Multi-Resolution Data Management for Opportunistic Networking*

Iacopo Carreras, David Tacconi, Daniele Miorandi and Imrich Chlamtac
CREATe-NET
Via Solteri 38 – 38100 – Trento (Italy)
email: {name.surname}@create-net.org

Abstract

In this work, we present a multi-resolution data management scheme for efficiently storing and diffusing information, which is originating from sensors and augmenting mobile applications. The proposed mechanism exploits the discrete wavelet transform properties for providing a location-dependent data management scheme, which alleviates the constraints on the storage resources needed for running context-aware mobile applications. The data management scheme has been implemented and evaluated by means of numerical simulations.

1 Introduction

The increasing pervasiveness of mobile devices is enabling totally new networks and mobile applications, where information is diffused with no need for dedicated network infrastructure by employing epidemic-like information dissemination techniques [5, 7, 16, 3]. By taking advantage of proximity communications, any communication opportunity, or “contact” [7], is exploited for exchanging information. Mobile nodes contributing to the diffusion of information may range from cars [16], to people [7], or buses [3]. Such enlarged networking possibility is enabling a wide range of innovative mobile application scenarios such as vehicular sensor networks [15], where sensed data is distributed by means of local communications among cars, or distributed context provisioning, where information is shared among mobile nodes in an ad-hoc fashion [20]. As the number of mobile users increases, and more rich content (i.e., images) enters the application scene [21], in order to apply such techniques, it becomes imperative to make a parsimonious use of the limited resources of mobile nodes, i.e., storage and battery. Hence, efficient mobile data management techniques need to be devised in order to control such an epidemic diffusion of data.

In this work, we consider an application scenario where mobile nodes are running context-aware applications [20]. The information, which is contributing to the creation of the surrounding context [12], is originating from sensors embedded in the environment, and is exchanged by mobile nodes whenever in mutual communication range. Typically, context-aware applications require fine-grained data on the surrounding environment, and only a coarse approximation of remote regions. It is therefore a natural choice to manage the information in such a way that data originating from nearby sensors is kept (and stored) at full resolution, while information originating from remote sensors is compressed by reducing its resolution. This multi-resolution property is achieved by exploiting the characteristics of the Discrete Wavelet Transform, which is applied to the stored information. The benefits of the proposed approach are twofold: first, it becomes possible to trade off the accuracy of the information for the required storage resources (i.e., memory allocation required on mobile devices for storing the context information) depending on the specific application scenario; second, by allowing a compact representation of the stored information, it becomes possible to apply epidemic-like dissemination mechanisms for diffusing data among mobile nodes.

The remainder of this paper is organized as follows. In Sec. 2, the motivation of this work, together with a high level description of the proposed data management scheme, are presented. This section includes also a brief summary of the most significant work in this area. Sec. 3 provides a short overview of wavelet techniques. In Sec. 4, the multi-resolution data compression scheme is detailed in the case of sensors measuring a random field. Sec. 5 presents the multi-resolution data management scheme, and evaluates it through extensive numerical simulations. Finally, Sec. 6 concludes the paper pointing out some promising directions for future work.

2 Overview and Related Work

The considered application scenario consists of mobile users moving around and running context-aware applications. Contextual information is originating from sensors embedded in the environment, and is gathered by mobile nodes when in the mutual communication range of sensors. We are therefore considering a network scenario where the architecture has 2 distinct tiers [5]. The lowest layer is constituted by miniaturized devices embedded in the environment; the task of such sensor devices is to measure a physical phenomenon upon demand and to transmit the collected

---

*This work has been partially supported by the European Commission within the framework of the BIONETS project IST-FET-SAC-FP6-027748, www.bionets.org
information to mobile nodes within radio range. Opposed to sensor nodes, the other layer is made by mobile users, which carry with them personal handheld device, where context-aware services are running. Such services are fed with information originating from sensors and dynamically build what is usually referred as the context for user-situated services [14].

