice of registration in a region that would minimize the number of future registrations as the mobile unit continues its journey across the coverage area.

A metric called ”expected minimum number of registrations” was defined and an algorithm that chooses a registration which minimizes that metric was proposed. The metric is the weighted sum of number of registrations along each possible path into the probability of the unit taking that path. A formal definition follows:

The *expected minimum number of registrations for location area* r *with look-ahead depth* d *starting at region* g_o *under model* G_R as the weighted sum of the lengths of the optimal registration sequence of every user path of length d starting at g_o into the probability that the user will take that path:

$$\text{Expct}(G_R, g_o, r, d) = \sum_{p \in \mathcal{P}(g_o, d)} \text{Prob}(p)|\text{DORS}(r, p)|$$

where $\mathcal{P}(g_o, d)$ is the set of all paths in G_R of length d that start at vertex g_o. Based on the metric defined above, we define the **stochastic optimal registration problem:** *given a mobility graph* G_R*, a starting vertex* g_o *in* G_R *and a look-ahead depth* d*, find a location area that yields minimal expected minimum number of registrations as expressed by Expct*(G_R, g_o, r, d).

The algorithm that solves the problem is simply a straightforward computation of the metric:

```
Algorithm $SOR$(Region $g$, integer $d$)
begin
  $Paths$ := Find all paths of length $d$ starting at region $g$;
  for all $r \in \mathcal{A}(g)$ set $opt[r]$ := 0;
  for each location area $r \in \mathcal{A}(g)$ do
      for each path $p_i \in Paths$ do
        $opt[r]$ := $opt[r]$ + Prob($p_i$)|DORS$(r, p_i)$|;
  select next location area $r$ such that $opt[r]$ is minimal
end
```

If $\alpha = \max_{g_i}\{|A(g_i)|\}$, and if the degree of the mobility model graph G_R is δ, Algorithm SOR takes $O(d\delta^d\alpha^3)$ steps to finish.

Based on an expansion of the metric's formula, we proved the following lemma:

Lemma 2 *If $r \in A(g_o)$ then* $\sum_{p\in\mathcal{P}(g_o,d)} (Prob(p)|DORS(p)|) \leq Expct(g_o, r, d) \leq 1 + \sum_{p\in\mathcal{P}(g_o,d)} (Prob(p)|DORS(p)|)$ *where* $\mathcal{P}(g_o, d)$ *is the set of all paths in* G_R *of length* d *that start at vertex* g_o*.*

Which proves our basic result that *under a uniform cost model, once you are registered with a location area you should stay registered as long as it is available.*

In [45] it is shown that if the availability sets have unlimited sizes, then the on-line problem is shown not to have a competitive solution. On the other hand, it is also shown if there is a known finite bound to the size of the availability sets, namely K, then it is shown that there exists a $(K+1)$-competitive algorithm. The two proofs combined show that given a bound K to the availability sets, the best competitive on-line algorithm has a competitive ratio of exactly $(K + 1)$.

4.1.3 other on-line formulations

We have also studied two other formulations of the on-line problem: the first is the on-line ORS problem under the model of soft registrations. The second is the on-line ORS problem model under an oblivious mobility model. The results of this study will be included in the paper to be submitted to IEEE TMC.

The model of soft registrations allows registrations of units between two location areas to have lower cost than a regular registration, when those registrations are performed in a common region of the two location areas. Under the same model, a mobile unit that enters a region where its location area is not available, the unit has to perform a hard registration. Hard registrations are time-critical

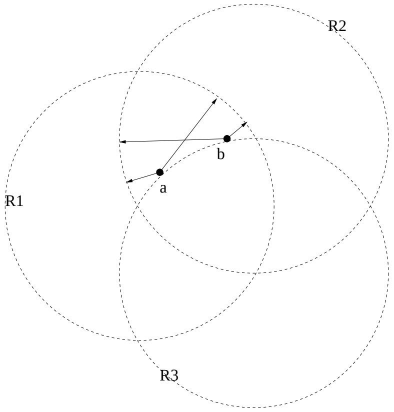


Figure 7: **When the mobility pattern is unknown, then the best strategy is to choose a registration whose borders are the farthest from the point of registration. According to the point of registration, the best choice varies: for point ”a” R1 is the best choice, while for point ”b” R2 is the best choice.**

while soft registrations need not be. This fact explains that the soft registrations can be assigned a cost that is lighter compared to the cost of hard registrations.

The model as studied considers a negligible cost for soft registrations, i.e. zero cost. The problem is therefore translated into minimization of hard registrations. An on-line algorithm has been proposed and its relation to the general ORS algorithm is shown.

Secondly, a model oblivious of the mobility pattern is considered. Instead of the mobility pattern, the location areas are depicted as circles and we are given the radii of each circle. An algorithm is proposed on this model that makes a choice of location area that maximizes the distance of the closest border from the point of registration. Maximizing the distance to the closest border maximizes the ”distance” to the next registration. Figure 7 gives two examples: when a mobile unit exits R3 at position ”a”, the maximum of the closest distances is R1, so the mobile unit should register with R1, while at position ”b”, the best choice is R2.

Summarizing on the ORS problem, under the assumption of uniform cost, it has been shown that it is possible to reduce the number of updates, just by making smart choices in registrations. Solutions are provided based on the knowledge of the input mobility (full/offline, stochastic, and oblivious).

