# Middleware for Positioning in Cellular Networks

Israel Martin-Escalona, Francisco Barcelo-Arroyo and Marc Ciurana
*Universitat Politècnica de Catalunya (UPC)*
*Spain*

## 1. Introduction

Middleware is defined in (Mahmoud, 2004) as, "a distributed software layer that sits above the network operating system and below the application layer and abstracts the heterogeneity of the underlying environment". According to this definition, the purpose of a middleware is to isolate technology. This task requires that a new layer responsible for handling data between two systems be defined, so that the technology used in each system is transparent to the other system. This situation is illustrated in Fig. 1, in which a middleware is used for communication between the two systems (i.e., A and B). Accordingly, two interfaces are defined, one for the data exchange between system A and the middleware and another for the link between the middleware and system B. Every time a system begins a communication, the middleware handles the data, so that they are adapted to the technology available in each system.

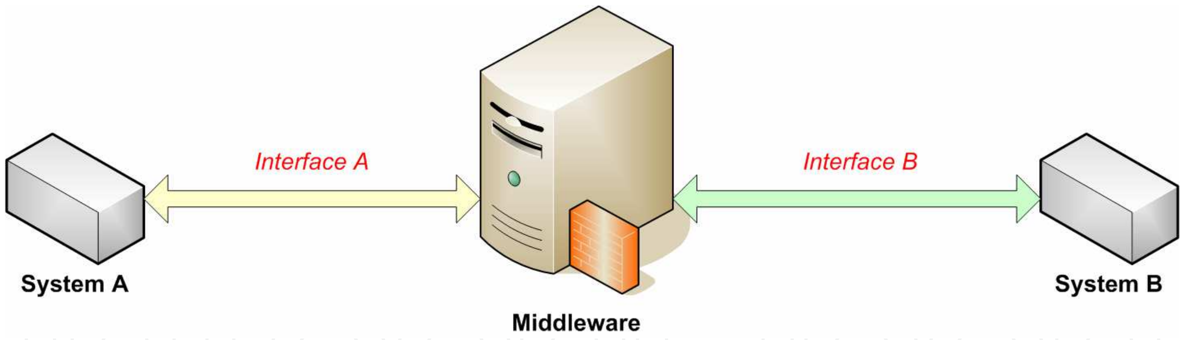


Fig. 1. Middleware interconnecting two systems

The scenario in Fig. 1, in which the middleware splits the communication between two systems, means that several abilities can be expected of a middleware even though these abilities can be grouped into three categories as shown in Fig. 2.As previously mentioned, the main purpose of a middleware is to isolate technology, i.e., making the technology of one system transparent to other systems so that any technology upgrade in one system does not affect any other one. This scenario is especially interesting in the positioning field, in which there are several technologies for fixing the position of customers. Location Based Services (LBS) cannot deal with all those technologies, and the role middleware plays, seems the natural solution to managing the location data. For instance, the Location API for Java 2 Micro Edition (Java Community Process) is a simple solution for writing LBS applications for constrained devices and uses this technology-isolation approach. A distributed approach for technology isolation was proposed by 3GPP in their location platform (3GPP, 2002), which gave the role of middleware to the Gateway Mobile Location Center (GMLC) that acts as mediation for LBS providers.

In the past, the roles of location and LBS providers were combined in a single entity: the network operator. However, the ability of network operators to provide new LBS applications is limited. Several LBS providers appeared in the market, with a large number of applications ready to be released. Although the first approaches for location middleware provided technology isolation, they did not account for quick development, deployment and maintenance of LBS by third parties. This led to middleware proposals that included a framework, i.e., an additional application layer with the purpose of allowing third parties to develop, deploy and maintain their applications in the middleware context. There were several proposals. The Place Lab platform (LaMarca et al., 2005) defines a location middleware based on gathering beacon data for several technologies and storing them in a central database. The main purpose was to provide richer cell-identification (cell-ID) position fixes because dense cells can overlap. Furthermore, the availability of the location solution was improved because several techniques were taken into account. The Place Lab also provides a framework to build clients (i.e., applications) that can exchange data with the platform. A more generic solution is NEXUS (Fritsch & Volz, 2003), which proposes a distributed middleware for a Geographical Information Systems (GIS) platform based on a three-tier architecture: application, federation and service. Application and service can be represented as the location client and server, and the federation is a middle layer that manages the data exchange in a distributed fashion. The main purpose of this solution is the provision of a distributed data model for GIS representation (Augmented World Query Language), which does not depend on the technology used for positioning and allowing spatial overlapping. Other solutions following the same approach are the MiddleWhere (Raganathan et al., 2004), the OpenLS (OGC) and the POLOS platform (Spanoudakis et al., 2004).

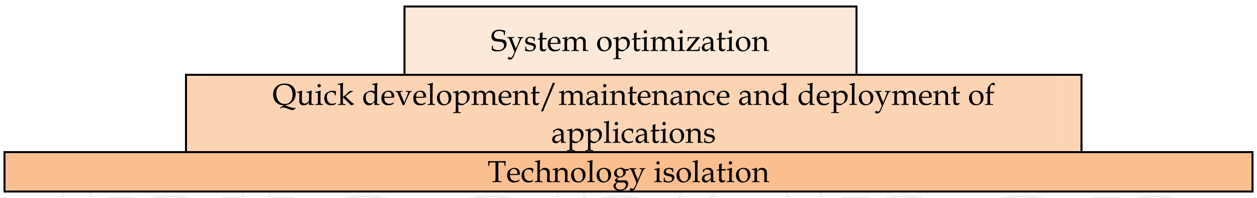


Fig. 2. Main purposes of a middleware

Most of the features associated with the middleware concept are addressed to isolating technology and framework provisioning. However, middleware can also include a functional layer, the purpose of which is to define how the duties are made in order to optimize the system. This optimization can involve several parameters according to the sort of middleware being implemented. In the case of a positioning middleware, optimization can involve economic-aspect enhancement or even tuning parameters directly involved in the operation and maintenance tasks of the communication networks supporting the positioning system. This chapter presents a middleware that has a twofold purpose: 1) optimizing the performance of the system and 2) fulfilling the Quality of Service (QoS) requested. There are few studies of middleware that address optimization. The closest to the proposal presented in this chapter is called TraX (Küpper et al, 2006), which proposes an intermediary device-centric architecture as presented in Fig. 3. The TraX platform distributes the middleware among an LBS provider, a location provider, a content provider and the target device. The *LBS provider* makes the LBS accessible to the external clients (i.e., users) and communicates with *content providers* to add value to the target's position, which is supplied by the *location provider*. The target device is responsible for gathering the measurements necessary to compute the position and, depending on the technique, fixing the position.

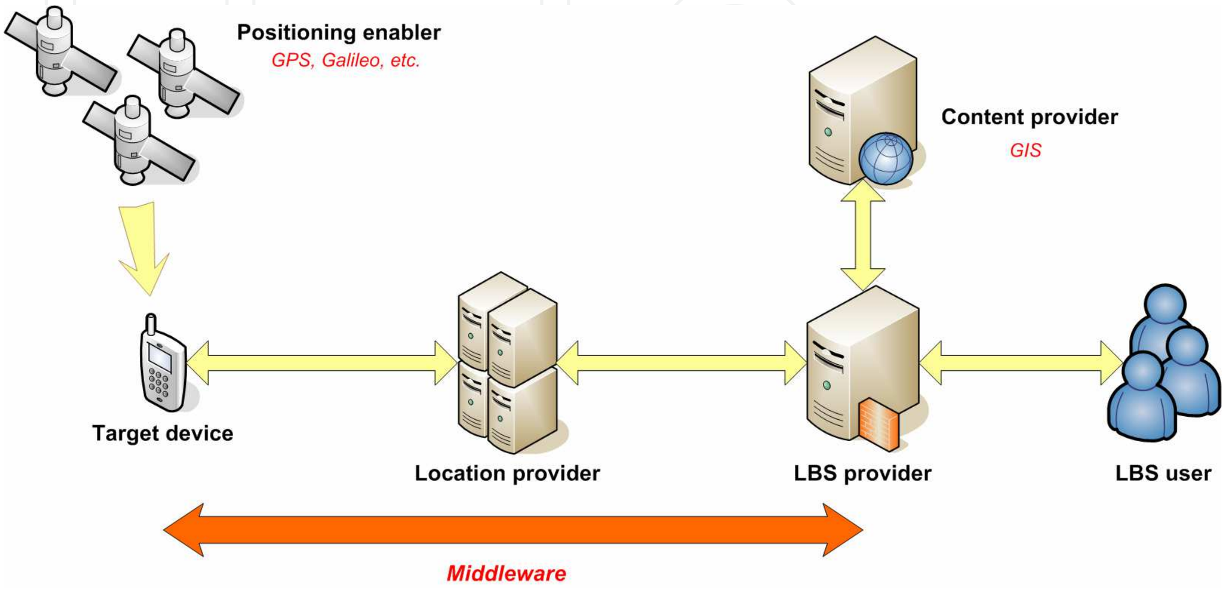


Fig. 3. Intermediary device-centric middleware architecture

The TraX platform consists of three layers: positioning (e.g., A-GPS, WLAN, RFID, OTDOA, etc.), position management (e.g., polling, periodic position updating context awareness, etc.) and application layers (e.g., emergency services, navigation LBS, goods tracking, …). In TraX, the most suitable location technique is selected and then activated to perform the Location Service (LCS) request.

The solution proposed in this document is a middleware for system optimization. It is designed to fulfill the requirements of LBS, and, at the same time, optimize the positioning procedure so that the performance of the whole location solution is improved. Further details of the architecture and performance assessment of this middleware location platform are provided in the following sections.

## 2. Middleware for location cost optimization

### 2.1 Middleware architecture

The resources consumed by location systems generally belong to the underlying networks, on which the location solution runs. It means that LBS share the resources with the regular services provided by the network. Thus, allocating resources for LBS involves reducing the carried traffic for these regular services. The solution proposed in this chapter is a middleware that addresses optimizing the use of resources in location systems. This middleware, which is named MILCO, i.e., Middleware for Location Cost Optimization, has been developed in the frame of (Ministerio de Ciencia y Educacion, 2009). The performance of MILCO consist of analyzing the QoS of the LBS requests, filtering out those location techniques not suitable for a specific request and selecting the optimum technique among the remaining ones according to the resources that are expected they use. MILCO accounts for other factors that constrain the performance location system, such as the location techniques implemented in both, user terminal and core network, the environment where the user is, etc. This approach differs from those taken in standard location middleware solutions because they are usually focused on providing technology-independence or the rapid development of LBS to third-parties, rather than on resource use efficiency.