4.2 The Problem of Dynamic Overlapping of Location Areas

As explained above, overlapping shows interesting properties concerning the reduction of update rates. Although a static determination of the LA overlapping is possible, it is known that the mobility

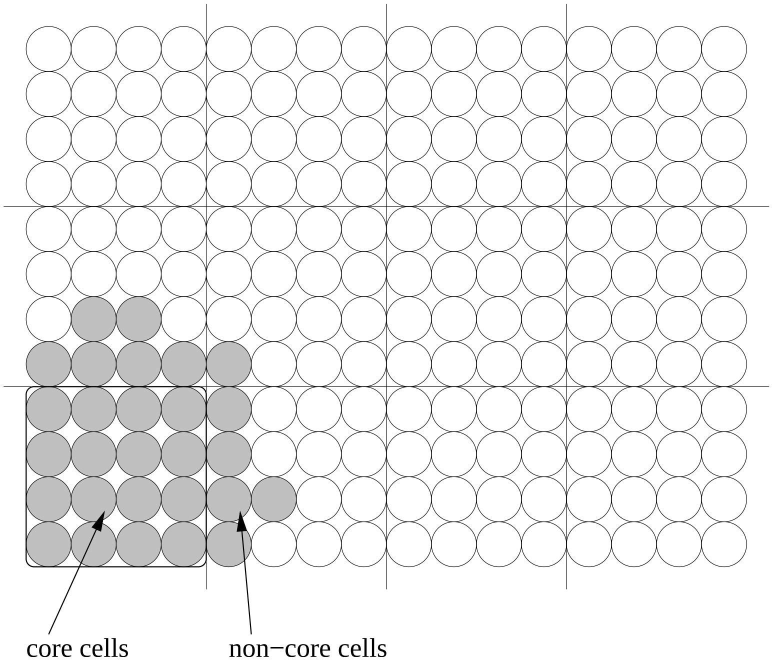


Figure 8: **In dynamic overlapping, location areas are able to include neighboring cells. The lower left LA is shown to have included a few of the neighboring cells. The increase of the LA size allows the LA to accommodate more traffic (reduce registrations) but also increases the paging cost.**

and call patterns change. The need for an adaptive overlapping scheme was realized. In [47] Dr Gupta and I extended the idea of overlapping LAs proposed in [18] and introduced the concept of dynamically re-sizing LAs which adapt to changes in call and mobility traffic. The scheme is based on monitoring the aggregate mobility and call pattern of the users during each *reconfiguration period* and adapting to mobility and call pattern by either expanding or shrinking location areas at the end of each reconfiguration period. The expansion is done by including cells from neighboring location areas in a non-exclusive manner. Shrinking is done by simply removing those initially added cells from the set of cells of the location area. The results of this research have been sent to IEEE TON and they have received a conditional acceptance status.

The inclusion of a cell imposes a trade-off in the costs. The most important impacts of a cell inclusion are that: (a) it converts some of the inter-LA hand-off s (registrations) into simple intra-LA hand-off, therefore it potentially reduces the frequency of registrations; and (b) it increases the paging cost, as now there is one more cell to page for each call that arrives. A detailed analysis of the impact of a cell inclusion follows:

1. **Cost of Intra-LA calls:** By adding the cell a to the LA k, all the calls to MUs of k in cell a

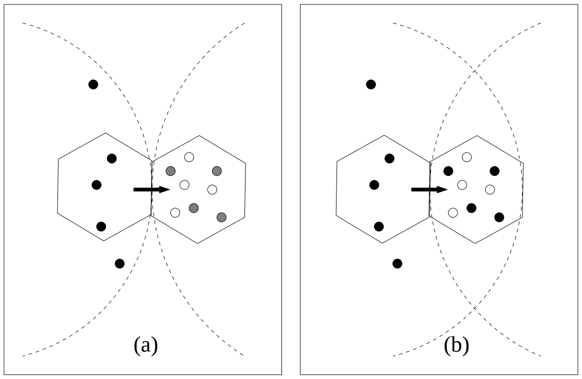


Figure 9: **A neighboring cell contains users that used to be registered to the location area on the left, marked as gray. When the cell is included, about the same amount of users is expected to be registered to the area on the left instead to the area on the right.**

from cells that were already Included to k become intra-LA calls. The same applies for the calls that derive from MUs of k in a to MUs of k in the rest of the cells.

2. **Cost of Inter-LA calls:** Since all calls to MUs of k in cell a become intra-LA calls, we need to subtract them from the inter-LA calls. Furthermore, MUs of k in a make and received inter-LA calls.

3. **HLR updates and searches:** The updates and searches are not expected to change immediately by the inclusion of a cell to the LA. Nevertheless, the reduction of the HLR updates and searches is something we are focusing on, and simulations show this. For the analysis in here, we consider the cost of updates and searches the same.

4. **Cost of Intra-LA hand-offs:** By adding cell a to the LA k, all the hand-offs from the rest of the cells in k by MUs of k become intra-LA hand-offs. The same applies to the hand-offs from cell a to the rest of cells in k for MUs registered with k.

5. **Cost of Inter-LA hand-offs:** For Inter-LA hand-offs, we have to subtract the hand-offs between a and k, because they are intra-LA now. Furthermore, we need to add the inter-LA hand-offs that happen by users of k in a that move out of k.

Predetermining the effects of a cell inclusion is based on the *potentiality* of mobile units, as demonstrated in Figure 9. In Figure 9(a), the left cell is considered to be included and the units marked as gray are the units that were previously registered with the location area on the left. In Figure 9(b), the cell is included and the users that move from the left LA are still registered in the same LA. This means that the effects of a cell inclusion affects only the ”potential” units. In other words, the units that used to be registered in the location area to be expanded.

Each cell base station divides the units in it according to what location area they are registered in. The base station collects rates of hand-offs and calls made by each set of units. For each hand-off that occurs (either incoming or outgoing) the appropriate hand-off counter of the corresponding set is increased. Similarly, there are call counters for each of the unit sets that are updated upon each call. Periodically, these counters are converted into rates by simply dividing the numbers by the elapsed time of the period.

Each cell also keeps a separate set of hand-off and call rates, called the *potential* rates (the rates mentioned in the paragraph above will be referred to as *actual* rates). The previous location area of each unit is known. When a hand-off occurs, there are two counters that are increased: the hand-off counter of the location area that the unit is currently registered to, and the potential hand-off counter of the location area that the unit used to be registered to. When a location area considers the inclusion of a neighboring cell, it uses the potential data of that cell, while for already included cells it uses the actual data.

As shown in [47], the load of a location area is divided into update load and search load posed by units registered to the location area in each cell of the area. The update load is expressed as the product of the hand-off activity into the cost of each hand-off activity. Similarly, the search load is expressed as the product of the call activity into the cost of each call activity. There are basically two types of update cost: the cost of an intra-LA hand-off, where no registration takes place, and the cost of an inter-LA hand-off, where a registration takes place. Similarly there are two types of calls: the calls that are made to units in the same LA (intra-LA calls) and the calls that are made to units outside the LA (inter-LA calls).

The layout of the network topology was chosen so that it reflects the layout of a conventional non-overlapping cellular network: the cells of the coverage area are divided into non-overlapping location areas. For each location area the cells are connected via a star topology, where at the center of the star the mobile switching center (MSC) and the LR reside. The cells of a location area connected directly to the star are the *core* cells of the location area. Non-core cells may be included later in the location areas as they dynamically adjust their size. Due to that topology, signaling to non-core cells has an increased cost. This increased cost further divides the initial two types of costs into 2 subtypes: cost for a unit in a core cell and cost for a unit in a non-core cell.

Formally, we define the basic network parameters as follows:

- δ_{cv}: communication cost between a BS and its MSC.

- δ_{vv}: communication cost between two neighboring MSC.
- ρ: average number of hops (switches/routers) between two MSCs. Hence, the average communication cost between two arbitrary MSC is $\rho\delta_{vv}$.
- γ: average number of hops between an MSC and the HLR of an MU in question. Hence, the average communication cost to an HLR is $\gamma\delta_{vv}$.
- α_v: cost of processing a hand-off/call-delivery at a MSC.

Based on the parameters above, for the a non-overlapping cellular network, the operation costs are as follows:

- Intra-LA hand-off (simple hand-off): $2\delta_{cv} + \alpha_v$
- Inter-LA hand-off (hand-off plus registration): $2\delta_{cv} + 3\alpha_v + 2\delta_{vv} + 2\gamma\delta_{vv}$
- Intra-LA call-delivery: $4\delta_{cv} + \alpha_v$
- $4\delta_{cv} + 3\alpha_v + 2\rho\delta_{vv} + 2\gamma\delta_{vv}$

Nevertheless, in the overlapping part, there are 3 cases of costs:

Case 1: Both MUs belong to the core of an LA

Case 2: One MU belongs to the core and the other belongs to the expanded part of the LA.

Case 3: Both MUs belong to the expanded part of LA.

Each case has separate costs, which are shown in Tables 1, 2, 3.

Based on the costs of each case, the algorithm uses the appropriate formulas to compute the cost contribution of each cell, and the potential cost of each candidate cell and decides whether to expand, contract or to remain. Every time the algorithm decides whether to expand or not, it calculates two sets: the cells for inclusion $I\_Boundary$, which are the external cells touching the border of the location area, and the set of cells for exclusion $E\_Boundary$, which are the included non-core cells at the border of the location area. The outline of the algorithm is:

1. Compute the following two costs for each candidate cell $i \in (I\_Boundary(k) - Core(k)) \cup E\_Boundary(k)$:
    (a) $Cost_{INC}(k, i)$: the portion of $Load(k)$ due to the cost of performing call deliveries and hand-offs for users in the candidate cell i, if cell i is included in the LA k.

Table 1: **Explanations of abbreviations used in subscripts.**

c	cell
v	VLR (or HLR)
t	intra-LA
r	inter-LA
i	incoming (initiated outside LA)
o	outgoing (initiated inside LA)
u	update (move)
s	search (call)
n	non-core
c	core

Table 2: **Costs for different hand-offs/call-deliveries.**

c_{tucc}	$2\delta_{cv} + \alpha_v$
c_{tucn}	$2\delta_{cv} + \alpha_v + 2\delta_{vv}$
c_{tunc}	$2\delta_{cv} + \alpha_v + 2\delta_{vv}$
c_{tunn}	$2\delta_{cv} + \alpha_v + 2\delta_{vv}$
c_{tscc}	$4\delta_{cv} + \alpha_v$
c_{tscn}	$4\delta_{cv} + \alpha_v + 2\delta_{vv}$
c_{tsnc}	$4\delta_{cv} + \alpha_v + 2\delta_{vv}$
c_{tsnn}	$4\delta_{cv} + \alpha_v + 4\delta_{vv}$
c_{rosc}	$2\delta_{cv} + 2\alpha_v + (2\gamma + \rho)\delta_{vv}$
c_{rosn}	$2\delta_{cv} + 2\alpha_v + 2\delta_{vv} + (2\gamma + \rho)\delta_{vv}$
c_{risc}	$2\delta_{cv} + \alpha_v + \rho\delta_{vv}$
c_{risn}	$2\delta_{cv} + \alpha_v + 2\delta_{vv} + \rho\delta_{vv}$
c_{rouc}	$\alpha_v + (2\gamma + 1)\delta_{vv}$
c_{roun}	$\alpha_v + (2\gamma + 1)\delta_{vv}$
c_{riuc}	$2\delta_{cv} + \alpha_v + \delta_{vv}$

   (b) $Cost_{DEC}(k,i)$: the portion of $Load(k)$ due to the cost of performing call deliveries and hand-offs with users in candidate cell i, if cell i is not included in the LA k.