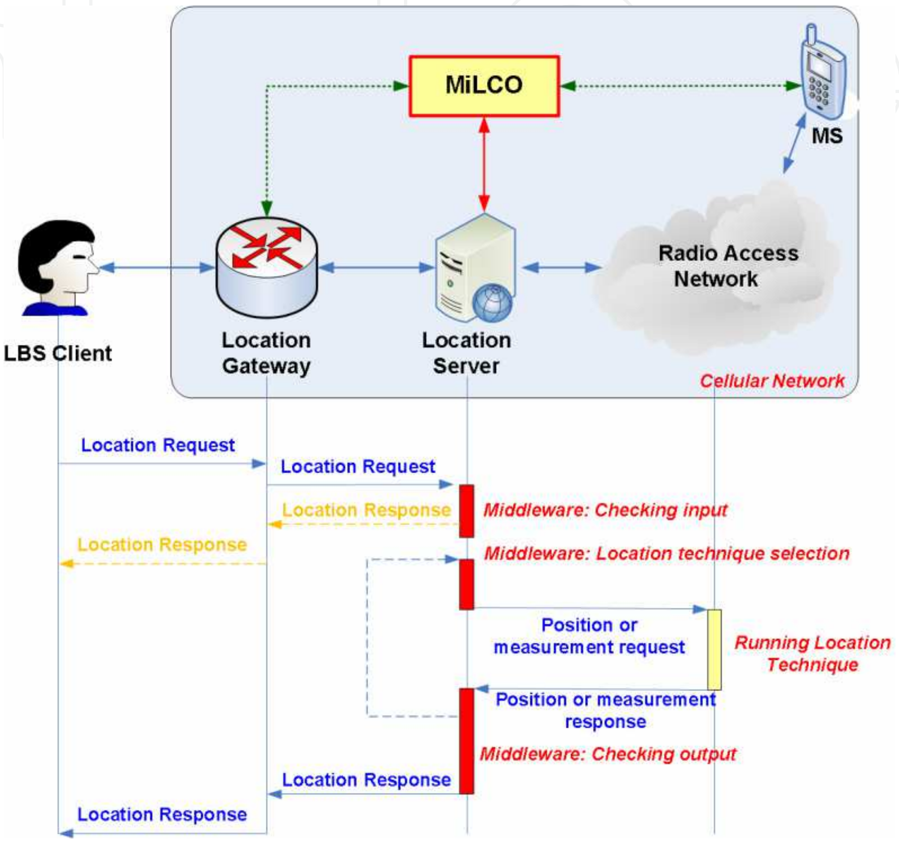


Fig. 4. MILCO system architecture

MILCO is designed to be implemented in terminals, location providers and LBS providers or in a subset of them. However, the usual implementation for MILCO is as a new piece of software inside location providers, e.g., inside Serving Mobile Location Centers (SMLCs) in the case of the ETSI/3GPP notation (3GPP, 2004). Fig. 4 shows a location system architecture that incorporates MILCO in the location provider. Nevertheless, the mobile station (MS) and LBS providers can include certain MILCO functionalities, which are illustrated as green dotted lines in Fig. 4. Under this architecture, each time a LBS request reaches the location system via the location gateway (e.g. GMLC in the case of ETSI/3GPP notation), it is delivered to a location server (e.g. the SMLC in the case of ETSI/3GPP notation). The location server handles the request and forwards it to the MILCO entity, which is placed in the topmost layer of the protocol stack. MILCO then run several input modules to assess whether the request requires executing a location technique. If it is not the case, the input modules will return an estimated position to the LBS client. Otherwise, MILCO selects the optimum location technique for the request, i.e., the one that is expected to provide the requested QoS at the minimum cost. Once it is selected, MILCO uses the network facilities provided by the location server to run the technique and fix the user's position. Finally, if the position fulfills the requested QoS, it will be forwarded to the LBS client. Otherwise, MILCO will iterate again using another location technique.

It must be noted that the MILCO architecture can easily be extended to any cellular system (e.g., 4G PLMNs, WLAN, etc.) as they only need to include MILCO as the topmost application layer in the location stack of any or several devices in the LBS supply chain.

MILCO requires several data in order to carry out its tasks. Most of these data is included in the LBS request or can be easily achieved. These data are detailed below:

- *Location request data.* This information is composed of all the data related to the LCS request, such as the LCS client identifier, the sort of position (i.e. 2D/3D), the periodicity, etc.
- *QoS requirements*. The QoS is required by the LBS client. This QoS can involve several parameters, but it is mainly measured in terms of the minimum accuracy and the maximum delay required by the service as stated in previous chapters.
- *Cell identity*. These data indicate the cell to which the target user is linked. This information is used to compute a coarse position for the target user as well as to optimize the performance of MILCO.
- *Network and handset capabilities*. This information feeds MILCO with the location techniques available in both the network and the target user's handset. MILCO uses these data to filter all the location techniques that are not available in both network and handset simultaneously.

MILCO's procedure is depicted in Fig. 5. This procedure comprises three stages:

- *Pre-filtering* is the process by which any location technique not suitable for the request is filtered out. Location techniques may be marked as unsuitable for three reasons:
  - *Missing technique*, i.e., the location technique is not implemented in either the network or the user terminal.
  - *Poor QoS*, i.e., the location technique is unable, even in the best case, to perform the QoS requested.
  - *Off-line estimation*, i.e., MILCO is able to attend the request and achieve the requested QoS without running any of the location techniques.
- *Selection* is the second stage, and it involves the selection of the best location technique for the request being performed. At this stage, MILCO ranks the remaining set of location techniques (i.e., those available after filtering) according to the optimality for attending the request. This step is achieved by means of a cost function, which quantifies the resource consumption of each of the location techniques. Further details on the cost function are provided in the next section.
- The *Post-processing* stage is responsible for managing the results. The procedure followed in the case of location failures, i.e. QoS offered by the system lower than the requested, is to execute the next location technique in the MILCO's ranking, provided the response-time required has not run out. Notice that this behavior can be modified adding as many output modules as necessary.

### 2.2 Input modules

Input modules are used at the pre-filtering stage to extend the functionalities of MILCO and to improve its performance. The reference implementation for MILCO accounts for two input modules: a location cache and a concurrence manager. These two modules help reduce the number of requests reaching the cost function. As a consequence, the overall

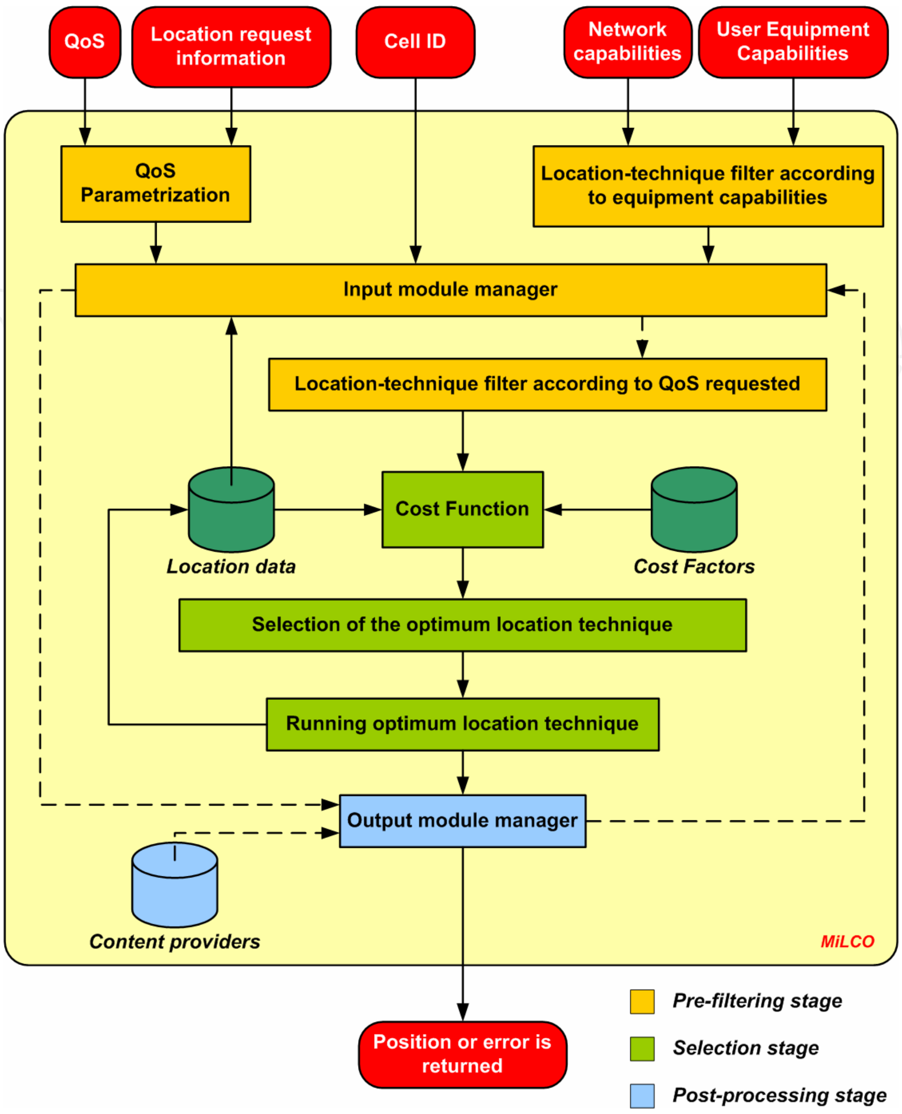


Fig. 5. Block diagram of MILCO

amount of resources used for the location is reduced, since no location technique is run to attend the request, and heavier traffic conditions can be handled.

The location cache saves positions reported in the past to estimate new positions in a near future. The main assumption taken by this module is the user being close enough to those past positions. It means that this module is addressed to users with a slow and pretty constant speed mobility pattern. There are several approaches to verify that the terminal position is close enough to the last stored position (Biswas et al., 2002). The one taken by MILCO consists of building a database with the positions fixed, the QoS achieved and the time at which positions where returned, using this latter information to compute the age of the stored positions and assess if cache module can be run. If the positions stored in the database are close enough to the current time, the cache modules computes the average speed and direction of the user terminal and uses these data to estimate the current position of the mobile station. Subsequently, this estimation may be sufficient, depending on the QoS required, for the task at hand; hence, fewer resources are required for positioning. Accordingly, the performance of the module depends on how old are the positions stored in the database, the mobility pattern of users and the level of QoS requested.

Concurrence aims at avoiding collisions at request level, i.e., a location request is received while another requiring better or equal QoSis still in progress, both asking for the position of the same user. Under such situations, the concurrence manager removes unnecessary traffic in the network, blocking the last received request until the ongoing one finishes. Then, the resulting position is shared by the two requests, even though this may result in a situation where some of the positions returned provide better QoS than necessary. Consequently, the concurrence manager is required to store the input data related to the request (i.e., those data feeding MILCO) to match the QoS obtained by the current technique to the one required by the blocked request. All these data are necessary to handle those cases in which concurrence fails and other input modules or the cost function must be run.

### 2.3 Cost function

The cost function can be considered the MILCO's core It ranks the location techniques suitable for the request (i.e., those available after filtering) according to each technique's resource consumption. Therefore, the more resources the technique consumes the lower it is ranked

The use of resources can be computed based on several factors. All these factors would be subsequentlycombined to obtain an overall cost so that location techniques are ranked. The way in which these factors are combined is defined by the cost function, as shown below:

$$Z_i(t) = f\left(\left\{\alpha_1(t), z_1^i(t)\right\}, \ldots, \left\{\alpha_n(t), z_n^i(t)\right\}\right), \tag{1}$$

where $Z_i(t)$ represents the resources consumed by the $i^{th}$ location technique at a specific time $t$, $f$ stands for a given function, and $a_j$ and $z_j^i(t)$ are the weight and the value of the $j^{th}$ factor applied to the $i^{th}$ location technique, respectively. Several functions $f$ may be used to calculate the use of resources. The reference implementation for MILCO uses a simple additive function with $m$, defined as

$$Z_i(t) = \sum_{j=1}^{m} \alpha_j(t) z_j^i(t). \tag{2}$$

It must be noted that Equation (2) is a first approach to the cost function. It has been formulated on the premise of simplicity and its main purpose is to evaluate the performance of MILCO under low-requirement conditions. Better results could be expected when using more complex functions, but the impact of such complexity on the response time of location requests needs to be quantified and could involve a serious constraint. Furthermore, the actual response time would depend on the hardware and software implementation, which is beyond the scope of this chapter.

### 2.4 Output modules

Output modules are responsible for managing the result of the positioning. The purpose of output modules is twofold: to help recover from location errors and to optimize the computed position. The basic output module deals with location errors and its performance consists in retrying the MILCO procedure as long as it is expected to conclude before reaching the QoS-imposed deadline.

Additional output modules are expected to work with MILCO, such as those related to content providers, which can greatly enhance the QoS of the position reported especially in terms of accuracy.

## 4. Performance assessment

The middleware has been analyzed through simulation. The simulator wraps the simulation area to minimize the impact of the edge effects on the results. The simulation area is turned into a torus (Zander & Kim, 2001) thus becoming a virtually infinite surface with regard to mobility and propagation patterns. This tool is used in upcoming sections to evaluate the middleware under several architectures, networks, location techniques and scenarios.

### 4.1 Network-based implementation

This section explores the performance of the middleware when it is implemented in the core elements of a UMTS network.

### 4.1.1 Cost factors

4.1.1.1 Signaling volume

This cost factor accounts for the amount of information exchanged by each technique. This factor is aimed at favoring lighter techniques, i.e., those requiring less traffic on the network to compute the target position.

In the computation of the signaling volume, the following assumptions are made:

- Only the topmost protocol in the stack (e.g. RANAP, NBAP, etc.) is taken into account.
- A-GPS does not include acquisition assistance information.
- OTDOA and A-GPS can be run with and without assistance data.
- A-GPS running without assistance data means not including the *Almanac* information.
- Hybrid OTDOA/A-GPS includes acquisition assistance information.

Table 1 summarizes the quantification of the signaling volume cost factor for the location techniques allowed by 3GPP in UMTS networks $N_{NB}$ and $N_{SAT}$ in Table 1 stand for the amount of Node-B and satellites involved in the positioning, respectively.

| **Technique** | ***Assistance*** | ***Cost*** |
|---|---|---|
| Cell-ID | No | 0 |
| OTDOA | Yes | $375+134 \cdot N_{NB}$ |
| OTDOA | No | 268 |
| A-GPS | Yes | $473+1199 \cdot N_{SAT}$ |
| A-GPS | No | $461+647 \cdot N_{SAT}$ |
| Hybrid | Yes | $653+134 \cdot N_{NB} + 1254 \cdot N_{SAT}$ |

Table 1. Quantification of the signaling volume

4.1.1.2 Use of wideband interfaces

This cost factor favors those techniques that use wideband channels. Accordingly, it favors those techniques operating in the core network (i.e., network-based techniques). The cost associated with this factor is computed as

$$z_1 = \sum_i r_i^{-1} \quad [ns/bit], \tag{3}$$

where $r$ stands for the throughput of a given channel $i$ and $z_1$ accounts for the cost of all the channels involved in the location process. The Cell-ID is assumed to be delivered to MILCO, and hence, the cost for this factor is 0. On the other hand, the other techniques (i.e. OTDOA, A-GPS and hybrid) are mobile-based and involve the same amount of messages and channels. Under the assumption of $I_{ub}$ and $U_u$ channels having a throughput of 155 Mbps and 384 Kbps respectively, $z_1$ for mobile-based techniques is

$$z_1 = 2\left(\frac{1}{155Mbps} + \frac{1}{384Kbps}\right)10^9 \quad [ns/bit], \tag{4}$$

4.1.1.3 Energy consumption

The last cost factor proposed for UMTS networks accounts for the amount of energy required by each technique to fix the position. This factor aims to maximize the lifetime of the terminal. Power consumption largely depends on the user terminal performance. Here, a simple approach for quantifying power consumption is proposed, which is based on the amount of sources involved in the positioning. The cost of this factor for the location techniques in UMTS is summarized in the Table 2. It must be highlighted that this approach is meant to qualitatively compare the battery consumption of the various techniques, not to set up differences of actual consumptions.

| *Technique* | *Cost* |
|---|---:|
| Cell-ID | 0 |
| OTDOA | $N_{NB}$ |
| A-GPS | $N_{SAT}$ |
| Hybrid | $N_{NB} + N_{SAT}$ |

Table 2. Quantification of the energy consumed by each location technique

#### 4.1.2 Scenarios simulated

The first scenario in which MILCO is evaluated corresponds to a UMTS network (Martin-Escalona & Barcelo-Arroyo, 2006). The call admission control (CAC) used in the simulator was proposed in (Capone & Redana, 2001) and it is based on the impact of new users on the Signal to Interference Ratio (SIR) of ongoing services. It accepts new users whenever the actual SIR ($SIR_2$) of all of ongoing calls in the cell does not drop below the target SIR by more than 1 dB. Otherwise, the service request is blocked.

The power control algorithm was borrowed from (Nuyami, Lagrange, & Godlewski, 2002). Its performance is illustrated in Fig. 6, where $P_{tx}$ and $P_{rx}$ stand for the signal strength transmitted by the mobile station and received in serving node-B, respectively. The algorithm checks whether the transmitting power of the MS should be increased or decreased $\Delta$ dB according to the target SIR and sensibility measured in serving node-B. Table 3 shows the values used in the simulator for all the parameters required by the power control algorithm.

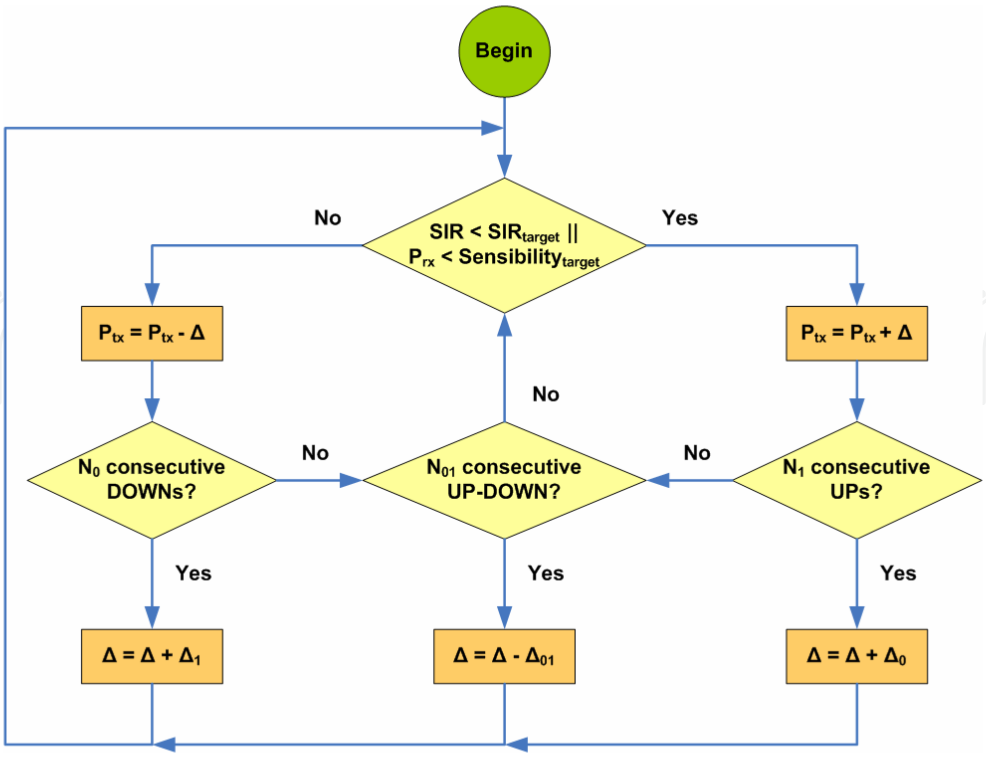


Fig. 6. Power control algorithm

A basic scenario simulates several location loads ranging from 0.01 to 1 request per second. This basic scenario was composed of 100 Node Bs (NBs), which were uniformly placed in a square-shaped simulation area. Each node-B, which involves a cell with a theoretical coverage of 1135-meters, is placed in the center of a square-shaped building. Fig. 7 shows the simulation layout. It must be noted that an important share of the whole area is an overlapping region, i.e., covered by more than one Node-B. This feature puts the simulation closer to reality and at the same time allows OTDOA, which is not possible in areas covered by only one or two Node-Bs.

| *Parameter* | *Value* |
|---|---|
| $\eta_0 = \eta_1 = \eta_{01}$ | 2 |
| $\Delta_0 = \Delta_1 = \Delta_{01}$ | 10 dB |
| Δ (maximum) | 10 dB |
| Δ (minimum) | -20 dB |
| Δ (initial) | 0 dB |
| Power updates between movements | 20 |

Table 3. Parameters of the power control algorithm

Buildings simulate indoor conditions and as consequence, the signal reception inside them is limited. Users move freely within the simulation boundaries and are able to enter the buildings. It must be noted that MILCO makes decisions according to the location request features, the ultimate target of which is a specific mobile station. Consequently, no matter how many users are in the network to carry out the performance assessment. The mobility pattern follows a random-walk approach (Atsan & Özkasap, 2006), in which the user's speed is updated once per second and velocity in both directions, *x* and *y*, are modeled as normal random variables. Pedestrian users are taken into account and therefore, a mean and a standard deviation of 0.6 m/s and 0.18 m/s respectively are set for the user's speed random variables.