2. Exclude all cell $i \in I\_Boundary(k) - Core(k)$ from LA k if $Cost_{INC}(k,i) > Cost_{DEC}(k,i)$.

3. Remove all cell j from $E\_Boundary(k)$ which are neighboring cell to any cell i removed in Step 2.

4. Include all cell $i \in E\_Boundary(k)$ in LA k if $Cost_{INC}(k,i) < Cost_{DEC}(k,i)$.

It is not hard to express the $Cost_{INC}(k,i) < Cost_{DEC}(k,i)$ in terms of δ_{cv}, δ_{vv}, ρ, γ, α_v and in terms of updates (hand-offs) and searches (calls). In order to see the behavior of the algorithm to the significance of each parameter, we establish a reference graph of $Cost_{INC}(k,i)/Cost_{DEC}(k,i)$ for an ad-hoc set of input values (δ_{cv}, δ_{vv}, ρ, γ, α_v), and then we variate one of them to see the differences in the fraction. Figure 10 shows a reference graph, and Figure 11 shows that the cost ratio drops, especially for low call rates. This means that when in a location area there are a lot of units far away from their HLRs and they cause a lot of inter-LA hand-offs, then it is better for the location area to include the neighboring cells in which the units roam. Similar comparisons can be done for the rest of the parameters. Figures 12 and 13 show the effect of high values of ρ to the

Table 3: **The costs of each case.**

Case #	Intra-LA hand-off	Inter-LA hand-off
	Intra-LA call-delivery	Inter-LA call-delivery
1	c_{tucc}	$c_{rouc} + c_{riuc}$
	c_{tscc}	$c_{rosc} + c_{risc}$
2	c_{tucn} OR c_{tunc}	$c_{roun} + c_{riuc}$
	c_{tscn} OR c_{tsnc}	$c_{risc} + c_{risn}$ OR $c_{rosn} + c_{risc}$
3	c_{tunn}	N/A
	c_{tsnn}	$c_{rosn} + c_{risn}$

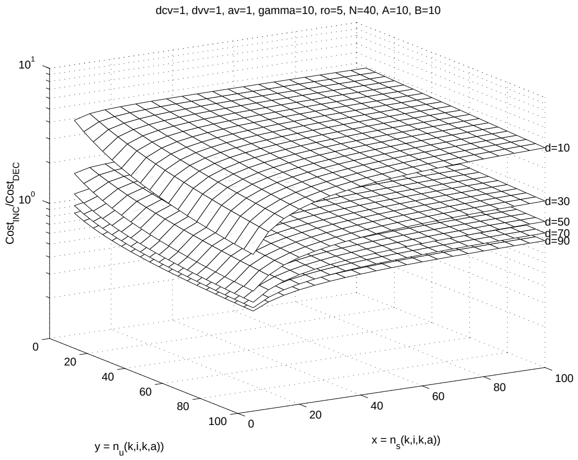
determination of cell inclusion. We can see that the ratio surfaces are flattened, which means that an increase in the signaling cost of long-distance calls does not favor the inclusion of cells.

Summarizing on the problem of Dynamic Overlapping, we have considered the problem under a network topology that causes overlapping to affect the paging cost. Under this realization, the problem becomes a traditional update/search optimization problem. Further research on efficient network topologies promises that dynamic overlapping will escape the update/search trade-off and will further help the minimization of updates, with respect to the *update rate minimization* problem.

5 Future Work

In this section I will discuss the research plans during my dissertation credits. The focus will continue to be distributed among the two topics of research that has been conducted so far (dynamic overlapping and optimal registration), by investigating the practicality by analyzing the problems under different models, and assessing the performance of the schemes by simulation.

In *dynamic overlapping*, there are some aspects of the problem that have not been studied, like efficient network topologies and proactive approach, as explained in subsection 5.1. I will devote time to finish the study of those issues. In the *optimal registration sequence* problem, the on-line version can have some interesting forms when it is applied outside the VLR/HLR architecture. The first form of it is the *optimization of location anchors* and the second is the *registration problem* of mobile units in an ad-hoc sensor network. I will continue with a more detailed description of each topic.



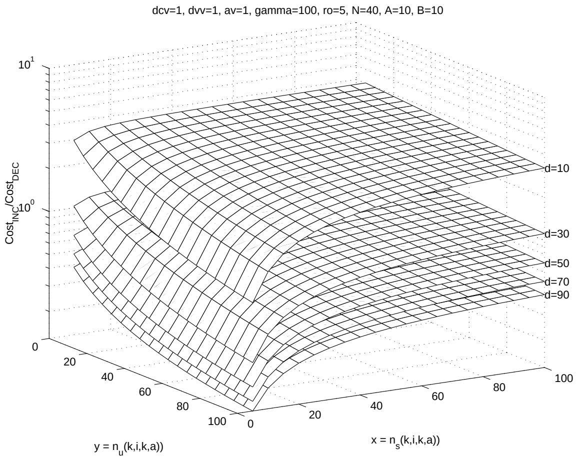
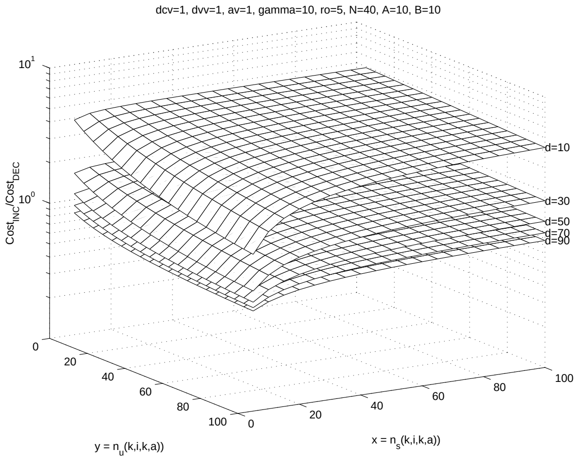