The propagation pattern is based on the Okumura-Hata model. According to (Holma & Toskala, 2000), the path-loss slope and zero-meter losses for the pretended scenario were set to 4 and 23dB, respectively. The SIR is calculated according to (3GPP, 2004), which accounts for a spreading factor of 10 dB and an orthogonality factor of 0.4, respectively. Handoffs are requested each time the received power or SIR in a Node B or MS fall below a given threshold, which is known as the handoff threshold. The handoff request is held until either a new channel becomes free and the handoff is then achieved or the SIR or the received power falls below the sensitivity level for more than 15 seconds, which produces a handoff failure and the service disruption. Successful handoffs drop all ongoing location requests carried by the mobile station and unsuccessful handoffs shut down the user terminal for a mean exponential time of 5 seconds. The main propagation pattern parameters have been taken from (Holma & Toskala, 2000) and (3GPP, 2004) and are displayed in Table 4.

| ***Parameter*** | ***Value*** |
|---|---:|
| Minimum SIR | -9 dB |
| Sensitivity of the stations | -109.2 dBm |
| Maximum MS transmission power | 21 dBm |
| Minimum MS transmission power | -44 dBm |
| Node B transmission power | 43 dBm |
| Handoff threshold for received power | -106.2 dBm |
| Handoff threshold for the SIR at reception | $SIR_{min}$ - 6 dB |

Table 4. Propagation pattern parameters

The cell-ID, OTDOA and A-GPS location techniques were taken into account, in addition to a hybrid tight-synchronized OTDOA/A-GPS location technique (Barcelo & Martin-Escalona, 2004). The QoS provided by such techniques, in terms of the expected accuracy and response time, is shown in Table 5, where the *mean* and the *std* stands for the average and standard deviation respectively and the *range* indicates the set of values that the variable may take. The availability of the OTDOA depends on the radio propagation pattern and it is computed on execution time depending on the received power and the SIR. In the case of satellite-based techniques, availability is accounted for differently. The default number of satellites at a sight is set to 5. This number drops to a uniformly distributed value from 0 to 2 satellites inside buildings. It must be noted that the QoS provided by the coupling technique is worse than that achieved by the A-GPS as standalone. This result is due to the greater availability of the hybrid technique, which is favored instead of the accuracy. Furthermore, the lifecycle of the assistance data for the OTDOA and A-GPS is set to 30 seconds, i.e., the assistance information expires 30 seconds after it has been received.

Four LBS generate requests for the station (Martin-Escalona & Barcelo-Arroyo, 2007): emergency, tracking, push and tracing. Table 6 shows the QoS requested by these services and their cadence, i.e., the time between consecutive requests. This later is exponentially distributed in all services. Tracing service differs from the rest in the fact request are received as a burst, i.e., each LBS request involve several LCS requests. The number of LCS requests in the burst is uniformly distributed from 1 to 5, each of the requests separated 20

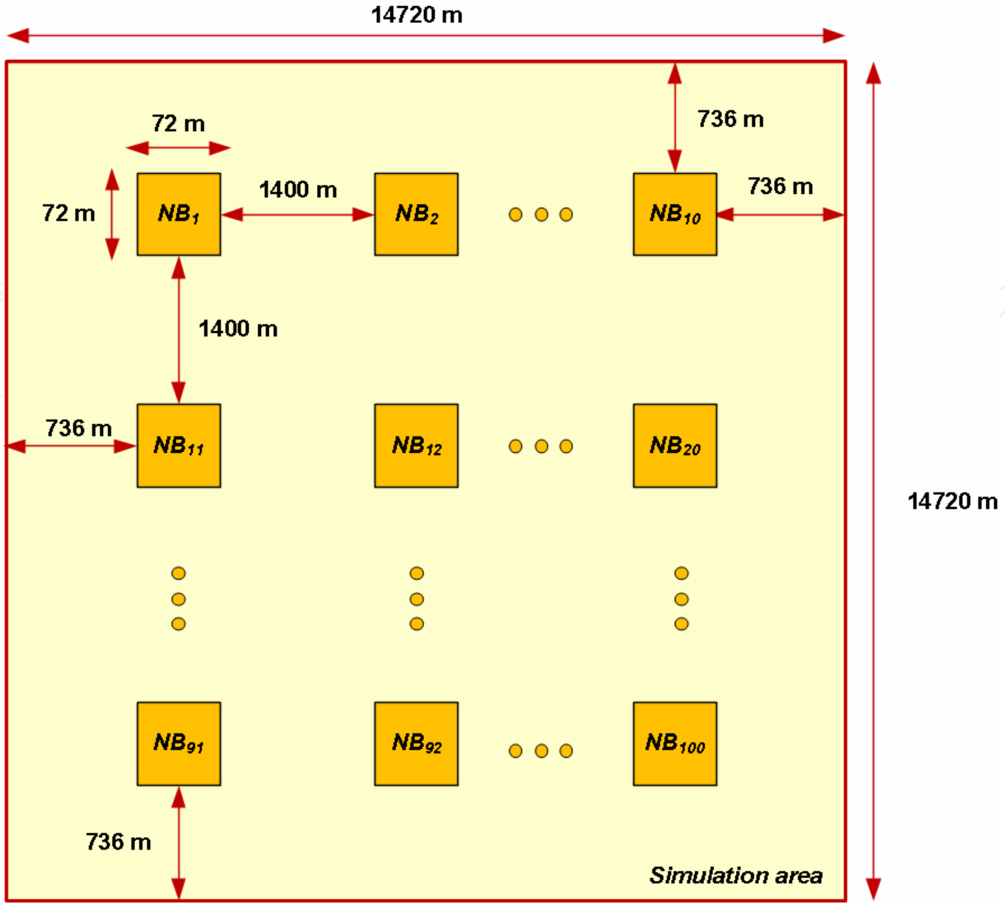


Fig. 7. Simulation layout

| | Accuracy (meters) | | | | Delay (seconds) | |
|---|---|---|---|---|---|---|
| | *Distribution* | *Mean* | *Std* | *Range* | *Distribution* | *Mean* |
| Cell-ID | Deterministic | 1135 | 0 | 1135 | Deterministic | 0 |
| OTDOA | Uniform | 100 | 28.7 | [50,150] | Exponential | 7 |
| A-GPS | Gaussian | 3 | 0.9 | [0,+∞) | Exponential | 11 |
| Hybrid | Gaussian | 50 | 15 | [0,+∞) | Exponential | 27 |

Table 5. QoS achieved by the location techniques

seconds. Not satisfying either the accuracy or the delay involves not fulfilling the QoS requested. Other QoS approaches are allowed, with more parameters and different constraints, but the most restrictive definition (according to 3GPP) is used in this performance assessment.

With respect to the input modules, location cache stores the positions for 2 seconds and then they are removed from the database. The maximum value of a weighted factor was set to 1, which means that the importance of all the cost factors is the same. The maximum value of the cost function is then 3. Weights for the cost factors are assumed to be deterministic and are computed according to that equality assumption. Tuning the weights of the factors is beyond the scope of this work because it is assumed that the setting of these weights would be a task for network operators, thus allowing them to focus their attention on the factors they consider more important at the time.

| *Service* | *Average time between requests* | *Accuracy* | *Response time* |
|---|---:|---:|---:|
| Emergency | 30 min | 50 m | 10 s |
| Tracking | 2 min | 150 m | 15 s |
| Push service | 300 min | 1500 m | 15 s |
| Tracing | 10 min | 50 m | 15 s |

Table 6. Main parameters for services simulated

### 4.1.3 Simulation results

The performance of plain MILCO systems, i.e., systems based only on the cost function and that discard all the input modules, must be analyzed first. Table 7 shows the percentage of successful LCS, the average number of location techniques used in successful LBS (i.e., the requested QoS was finally delivered) and the cost of delivering the LBS. The latter applies not only to those LCS successfully attended, but also accounts for all the LCS run until the QoS requested for the LBS is achieved. Thus, the cost per LBS can exceed the maximum per LCS, i.e., 3. The results in Table 7 correspond to the scenario based on the data in Table 6. Figures for scenarios with a heavier load are not included since they are statistically the same in all the scenarios (i.e., they are not sensitive to the load). Location techniques used as standalone are included for the sake of comparison.

| *Location Technique* | *Average number of techniques* | *Percentage of successful LCS* | *Overall cost* |
|---|---:|---:|---:|
| MILCO | 1.36 | 64.84 % | 2.06 |
| CI | 1.00 | 00.39 % | 0.00 |
| OTDOA | 1.58 | 52.17 % | 3.02 |
| A-GPS | 1.00 | 64.11 % | 2.70 |
| HYBRID | 1.04 | 16.01 % | 2.81 |

Table 7. Performance of MILCO based on the cost function

According to data in Table 7, MILCO achieves the best performance in terms of successful LBS, with figures very close to those achieved by A-GPS. Statistically, it can be stated that there are no differences between them. However, MILCO provides all these LBS with the lowest cost. The performance of MILCO is noticeable better if compared with the OTDOA, both in terms of technique executions and cost. It must be noted that MILCO runs more than one technique per LBS to achieve these figures. However, the cost function compensates this increase in the amount of techniques run does not impact the overall cost because *cheaper* techniques are run first. The poor availability and high cost of the hybrid technique constrains its results when used as standalone. Finally, Cell-ID is the more available and least costly technique, but it yields the least successful LBS rate. According to the results, Cell-ID and hybrid solutions are not suitable for being used as standalone; OTDOA and A-GPS can be understood as a trade-off solution, while MILCO provides the best results.

Performance was expected to be improved by input modules. Hereafter, all the results account for the cost function, and the location cache and the concurrence manager input modules are already enabled. Fig. 8 displays the evolution of the successful LCS requests with the load. The request rates in Fig. 8 start from the rate of services in Table 6 up to 100 times these rates. Therefore, simulations ranging from 0.01 requests per second (light-load profile) to 1.45 requests per second (heavy-load profile) were performed, which is assumed to be is sufficient for assess MILCO under the most demanding applications (e.g., tracking, tracing, etc.). Fig. 8 shows that at medium/low request rates (e.g., 0.05 requests per second), MILCO gives a successful LCS rate of around 65%. These poor figures are due to the QoS definition used in this performance assessment: an LCS is successful only if the accuracy and response time requirements are met. Furthermore, it must be noted that the higher the load, the higher the successful LCS rate. This behavior is due to the fact that the cache and concurrence modules are more likely to be used when less time is spent between consecutive location requests, i.e., more intense is the location traffic. This result proves that the scalability of the proposed approach is guaranteed. In the case of heavier loads, input modules enhance the percentage of successful requests.