Figure 10: **The ratio $Cost_{INC}/Cost_{DEC}$ as computed analytically for various call and mobility rates. Parameter d denotes the number of times the local traffic is greater in comparison with the remaining traffic; we see that high values of d favor the inclusion of external cells.**

Figure 11: **The ratio $Cost_{INC}/Cost_{DEC}$ as computed analytically for various values of call and mobility rates. In this graph, the value γ is higher than previously, reducing the ratio, thus favoring the inclusion of cells.**



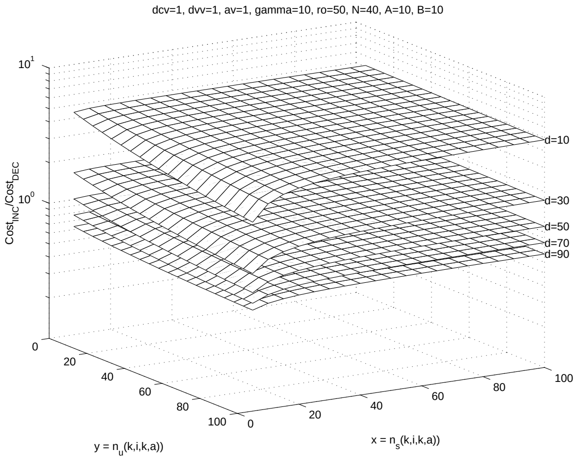


Figure 12: **This figure is identical to Figure 10. It is provided here for comparison with Figure 13.**

Figure 13: **The ratio $Cost_{INC}/Cost_{DEC}$ as computed analytically for various values of call and mobility rates. In this graph, the value γ is higher than previously, and as the graph shows, it does not favor the inclusion of cells.**

5.1 More on dynamic overlapping

The dynamic overlapping strategy, as proposed has some limitations. There are three aspects of this strategy that need improvement: The first is finding *appropriate network topologies* for overlapping. Cellular networks have been planned without LA overlapping in mind. The access cost to a cell is very low for one of the LRs and much higher for all the rest.

The second aspect to be studied is the avoidance of *local optima* in the process of optimal location area size. Mobility and call patterns, in conjunction with the network topology, may produce local optima in the performance of LA layouts. The incremental nature of the inclusion/exclusion algorithm seems to be vulnerable to local optima.

Lastly, the need for a *proactive inclusion/exclusion algorithm* is foreseeable: an algorithm that keeps track of changes and can predict an imminent change, can re-configure the LAs so that they are at optimal layout at the time the change occurs. We will continue with a closer look at each of these issues.

5.1.1 Network topologies for dynamic overlapping

The current network topologies are designed for non-overlapping location areas. Most of the networks are ”star of stars” (in short $star^2$)or generally tree-based topologies, which can show large signaling gaps between two cells. The objective of this topic is to design and propose a topology framework that is suitable for dynamically resizing location areas and does not show a signaling gap between core and non-core cells.

A hierarchical topology, in which the $star^n$ topologies fall under have a very bad bound to the maximum signaling cost between two neighboring cells. In terms of the diameter of a location area (maximum distance between any cells in the location area), the maximum signaling cost can grow in a super-linear fashion.

Restricting the signaling cost to a linear function of the location area diameter, we find out that the ”mesh of stars” topology is very promising in that direction, which is also a scalable network topology. If we want to restrict the signaling cost to a constant, this would imply sort of full-graph network topology, therefore a non-scalable network architecture. Under that principle, we can restrict the maximum size of a location area, then we can design a network topology that resembles blended cliques which provides constant signaling cost to the maximum extent a location area can reach. Perhaps designing a network with a non-constant yet sub-linear signaling can be ”good enough” for a practical solution to the update rate minimization problem.

The results of this research will strengthen the applicability of overlapping in cellular networks and will impact network planning in the future. Most importantly, it will enable overlapping to be a technique that can reduce the update cost without affecting the search cost.

5.1.2 dynamic overlapping and local optima

The algorithm, as it has been developed, can expand or contract a location area only by a zone of 1 cell size. This has problems when there are local optima in the performance of the location area size. An inclusion/exclusion algorithm that does not get trapped at local optima needs to be developed in order to ”replace” the proposed algorithm.

The challenge in this is to find out a method to determine larger parts of the network that are candidates for inclusion. This will allow the inclusion/exclusion algorithm to avoid getting trapped at local optima. Part of the mechanism for determination of the candidate areas is the preservation of potentiality of the mobile units. This can be done if the mobile holds the id of its previous VLA and reports it to each base station it gets handed-off.

An approach to solving this problem would be considering a large domain of possible solutions, i.e. all possible layouts of a location area up to a given extent, assessing their efficiency and selecting the one with the lowest efficiency. Perhaps there is a way to make this solution search faster. This can be viewed as an off-line version of the algorithm that has been proposed.

The results of this research will make the inclusion/exclusion algorithm more robust and suitable for a larger type of mobility and call patterns.

5.1.3 profile-based proactive dynamic overlapping

The inclusion/exclusion algorithm that has been developed is a re-active algorithm; it tries to adapt the location areas to the changing patterns, but after changes have been detected. There is delay between the time a change starts to the time it is detected and finally to the time the reaction has stabilized to the change. This period of ”sub-optimality” can be longer than the period of the optimality that follows.