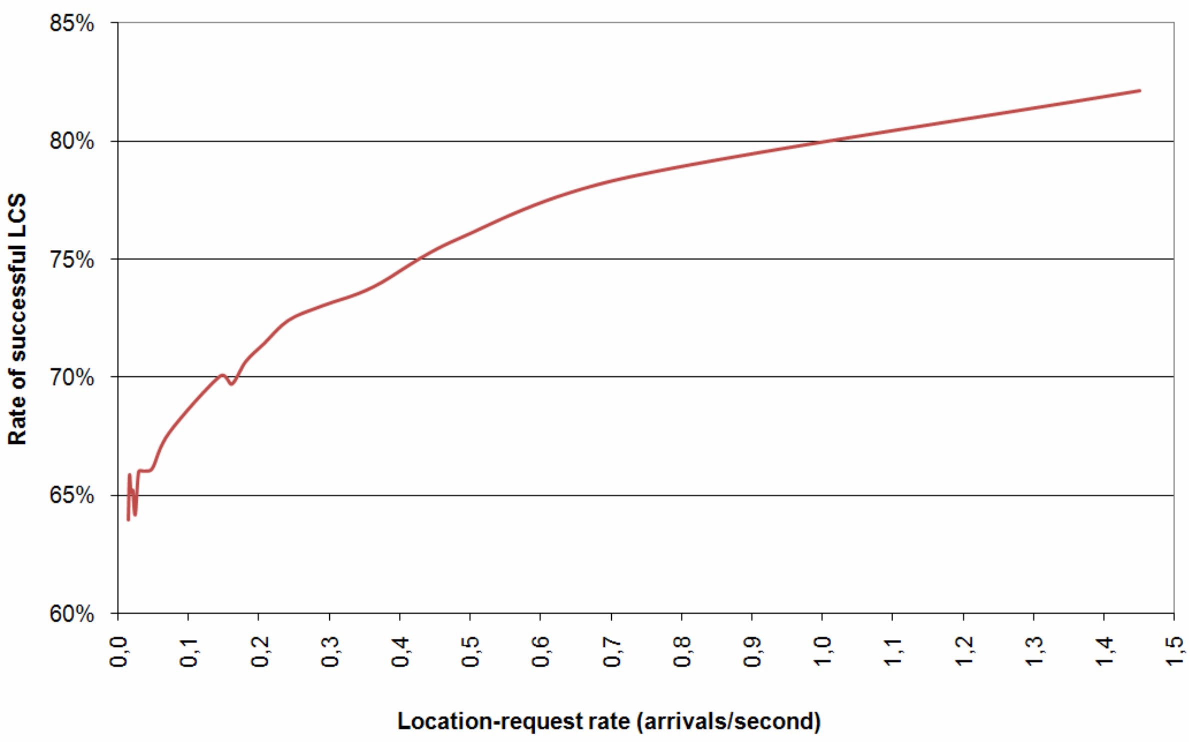


Fig. 8. Evolution of LCS successfully attended

Reducing the use of resources is another strong point of MILCO. Fig. 9 shows the average resources consumed by LBS successfully performed by MILCO. The maximum resources consumed by a single technique is set to 3 under the assumption that all cost factors are weighted the same. This cost is achieved by the hybrid approach, which usually consumes more resources to fix a position. This threshold is depicted in Fig. 9 with a green line. The resource consumption for successful LBS is always below the threshold. Furthermore, the consumption of resources drops as the load increases. This improvement is due to the increasing use of input modules (i.e., the location cache and concurrence manager) because these modules deal with location requests at no cost. Consequently, MILCO is a good approach for reducing the consumption of resources in location systems.

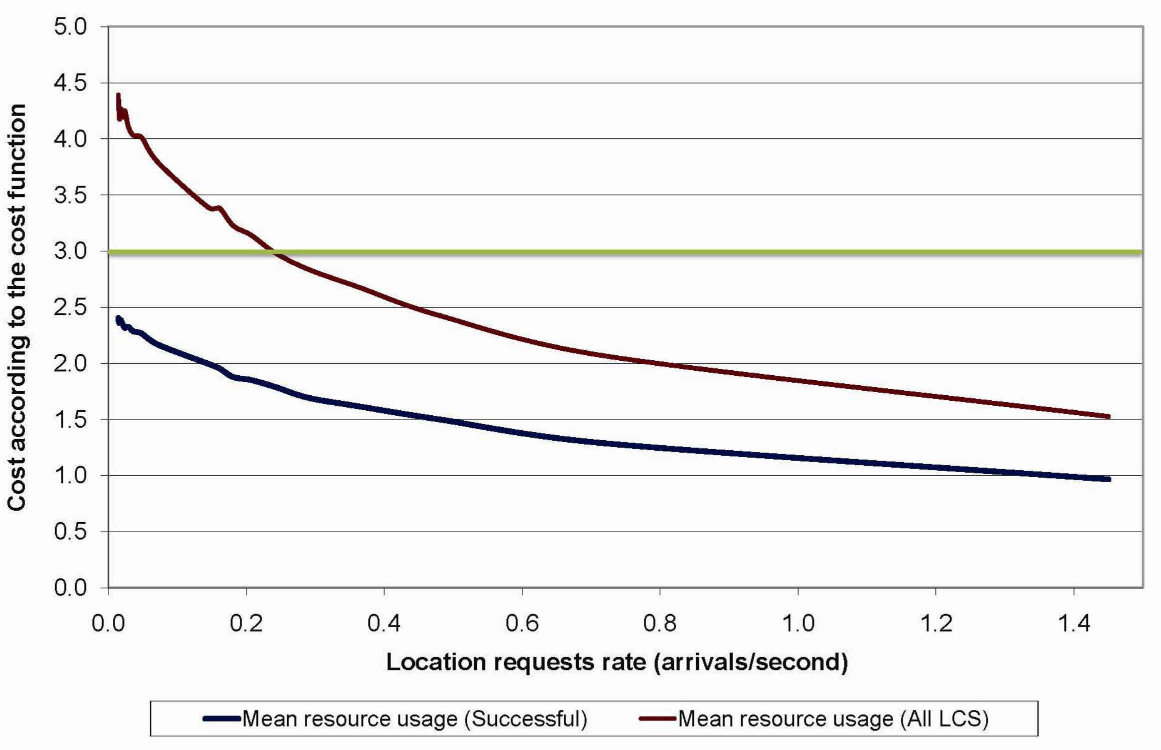


Fig. 9. Average cost of providing LCS in MILCO

Fig. 9 also shows the resources required by MILCO for attending all the LBS, i.e., those successful and unsuccessful. With lighter loads, the cost of providing the LBS is higher than the threshold. This result is due to the fact that unsuccessful LBS usually involve several techniques and hence a cost that is likely to be higher than 3. However, as the successful LBS rate increases with the load, unsuccessful LBS have less impact on the total amount of resources required for LBS delivery. Therefore, the advantages of using MILCO are more noticeable for heavier loads. The resources used are reduced by up to 50.88% in the scenario loaded with 1.45 requests per second and up to 32.2% in the same scenario if only successful LBS are taken into account.

Fig. 10 shows the performance of the cache input module as well as the average number of techniques run per successful LBS. The impact of the cache is stronger as the load becomes heavier. This result was expected because the system is more likely to receive several requests involving short displacements and consequently use the cache feature. In fact, in the scenario involving the heaviest load, the cache handled 52.57% of the requests. The intensive use of the cache results in a reduction of the average number of location techniques used per LBS, and consequently, there is a drop in the amount of resources consumed to attend the location traffic. Furthermore, because the cache is only valid for 2 seconds, 100% of the positions fixed through the cache fulfilled the QoS requirements. The lifetime of cache data could be extended according to the mobility pattern of users at the cost of more complexity in the MILCO implementations. Moreover, Fig. 10 demonstrates the scalability of MILCO, which reduces a 53.67% the average number of techniques used if compared with figures reported in the lightest load scenario.

Simulations show that the impact of the concurrence manager is negligible if compared with the cache module. Fig. 11 displays the percentage of LBS in which the concurrence manager