On the other hand, there are many changes that are predictable, as they are repetitive: morning rushes, evening rushes, weekend getaways etc, are mobility patterns that are predictable and periodical. Similarly, call patterns may show some form of periodicity and predictability.

An algorithm that would keep track of the mobility patterns and the best layout for each phase in a profile can reconfigure the location areas to the best layout at the beginning of each phase. The development of this algorithm will be the topic of this research.

The main challenge of this problem is to identify phases in the mobility patterns, and keep

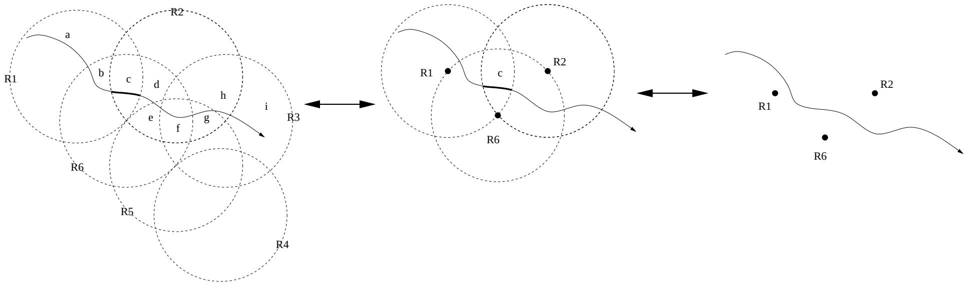


Figure 14: **The uniform cost model with bounded location areas can be transformed into a non-uniform cost model with unbounded location areas.**

then in a profile. Also, a method to identify the beginning of each phase by knowing the time of detection also needs to be developed. Also, an off-line method, the method developed in the previous subsection, to compute the best layout for each mobility pattern.

Wu *et al.* in [51] have worked on a method to identify and represent a single user's mobility pattern. The pattern is then used to develop paging areas for the user for each phase of the mobility pattern. This method could be adapted to identify aggregate mobility patterns, or a similar method could be developed to do the same, which can be then used to select the optimal layout.

This is probably the most complex yet most appealing form of the algorithm. It contains both off-line and on-line methods in it and is supposed to be very robust to the changes of mobility patterns.

5.2 More on optimal registration problem

The optimal registration problem has been well studied under a uniform cost model. In reality though, the cost model is not uniform. Load and signaling costs vary among location registers and an algorithm that considers a non-uniform cost model needs to be developed.

The non-uniform model can be studied in conjunction with what is called *unbounded location areas*. Location areas do not have borders, instead there is a distance-related cost that makes it nearly non-beneficial to service distant mobile units – or is it not? This research is going to study the optimal registration problem in a system model similar to the system model in [7]. Figure 14 shows how the uniform model

The problem seems to be identical to the problem of *optimal distance-based registration policy* or *optimal anchoring* (see anchoring schemes in [17]), where a mobile can choose the location anchor among all location registers available. The knowledge of mobility and of call pattern can be assumed to help choose a better anchor. An off-line version of the algorithm is easy to be

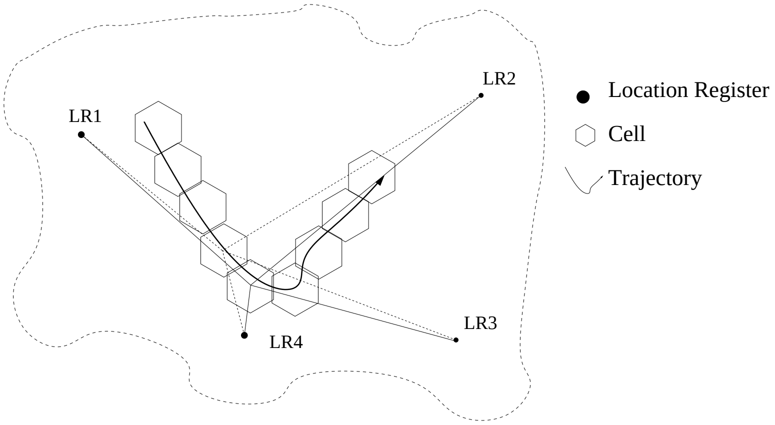


Figure 15: **The problem of best registration is now depended on the route and the distance (cost) of the service of each LR at each point in the coverage area.**

conceived, as presented in Figure 15: the trajectory of the mobile unit is known, as well as the distance-dependent service costs by each LR. An off-line solution can use the given information to find an optimal solution.

The same anchoring problem can be found in ad-hoc sensor networks servicing mobile units. The system model is almost the same, but the arrangement is not planned, and the range of each sensor-cell is unknown. Although the basic challenges are the same for the optimal registration (or optimal anchoring) in sensor networks, there are also the impacts of low battery life of networks, a nature that mandates the use of a *fault-tolerant* registration algorithm

Both aspects of the problem stated above are equally important. Studying the first aspect of the problem, i.e. placing the problem with respect to the distance-based update problem, will give a better theoretical support for the relative importance of this problem to the other optimization schemes. Studying the second aspect, i.e. adaptation to sensor networks, will improve the applicability of the registration algorithm developed in the research.

5.3 Plan of Work

For the topic of *network topology specialized for overlapping*, I will need to perform performance analysis on the various candidate topologies and assess the implementation implications of them. I believe that performing simulation of the a dynamic LA resizing algorithm is of minimal importance. Analysis of the various candidate topologies will take about 2 weeks per topology, depending how promising each topology appears to be.

For the topic of *dealing with local optima* I need to develop an offline overlapping version of the algorithm, similar to the offline algorithm proposed in [18]. This would take about 2 weeks. Analysis and simulation will take another 6-8 weeks, depending on the extent of the simulations required.

For the topic of *pro-active dynamic overlapping* I will need the results of the previous topic. Then I will have to develop an algorithm that identifies and keeps track of the various topologies. This would take about 4-6 weeks, as it requires the study of gathering,organizing and storing of statistical data. Performance analysis and simulation will follow and will take another 4-6 weeks.

For the topic of *optimal dynamic anchoring* I will need about 2-4 weeks to develop and analyze. Simulation, if deemed necessary, will take about 2-4 weeks. For the topic of *optimal registration in ad-hoc sensor networks* I will need about 2 weeks to develop and analyze a fault-tolerant variation of the dynamic anchoring algorithm. Simulation will take about 2-4 weeks.

The total time estimated is 6-8 months. I am expecting to have the work finished by the end of the Fall Semester 2003, and a dissertation ready for defense in December 2003.

References

[1] A. Abutaleb, V. O. K. Li. Paging strategy optimization in personal communication systems. *Wireless Networks*. pp. 195-204, Issue 3, Volume 3, August 1997

[2] A. Abutaleb and V. O. K. Li. Location update optimization in personal communication systems. *Wireless Networks*. pp. 205-216, Issue 3, Volume 3, August 1997

[3] S. Acharya and S. Muthukrishnan. Scheduling on-demand broadcasts: new metrics and algorithms. In *Proc. 4th Annual International Conference on Mobile Computing and Networking (MobiCom98)*, 1998

[4] A. Ahmad. A CDMA Network Architecture Using Optimized Sectoring. *IEEE Transactions on Vehicular Technology*, Volume 51, Number 3, May 2002.

[5] S. Albers and S. Leonardi. Online Algorithms. *ACM Computer Surveys*, September 1999.

[6] B. Awerbuch and D. Peleg. Online Tracking of Mobile Users. *Journal of the Association for Computing Machinery*, pp. 1021-1058, Issue 5, Volume 42, September 1995

[7] A. Bar-Noy, I. Kessler and M. Sidi. Mobile users: To update or not to update? *Wireless Networks*. pp. 175-195, Issue 2, Volume 1, July 1995

[8] Y. Bejerano and I. Cidon. Efficient location management based on moving location areas. In *Proc. 20th Annual Joint Conference of the IEEE Computer and Communications Societies (InfoCom01)*, 2001

[9] Y. Bejerano and I. Cidon. An Efficient Mobility Management Strategy for Personal Communication Systems. In *Proc. 4th Annual International Conference on Mobile Computing and Networking (MobiCom98)*, 1998

[10] A. Bhattacharya and S. K. Das LeZi-update: an information-theoretic approach to track mobile users in PCS Networks. In *Proc. 5th Annual International Conference on Mobile Computing and Networking (MobiCom99)*, 1999

[11] A. Borodin and R. El-Yaniv *Online Computation and Competitive Analysis*. Cambridge University Press, 1998

[12] J. Cai and K. -L. Tan. Energy-efficient selective cache invalidation. *Wireless Networks*. pp. 489-502, Issue 6, Volume 5, December 1999.

[13] C. Castelluccia. Extending mobile IP with adaptive individual paging: a performance analysis. *ACM Mobile Computing and Communications Review*, 2(1):1–13, February 1997.

[14] H. S. M. Coxeter. *Introduction to Geometry*. Second edition. John Wiley & Sons, Inc. 1969.

[15] S. K. Das and S. K. Sen. Adaptive Location Prediction Strategies Based on a Hierarchical Network Model in Cellular Mobile Environment. Proceedings of Second International Mobile Computing Conference (IMCC), March 1996.

[16] P. R. L. Gondim. Genetic Algorithms and the Locaiton Area Partitioning Problem in Cellular Networks. In *Proc. 46th Conference on Vehicular Technology (VTC'96)*, May 1996.

[17] J. S. M. Ho and I. F. Akyildiz. Local anchor scheme for reducing location tracking costs in PCNs In *Proc. 1st International Conference on Mobile Computing and Networking (MobiCom95)*, Berkeley, California, 1995

[18] J. S. M. Ho and I. F. Akyildiz. Dynamic hierarchical database architecture for location management in PCS networks. *IEEE/ACM Transactions on Networking*, Issue 5, Volume 5, October 1997

[19] S. Hurley. Planning Effective Cellular Mobile Radio Networks. *IEEE Transactions on Vehicular Technology*, Volume 51, Number 2, March 2002.

[20] R. Jain, Y. B. Lin, C. N. Lo and S. Mohan. A Caching Strategy to Reduce Network Impacts of PCS. *IEEE Journal on Selected Areas n Communications*, pp.1434-1445, Issue 8, Volume 12, October 1994

[21] J. Jannink, *et al*. Efficient and flexible location management techniques for wireless communication systems. In *Proc. 2rd Annual International Conference on Mobile Computing and Networking (MobiCom96)*, 1996

[22] Z. Jiang and L. Kleinrock. An adaptive network prefetch scheme. *IEEE Journal on Selected Areas in Communications*, pp. 358-368, Issue 3, Volume 16, April 1998.

[23] C. Y. Lee and S. G. Chang. Determination of the Registration Point for Location Update by Dynamic Programming in PCS. *Wireless Networks*. pp. 331-341, Issue 4, Volume 7, August 2001

[24] H. Levy, Z. Naor. Active tracking: locating mobile users in personal communication service networks. *Wireless Networks*. pp. 467-477, Issue 6, Volume 5, December 1999

[25] Q. Li, J. Aslam and D. Rus. Online Power-aware Routing in Wireless Ad-hoc Networks. In *Proc. 7th Annual International Conference on Mobile Computing and Networking (MobiCom01)*, 2001.