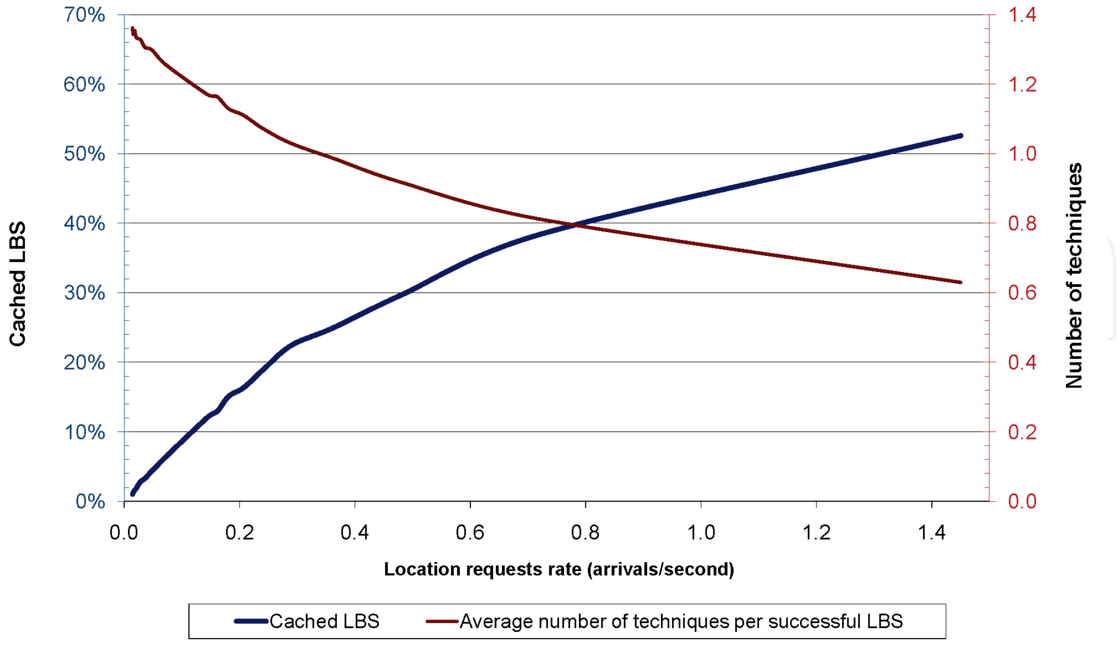


Fig. 10. Performance of the cache module

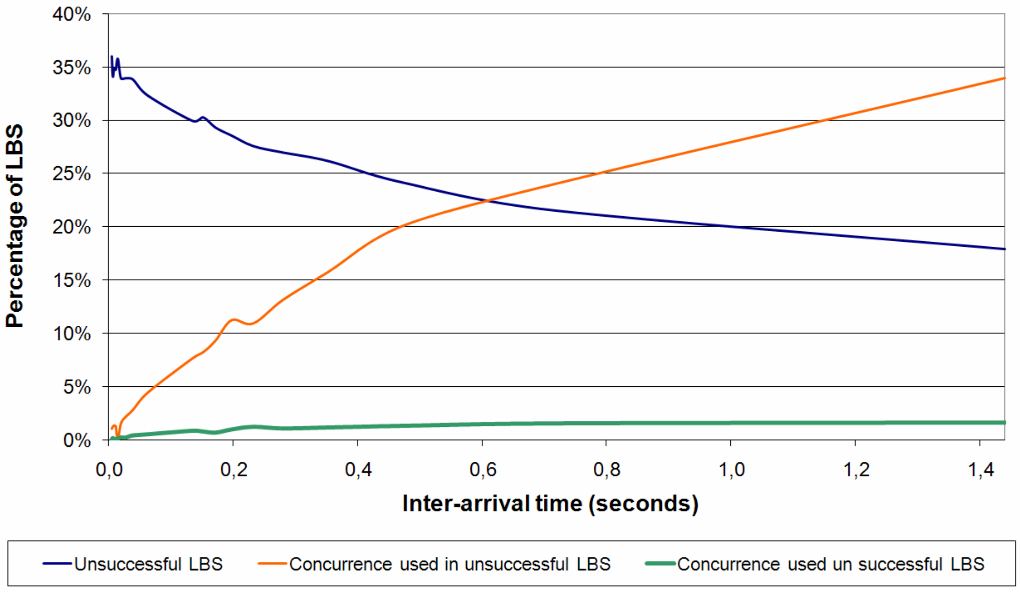


Fig. 11. Performance of the concurrence module

is involved. The rate of unsuccessful LBS is included as a reference. In the best case, only 1.3% of the successful LBS are handled by the concurrence manager. This behavior does not depend on the load. Although every improvement on the successful LBS is welcome, the performance of the concurrence manager for the location is far from being optimum. As long as the load increases, the percentage of unsuccessful LBS decreases and concurrence management appears to be the main reason for LBS to fail. This behavior is due to the fact that a higher load involves more blocked requests and therefore a greater impact on positioning failures. It is expected that running multiple location techniques instead of blocking them will slightly improve the percentage of successful LBS but at the cost of a noticeable increase in the resources consumed. Furthermore, location devices are usually small and computationally restricted, which means that several techniques can rarely run simultaneously.

### 4.2 Handset-based implementation

MILCO can be implemented either in the core network or the user terminal as a new piece of software. The latter approach has been followed to evaluate the performance of MILCO in wireless LAN (WLAN) networks (Martin-Escalona & Barcelo-Arroyo, 2008). Handset-based implementations allow the middleware to use any information available in the user terminal with minimal delay because all data exchange to the middleware is done locally. However, these operations are performed at the cost of reducing the grade of optimization that can be achieved. Only optimizations local to the user terminal can be done because full system optimizations require middleware components to be distributed along the entire LBS supply chain.

The location system architecture is similar to the one presented in Fig. 4. Each time a location request reaches the location system, it is delivered to the user terminal, where the request is finally handled by the middleware. Once there, the middleware analyzes all the requirements included in the location request (e.g., the QoS demanded) and gathers all the facilities provided by the user terminal (e.g., the location techniques implemented). Then, the middleware selects the location technique that best fits the request, i.e., the one expected to achieve the requested QoS with the minimum amount of resources. Finally, the middleware uses the user-terminal facilities to fix the user's position and forward the result to the location service (LBS) client that requested it.

In this implementation of MILCO, input modules are not accounted for even though the application of those modules to MS-based MILCO is obviously feasible. This was ignored because the main purpose of this study was to evaluate the performance of the cost function because similar results for cache and concurrence manager modules are expected independent of the device in which MILCO is implemented.

### 4.2.1 Cost factors

4.2.1.1 Success probability

This cost factor computes the probability of a location technique reaching the QoS requested by means of two histograms, one for the accuracy and one for the response time. Then, the success probability is calculated as:

$$z_2 = Pr\left[A\left(LT_i\right) \leq Accuracy_{requested}\right] \cdot Pr\left[\tau\left(LT_i\right) \leq \tau_{requested}\right], \tag{5}$$

where $z_2$ stands for the success probability, and $A$ and $\tau$ are the estimates for the accuracy and the response time of location technique $LT_i$. Histograms are built locally to a certain area (SP_CELL), usually smaller than the simulation area, to increase the precision of the success-probability estimation. The smaller the *SP_CELLs* are, the more accurate. The drawback of this cellular-fashioned approach is the memory requirement, which increases according to the number of *SP_CELLs* (i.e., the number of histograms computed). Therefore, there is a trade-off between accuracy and memory consumption. The *SP_CELL* matches the coverage area of an access point (AP), i.e., of a cell. Therefore, two histograms are built for each access point available in the network. The mobile equipment uses the pair corresponding to the access point that it is associated with or the one corresponding to the access point with highest RSSI.

WLAN networks are usually deployed indoors, and consequently, the location solution is expected to work under constrained conditions. This behavior means that signal conditions and consequently QoS offered by location techniques may change drastically. Consequently, the histogram computation follows a non-linear approach. Thus, not all the samples in the histogram are weighted the same. Recent samples are favored because they are more likely to be correlated with future positions than the older samples stored in the histogram. The weight of each sample is computed as

$$\alpha_2(n) = \begin{cases} g_{min} + B \cdot \log(n), & 1 \le n \le M \\ g_{max} & , \quad M \le n \le N \end{cases}, \tag{6}$$

where $g_{min}$ and $g_{max}$ are the minimum and maximum gains, respectively, $M$ stands for the number of weighted samples and $N$ is the maximum number of samples used to compute the histogram. $B$ is a scale factor that is based on the $g_{min}$, $g_{max}$ and $M$ parameters. A sliding windows of N samples is run to build the histogram, i.e., if a new sample is added to a histogram with $N$ samples, the oldest sample is removed to make room for the new one and the rest of samples are shifted one position. This approach allows memory in the user terminal to be saved. Notice that with larger values of N, more accurate results are expected.

#### 4.2.1.2 Energy consumption

Energy consumption is one of the most common issues in user terminals implementing location techniques because running such techniques usually demands much more energy than simple communications tasks. This drain is much more noticeable as the number of techniques implemented in the terminal increases. As in case of UMTS networks, this factor constrains the use of the techniques according to the energy consumption and the remaining battery in the terminal. The values proposed for this cost factor, which are displayed in Table 8, are only provided as a proof-of-concept of the location middleware according to the authors' experience. $N_{AP}$ and $N_{SAT}$ in Table 8 stand for the number of access points and satellites that are involved in the positioning process, respectively.

The user terminal consumes energy for several reasons:

- *Attending to incoming services*. These tasks involve an energy drop due to signal demodulation and packet building and interpretation. This process is quantified as one unit of energy dropping.
- *Location technique execution*. The station consumes energy each time a location technique is run.

Table 8 shows, comparatively, the energy drop expected from each location technique. The quantification of this factor should depend on the remaining battery of the terminal because highly demanding location techniques could deplete the battery in a short time making further positioning impossible. The middleware weights this factor as

$$\alpha_3(t,t_0)=\begin{cases}\alpha_3(t_0)\left(1-log\left(\dfrac{Battery(t)}{Battery(t_0)}\right)\right)^{3/2} & ,\alpha_3(t,t_0)<\beta \\ \beta & ,\ \alpha_3(t,t_0)\geq\beta\end{cases} \tag{7}$$

where $t_0$ is the time at which the battery is completely charged, and *Battery(t)* is a function that calculates the remaining battery in the terminal at a certain time *t*. The maximum weight for this factor needs also to be set to limit the impact of this factor in the cost function.

| *Technique* | *Cost* |
|---|---:|
| WLAN Fingerprinting | $10 + N_{AP}$ |
| MEMS | 1 |
| Assisted GPS | $10 + N_{SAT}$ |

Table 8. Energy consumption according to the location technique

4.2.1.3 Expected accuracy

Although location QoS includes several parameters, it is often reduced to a couple of metrics describing the accuracy or delay. An examination of user requirements reveals that accuracy is more restrictive than delay, i.e., users are willing to wait longer for more accurate results. This cost factor aims at giving less weight to those techniques that are not likely tofulfill the accuracy requirements.

The expected accuracy is computed as the average accuracy of each location technique. This should be a static cost factor because the expected accuracy comes from a previous analysis of the performance of each location technique. However, the accuracy of some techniques is dependent on time. It is the case of inertial solutions (MEMS), which depends on the distance travelled since the last reference positioning (e.g., a GPS position). Consequently, this cost factor updates its value over time for these time-dependent techniques while for other techniques, such as WLAN fingerprinting (WLAN-FP) or A-GPS, the cost factor value is constant. Further details on the values of this factor can be found in Table 11, in which average accuracies for the simulated location techniques are provided.

### 4.2.2 Simulated scenario

The simulator was used to model an indoor WLAN network. The proposed scenario consists of a single service and station. The simulation layout models a square-shaped corridor. The user moves freely through corridors, which are 4 m wide. However, they cannot cross the forbidden area (simulation area outside the corridors). Access points placed in the forbidden area simulate instances in different rooms/floors than the user. The propagation pattern follows the Okumura-Hata model for indoor scenarios, with path-loss slope and zero-meter losses set to 3.5 and 40 dB, respectively. Handoffs are handled as explained for the UMTS network simulation. If a new channel is not found during the handoff, the service is interrupted and the user terminal backs off for an exponential time with a mean of 5 seconds. Table 9 reports the main parameters of the propagation pattern, which are based on current industry equipment and the authors' experience.

| *Parameter* | *Value* |
|---|---:|
| Minimum SIR | -9 dB |
| Sensitivity of the stations | -65 dBm |
| Maximum MS transmission power | 17 dBm |
| Minimum MS transmission power | 0 dBm |
| AP transmission power | 17 dBm |
| Handoff threshold for received power | -62 dBm |
| Handoff threshold for SIR at reception | -6 dB |

Table 9. Main parameters of the propagation pattern

This basic scenario is simulated with 9, 16, 25 and 36 access points. These access points are uniformly spread along a square-shaped simulation area, which gives each of them a minimum of 63 meters of coverage at minimum throughput according to the data in Table 9. Table 10 shows the coverage expected in terms of access points available in each scenario. *Scenario_3* models regular network deployments, where the stations receive signal from 2 to 4 APs. More than 4 APs under coverage are not considered because it is unlikely that such a network plan exists in actual WLAN deployments. Hence, *Scenario_4* is included as an example of an over-coverage network, whereas *Scenario_1* and *Scenario_2* are examples of constrained scenarios. These latter scenarios represent realistic situations with only a partially working infrastructure. The minimum coverage is computed according to analytical propagation models. However, the simulations involve factors not included in the analytical calculation.

| *Scenario name* | *Number of APs* | *Minimum coverage* | *Maximum coverage* |
|---|---:|---:|---:|
| Scenario_1 | 9 | 0 AP | 1 AP |
| Scenario_2 | 16 | 0 AP | 2 AP |
| Scenario_3 | 25 | 2 AP | 4 AP |
| Scenario_4 | 36 | 4 AP | 4 AP |

Table 10. The scenarios simulated

Four location techniques have been taken into account in these scenarios: WLAN fingerprinting (FP) and A-GPS as standalone techniques and A-GPS/MEMS and FP/MEMS couplings. Table 11 shows the average accuracy expected for each standalone technique, in which *d* stands for the distance travelled since the last positioning was calculated with WLAN-FP or A-GPS. All these data (along with other data related to the capabilities of the location techniques) have been borrowed from (Thales Selena Space et al., 2007).

Fig. 12 displays the distribution of the positioning error of the WLAN-FP technique according to the data supplied in (Thales Selena Space et al., 2007). The first and second rows in Fig. 12 stand for the error module in the *x* and *y* coordinates, respectively. Only 2D positioning is considered. The column in Fig. 12 represents the number of access points involved in the positioning, which range from 1 (top left) to 4 (top right). Fig. 13 shows the accuracy expected from MEMS in a light indoor scenario according to the data provided in (Thales Selena Space et al., 2007). The simulator couples MEMS with another technique as long as the position provided by such a technique has better accuracy than 4 meters. MEMS keeps working in coupled mode until a position is expected to provide an error beyond 6 meters. Consequently, the results are expected to be slightly conservative because in real scenarios MEMS could be used in more positioning processes.

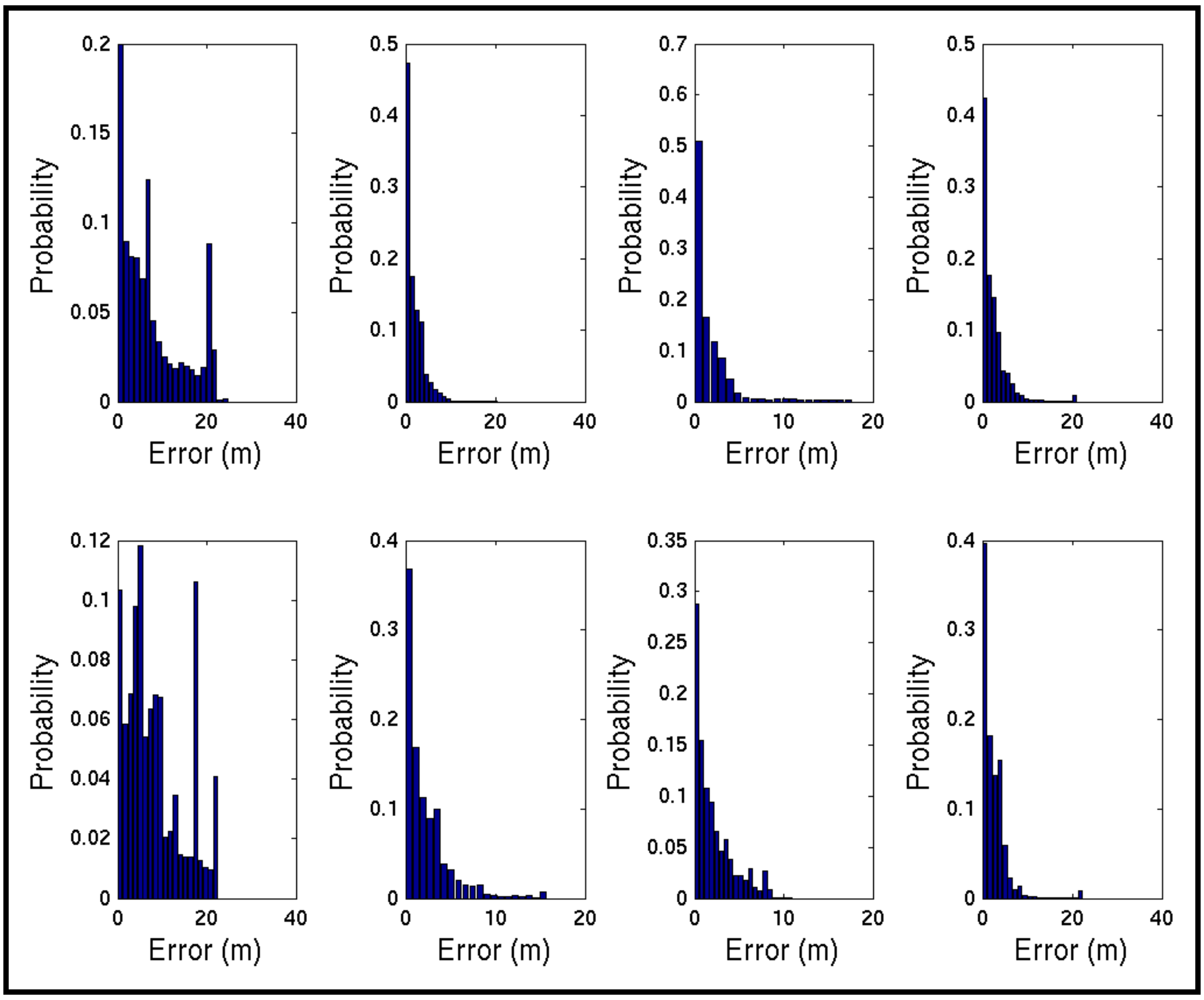


Fig. 12. The accuracy of $x$ (first row) and $y$ (second row) coordinates provided by the WLAN-FP technique with 1 AP (leftmost column) to 4 AP (rightmost column) in sight.

To reduce the complexity of implementing the whole satellite map and estimate the signal availability, the simulator computes the availability of GPS similarly to the UMTS case. the simulator provides the availability for A-GPS satellites uniformly distributed from 2 to 4 satellites if the user is at most 1 meter away from the simulation area edges. These emplacements are considered as light indoor scenarios (i.e., close to windows) and thus A-GPS would be able to receive weak signals from few satellites. Other locations are assumed to be in deep indoor conditions and thus no position at all is provided by A-GPS. The expected values for the accuracy of all techniques are presented in Table 11. In the case of A-GPS, the positioning error is Gaussian distributed with a square coefficient of variation of 0.3.

The cost function includes all the cost factors presented: success probability, energy consumption and expected accuracy. The $N$ and $M$ parameters in Equation (6) are set to 256 and 512 samples, respectively, and the minimum ($g_{min}$) and maximum ($g_{max}$) gains for those samples are 1 and 8, respectively.

The weights of the factors in the cost function are 1 and 0.0017 for the successful probability and expected accuracy, respectively. These figures are used to provide a cost of 1 under the worst conditions. The weight of the energy-consumption factor is provided by Equation (7), in which $\alpha_3(t_0)$ is set to 1 and the maximum value ($\beta$) is limited to 3. Consequently, the cost

| *Technique* | *Expected accuracy* |
|---|---|
| WLAN Fingerprinting | 12.2766 m (1 access point)<br>3.4058 m (2 access points)<br>3.1982 m (3 access points)<br>3.9329 m (4 access points) |
| MEMS | $1.65 + 0.2825 \cdot d$ meters |
| Assisted GPS | 3 meters (only very light indoor scenarios) |

Table 11. Expected accuracy of location techniques as standalone

function can produce values from 0 to 5. These values allow the optimum technique to be used as long as the battery in the user equipment has enough charge and smoothly switches to a power-saving technique as long as the energy is going to run out. Once the battery runs out, the station switches off for 5 seconds and then turns on completely recharged. The time the station spends between switching off and on simulates the network re-association process.

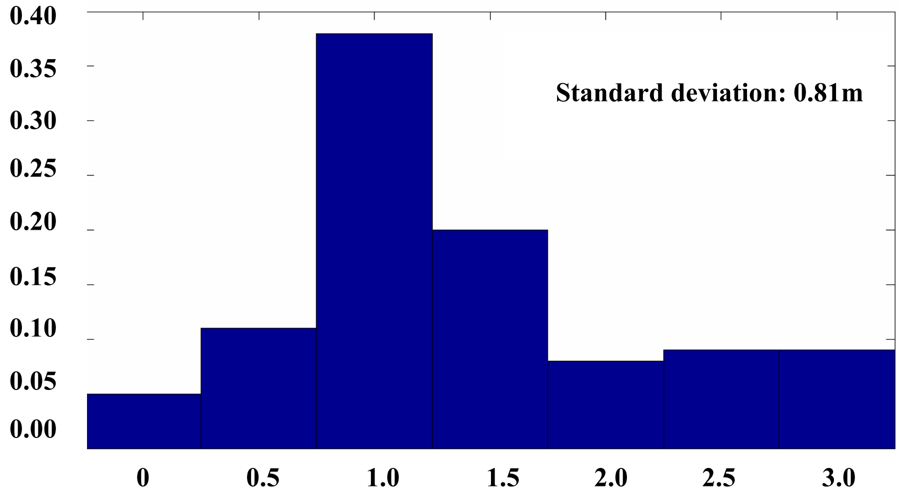


Fig. 13. The accuracy of MEMS in light indoor scenario

One single LBS is simulated, generating one request each 5 seconds, and requesting an accuracy of 6 meters. Simulations do not account for the response-time in the QoS requirements. This approach was taken because in indoors, customers perceive more degradation in the QoS when the required accuracy is not achieved. Furthermore, the response-time in mobile-based techniques is expected to be mostly the same, more if it is taken into account that most of the time used by the LCS is spent communicating with the network, not on executing the technique. The cost function is run twice at most to avoid infinite looping and save resources in the terminal.

### 4.2.1 Simulation results

This section presents the performance results obtained by MILCO and compares them with those achieved using WLAN-FP and A-GPS as standalone. MEMS is not evaluated on its own because this technique positions relatively to a previous location provided by WLAN-FP or A-GPS. Therefore, the positioning error in MEMS drifts with the distance covered, and as consequence MEMS needs correction updates from other location techniques periodically.

Table 12 presents the QoS achieved by means of the location techniques when used as standalone and the results obtained when middleware is run. The first parameter taken into account is the location traffic carried. Two situations may lead to a location request not being performed: the station being in a position without radio network coverage, and the station being in a *recharging* condition. The best performance was achieved by WLAN-FP and MILCO, which are able to carry more than the 80% of the traffic. Only slight differences can be found between the performances of these two techniques. However, MILCO's performance is more stable due to the better management of resources (i.e. less *recharging* situations), and the support of MEMS (i.e. coverage improvement). Although A-GPS performs poorly as a standalone system, because it only works under light indoor conditions, it must be noted that a single A-GPS position can enable the MEMS techniques for a long time. In all scenarios, the traffic carried by MILCO is at least as good as that carried by WLAN-FP used as a standalone system and usually better than the latter. It must be noted that A-GPS as a standalone system provides excellent figures for the rest of the variables in this study, but they only apply to less than 0.5% of the traffic (i.e., location traffic already carried). Accordingly, A-GPS results cannot be considered suitable to deliver any kind of LBS as a standalone system, and consequently, related results will not be commented on hereafter.