[26] K. Lim, Y. -H. Lee. Optimal Partitioning of Heterogeneous Traffic Sources in Mobile Communications Networks. IEEE Transactions on Computers, Volume 46, Number 3, March 1997.

[27] U. Madhow, L. Honig and K. Steiglitz. Optimization of wireless resources for personal communications mobility tracking. *IEEE/ACM Transactions on Networking*, pp. 698-707, Issue 1, Volume 3, December 1995.

[28] J. G. Markoulidakis, G. L. Lyberopoulos, D. F. Tsirkas and E. D. Sykas. Evaluation of Location Area Planning Scenarios in Future Mobile Telecommunication Systems. *Wireless Networks*, pp. 17-29, Issue 1, Volume 1, February 1995.

[29] S. Okasaka, S. Onoe, S. Yasuda, A. Maebara. A new location updating method for digital cellular systems. *41st Annual Conference on Vehicular Technology (VTS)*, May 1991.

[30] A.-C. Pang, Y. -B. Lin and Y. Fang. Implicit deregistration with forced registration for PCS mobility management. *Wireless Networks*. pp. 99-104, Issue 1, Volume 7, January 2001

[31] G. P. Pollini and C. -L. I. A profile-based location strategy and its performance. *IEEE Journal on Selected Areas in Communications*, pp. 1415-1424, Issue 8, Volume 15, October 1997

[32] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Second edition, Prentice-Hall, 2002.

[33] R. Rezaiifar and A.M. Makowski, From optimal search theory to sequential paging in cellular networks. *IEEE Journal on Selected Areas in Communications*, pp. 1253 -1264, Issue 7, Volume 15, Sept. 1997

[34] S. K. Sen, A. Bhattacharya and S. K.Das. A selective location update strategy for PCS users. *Wireless Networks*. pp. 313-326, Issue 5, Volume 5, October 1999

[35] N. Shivakumar and J. Widom. User profile replication for faster location lookup in mobile environments In *Proc. 1st International Conference on Mobile Computing and Networking (MobiCom95)*, Berkeley, California, 1995

[36] C.-J. Su, L. Tassiulas, V. J. Tsotras. Broadcast scheduling for information distribution. *Wireless Networks*. pp. 137-147, Issue 2, Volume 5, March 1999

[37] K.-L. Sue and C.-C. Tseng. One-step pointer forwarding strategy for location tracking in distributed HLR environment. *IEEE Journal on Selected Areas in Communications*, pp. 1455-1466, Issue 8, Volume 15, October 1997

[38] L. Tassiulas and F. M. Anjum. *Efficient Location Tracking of Mobile Nodes for Situation Awareness*. Technical Report. Center for Satellite and Hybrid Communication Networks. University of Maryland at College Park, 1997.

[39] TIA/EIA. Cellular radio-telecommunication intersystem operations. Technical Report IS-41, Revision B, EIA/TIA, 1991.

[40] TIA/EIA Interim Standard 95. Mobile Station–Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System. EIA/TIA, July 1993.

[41] J. Tisal. *GSM Cellular Radio Telephony*. John Wiley & Sons, 1998.

[42] GSM Association, *GPRS Service Description, Stage 2,* GSM 03.60, 1998

[43] C.-K. Toh. The design and implementation of a hybrid handover protocol for multi-media wireless LANs In *Proc. 1st International Conference on Mobile Computing and Networking (MobiCom95),* Berkeley, California, 1995

[44] I.-F. Tsai and R.-H. Jan. The lookahead strategy for distance-based location tracking in wireless cellular networks. *Mobile Computing and Communications Review, Issue 4, Volume 3, October 1999.*

[45] G. Varsamopoulos, G. Konjevod and S. K. S. Gupta. A Theoretical Study of Optimization Techniques Used in Registration Area Based Location Management: Models and Online Algorithms. In *Proc. Discrete Algorithms on Mobile Networks (DIALM'02),* Atlanta, Georgia, September 2002.

[46] G. Varsamopoulos and S. K. S. Gupta. On Finding Optimal Registrations in Presence of Overlapping Registration Areas In *Proc. 5th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM 2001),* Rome, Italy, July 2001.

[47] G. Varsamopoulos and S. K. S. Gupta. On dynamically adapting registration areas to user mobility patterns in PCS networks. In *Proc. Int'l Workshop on Collaboration and Mobile Computing (IWCMC'99),* Aizu, Japan, September 1999.

[48] O.-C. Yue. Design Trade-Offs in Cellular/PCS Systems. *IEEE Communications,* Volume 34, Issue 9, September 1996.

[49] G. Wan, E. Lin. A dynamic paging scheme for wireless communication systems. In *Proc. 3rd Annual International Conference on Mobile Computing and Networking (MobiCom97),* 1997

[50] W. Wang and I. F. akyildiz. Intersystem Location Update and Paging Schemes for Multitier Wireless Networks. In *Proc. 6th Annual International Conference on Mobile Computing and Networking (MobiCom00),* 2000

[51] H.-K. Wu, M.-H. Jin, J.-T. Horng, and C.Y. Ke. Personal Paging Area Design Based on Mobile's Moving Behaviors. In *Proc. IEEE Infocom'01,* Anchorage, Alaska, April, 2001.

[52] H. Xie, S. Tabbane, and D. J. Goodman. Dynamic location area management and performance analysis. IEEE Vehicular Technology Conference, May 1993.

[53] X. Zhang, J. Gomez Castellanos, and A. T. Campbell. P-MIP: paging extensions for mobile IP. *Kluwer Mobile Networks and Applications,* 2(7):127–141, April 2002.