The percentage of traffic successfully handled measures the amount of carried traffic that yields a successfully attended request (i.e., with the required QoS already achieved). Under excellent coverage conditions (i.e., *Scenario_4*), MILCO and regular location techniques provide almost the same ratio of successfully handled LBS. However, reducing the number of access points in sight drastically impacts the figures provided by the WLAN-FP solution. The same does not apply to MILCO, which is not as sensitive to the number of access points in sight. This behavior is due to the fact that MILCO is able to use MEMS when positions provided by WLAN fingerprinting become noisy. As previously observed, under the worst conditions (i.e. *Scenario_1*), MILCO successfully handles 91% of carried traffic versus 49.1% achieved by WLAN fingerprinting used as a standalone system. It is because MILCO takes also benefit from A-GPS positions. According to these results, MILCO is more robust in front of integrity failures, since it manages several location techniques and modulates their use according to the resources in the network.

Data in Table 12 shows that in the first three scenarios MILCO outperforms WLAN-FP in terms of average accuracy, whereas WLAN-FP provides better accuracy in *Scenario_4*. However, the LBS client demands positioning errors lower than 6 meters, and in this scenario, both MILCO and WLAN-FP provide figures for positioning error below this threshold. The less accurate positions for MILCO in this scenario are a consequence of the noisier positions provided by the MEMS technique. On the other hand, the use of MEMS reduces battery consumption. MILCO looks for the optimum technique for each request and thus modulates the use of MEMS to fulfill the QoS requirements and at the same time save network resources.

Similar results are reported in terms of resource consumption (according to the cost function). The maximum cost expected according to the simulation parameters is 10, which is twice the achievable maximum cost. According to the data in Table 12, MILCO reduces the cost of providing LBS by more than 46% in all scenarios. As expected, the cost increases with the lack of available access points because unsuccessful LBS involve several techniques being run. Better results are observed only if successful LBS are taken into account, which achieve a reduction in the consumption of resources higher than 50% in all the scenarios.

| *Parameter* | *Scenario* | *A-GPS* | *WLAN-FP* | *MILCO* |
|---|---|---|---|---|
| Carried location traffic | Scenario 1 | 0,39% | 85,70% | 90,56% |
| | Scenario 2 | 0,42% | 80,14% | 99,93% |
| | Scenario 3 | 0,39% | 99,73% | 99,94% |
| | Scenario 4 | 0,42% | 92,29% | 99,97% |
| Successful LBS (only carried traffic) | Scenario 1 | 100,00% | 49,11% | 91,01% |
| | Scenario 2 | 100,00% | 51,17% | 92,70% |
| | Scenario 3 | 100,00% | 61,96% | 94,81% |
| | Scenario 4 | 100,00% | 97,45% | 99,60% |
| Accuracy | Scenario 1 | 2,82 m | 9,72 m | 5.23 m |
| | Scenario 2 | 3,13 m | 9,15 m | 5.06 m |
| | Scenario 3 | 3,09 m | 7,69 m | 4.73 m |
| | Scenario 4 | 2,96 m | 2,60 m | 3.92 m |
| Average cost | Scenario 1 | 2,22 | 5,40 | 2,88 |
| | Scenario 2 | 2,21 | 5,35 | 2,79 |
| | Scenario 3 | 2,05 | 5,11 | 2,61 |
| | Scenario 4 | 2,18 | 4,13 | 1,97 |
| Amount of location techniques per LBS | Scenario 1 | 1.00 | 1,71 | 1,56 |
| | Scenario 2 | 1.00 | 1,67 | 1,51 |
| | Scenario 3 | 1.00 | 1,55 | 1,41 |
| | Scenario 4 | 1.00 | 1,10 | 1,12 |

Table 12. The QoS achieved by techniques as standalone and MILCO

According to the results shown, extended battery lifetime and improved performance is expected when using MILCO. Furthermore, the average number of techniques required by LBS is reduced as the availability conditions improve (as can be expected). MILCO uses fewer techniques to attend LBS on average, except for the best scenario, in which the excellent success percentages cause MILCO to achieve the same performance as WLAN-FP. Even though the results are statistically similar, the techniques used by MILCO involve less resource consumption than in the case of WLAN-FP as a standalone technique.

## 5. Concluding remarks

This chapter offers a brief overview of middleware for positioning. A new middleware for optimizing the cost of LBS provisioning was presented. This novel approach has not been examined closely by the research community even though a great demand for LBS is expected. Different implementations of the middleware (handset-based and network-based) were presented and evaluated, and in all of them, the middleware provides a way to reduce the resources consumed to provide LBS and, at the same time, optimize several parameters of the LBS provisioning chain, such as the stability of the accuracy or the scalability of the location system.

## 6. References

3GPP. (2004). Functional stage 2 description of Location Services (LCS). Release 6. *3GPP TS 23.271 (v6.8.0)* . 3GPP.

3GPP. (2002). Location Services (LCS); Functional description; Stage 2. *3GPP TS 03.71 (v8.7.0)* . 3GPP.

3GPP. (2004). Universal Mobile Telecommunications System (UMTS); RF system scenarios. *3GPP TR 25.942 (ETSI TR 125 942) , 6.3.0* . 3GPP.

Atsan, E., & Özkasap, Ö. (2006). A Classification and Performance Comparison of Mobility Models for Ad Hoc Networks. *Ad-Hoc, Mobile, and Wireless Networks (LNCS) , 4104/2006*, 444-457.

Barcelo, F., & Martin-Escalona, I. (2004). Coverage of Hybrid Terrestrial-Satellite Location in Mobile Communications. *European Wireless Conference: Mobile and Wireless Systems beyond 3G*, (pp. 475-479).

Biswas, P., Han, S., & Wu, J. (2002). Location Caching in the Mobile Middleware Platform. *Third International Conference on Mobile Data Management*, (pp. 172-174).

Capone, A., & Redana, S. (2001). Call Admission Control Techniques for UMTS. *Vehicular Technology Conference (VTC-Fall). 2*, pp. 925-929. IEEE.

EMILY system trials report. (2005). *EMILY project (IST-2000-26040)* . European Comission.

Fritsch, D., & Volz, S. (2003). Nexus - the mobile GIS-environment. *Workshop on Mobile Future and Symposium on Trends in Communications (SympoTIC)*, (p. 1.4).

Holma, H., & Toskala, A. (Eds.). (2000). *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications.* Wiley Technology Publishing.

Java Community Process. (n.d.). *JSR 179: Location API for J2METM*. Retrieved 2009 йил May from Community Development of Java Technology Specifications: http://www.jcp.org/en/jsr/detail?id=179

Küpper, A., Treu, G., & Linnhoff-Popien, C. (2006). TraX: a device-centric middleware framework for location-based services. *Communications Magazine , 44* (9), 114-120.

LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., et al. (2005). Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Pervasive Computing (Lecture Notes in Computer Science)* (Vol. 3468/2005, pp. 116-133). Springer Berlin / Heidelberg.

Mahmoud, Q. H. (Ed.). (2004). *Middleware for Communications.* John Wiley & Sons, Ltd.

Martin-Escalona, I., & Barcelo-Arroyo, F. (2006). A middleware approach for reducing the network cost of location traffic in cellular networks. *Journal of Communications Software and Systems (JCOMSS) , 2* (4), 305-312.

Martin-Escalona, I., & Barcelo-Arroyo, F. (2008). An approach to increase the scalability of Location Systems in WLAN networks. *Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications (MOBILWARE). 278*, pp. 1-6. Innsbruck (Austria): ACM.

Martin-Escalona, I., & Barcelo-Arroyo, F. (2007). Performance evaluation of Middleware for provisioning LBS in cellular networks. *International Conference on Communications (ICC)* (pp. 5537 – 5544). Glasgow: IEEE.

Ministerio de Ciencia y Educacion. (2009). Encaminamiento en redes ad-hoc basado en localización de terminales. Aplicaciones y sinergias con localización pasiva. *TEC2009-08198* .

Nuyami, L., Lagrange, X., & Godlewski, P. (2002). A Power Control Algorithm for 3G WCDMA System. *European Wireless.* IEEE.

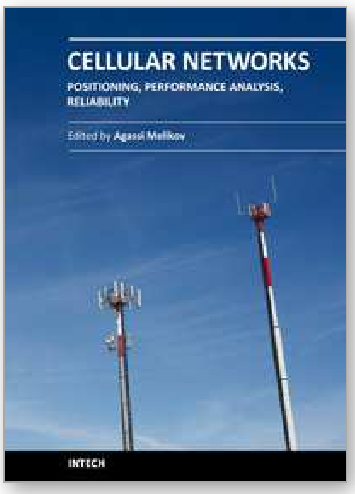
OGC. (n.d.). *Location Service (OpenLS)*. Retrieved 2009 йил May from Open Geospatial Consortium, Inc.: http://www.opengeospatial.org/standards/ols

Ranganathan, A., Al-Muhtadi, J., Chetan, S., Campbell, R., & Mickunas, M. D. (2004). MiddleWhere: a middleware for location awareness in ubiquitous computing applications. *ACM/IFIP/USENIX international conference on Middleware*, *78*, pp. 397-416.

Spanoudakis, M., Batistakis, A., Priggouris, I., Ioannidis, A., Hadjiefthymiades, S., & Merakos, L. (2004). Extensible Platform for Location Based Services Provisioning. *International Conference on Web Information Systems Engineering Workshops (WISEW)* (pp. 1-8). IEEE.

Thales Selena Space; EPFL; UPC; France Telecom. (2007). *Deliverable D077,* . 6th FP IST Liaison project: LIAISON.

Zander, J., & Kim, S. L. (2001). Radio Resource Management for Wireless Networks. Artech House.



**Cellular Networks - Positioning, Performance Analysis, Reliability**
Edited by Dr. Agassi Melikov

ISBN 978-953-307-246-3
Hard cover, 404 pages
**Publisher** InTech

**Published in print edition** April, 2011

Wireless cellular networks are an integral part of modern telecommunication systems. Today it is hard to imagine our life without the use of such networks. Nevertheless, the development, implementation and operation of these networks require engineers and scientists to address a number of interrelated problems. Among them are the problem of choosing the proper geometric shape and dimensions of cells based on geographical location, finding the optimal location of cell base station, selection the scheme dividing the total net bandwidth between its cells, organization of the handover of a call between cells, information security and network reliability, and many others. The book focuses on three types of problems from the above list - Positioning, Performance Analysis and Reliability. It contains three sections. The Section 1 is devoted to problems of Positioning and contains five chapters. The Section 2 contains eight Chapters which are devoted to quality of service (QoS) metrics analysis of wireless cellular networks. The Section 3 contains two Chapters and deal with reliability issues of wireless cellular networks. The book will be useful to researches in academia and industry and also to post-gradute students in telecommunication specialitiies.

Israel Martin-Escalona, Francisco Barcelo-Arroyo and Marc Ciurana (2011). Middleware for Positioning in Cellular Networks, Cellular Networks - Positioning, Performance Analysis, Reliability, Dr. Agassi Melikov (Ed.), ISBN: 978-953-307-246-3, InTech, Available from: http://www.intechopen.com/books/cellular-networks-positioning-performance-analysis-reliability/middleware-for-positioning-in-cellular-networks

INTECH
open science | open minds

**InTech Europe**
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

**InTech China**
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821