Furthermore, in order to diffuse environmental information with no need for dedicated network infrastructure, epidemic information dissemination techniques are employed [5, 7, 16]. A central issue in order to apply such techniques, anyhow, is to make a parsimonious use of the limited resources of mobile nodes, i.e., storage and battery. For this scenario, the original contribution of this paper is an entire location-dependent degrading data management model, which operates a wavelet-transform of data with a variable compression ratio (and thus memory allocation), depending on the current location of the mobile user. This lets us trade off accuracy of the information for the resources needed to store it. Notice that this is a reasonable trade-off, since context-aware services are expected to require precise information on the surrounding environment, and a brief summary of outer regions.

The same degrading storage model is then exploited for efficiently exchanging the stored information between two mobile users encountering on-the-move.

2.1 Related Work

Wavelet techniques have been widely adopted in image and audio processing for dynamically varying the resolution and compression of data, or for efficiently denoising signals. As an example, the wavelet transform is the basis of the video MPEG and image JPEG standards [17, 26]. The wide adoption of JPEG standard is a testament of its success. A similar approach is at the basis of the JPEG2000 [22] standard, which varies the applied wavelet decomposition depending on the target compression ratio.

Wavelets technique have being recently considered in the area of Wireless Sensor Networks for providing in-network wavelet-based aggregation of information originating from sensors [13]. In response to sink queries, only a compressed aggregate information is sent. Since the wavelet decomposition preserves the properties of the original signal (i.e., trends, peaks, average, etc.). the sink can successively ask for detailed information only from the subset of sensor nodes that presented interesting data in the aggregate response. This results in a significant reduction of data traffic circulating in the network and, thus, in an extension of the network life-time. The work of [13] was successively extended by [24], where a wavelet transform, specifically tailored to non-regular deployments of sensor nodes, is defined.

3 Wavelets Decomposition

Wavelet analysis represents a powerful tool to obtain a compact and highly flexible representation of signals. Wavelet representation offers significant advantages over, e.g., Fourier series analysis, thanks to its capability of carrying a joint time-frequency representation of the considered signal [10, 8]. A wavelet transform consists in decomposing an input signal over a set of basis of wavelets, which are functions with a compact support (i.e., they are defined over a finite and convex interval) and are oscillatory (i.e., their integral over the support is zero), from which it comes the name “wavelets”. Small waves.

![Figure 1. Wavelet decomposition in the frequency domain.](image)

The Discrete Wavelet Transform (DWT) is the mostly used practical tool in wavelet applications. The DWT uniquely associates a signal (square-integrable over a n-dimensional space $\mathbb{R}^n$) to a sequence of coefficients, representing the projections of the signal over the basis of $\mathbb{R}^n$. This operation is called wavelet series decomposition. Decomposing a signal over a wavelet basis corresponds to cut the original signal in different pieces and analyze each piece separately. As depicted in Fig. 1 for the frequency domain, each piece is cut by means of a fully scaled and modulated window, which is applied over the signal to be transformed. This is also referred in the literature as Multi-Resolution Analysis (MRA) [10]. Representing the full original signal information would require an infinite number of wavelet functions (and coefficients), thus being able to analyze the original signal over the full range of frequencies and for all its time duration. In order to overcome this problem, wavelet decomposition is conducted down to a certain frequency $f^*$, whereas all the lower frequencies are “wrapped up” in a trend component obtained by applying a low-pass filter. The described process is briefly summarized in Fig. 1.

The $k-level$ multiresolution analysis of an input signal $f$ can be synthetically summarized as follows:

$$f \rightarrow A_k + D_k + \ldots + D^2 + D^1,$$  \hspace{1cm} (1)

where $A^k$ represents the trend component of the k-MRA, while $D^i$ the detail component of the $i-th$ level of decomposition. It is important to notice that it is possible to reconstruct perfectly the trend coefficients at level $k-1$ by applying the inverse DWT to $A^k$ and $D^k$. It follows that it is possible to reconstruct perfectly the original signal starting from (1).

A similar approach applies to the 2-dimensional case. In this case, the 1-level wavelet decomposition of an input signal $f$ can be summarized as follows:

$$f \rightarrow \left( \begin{array}{c|c}
LL & HL \\
\hline
LH & HH
\end{array} \right),$$  \hspace{1cm} (2)
where (a) $LL$ is computed by calculating the trend along the rows of $f$, followed by computing the trend over the columns; (b) $HL$ is the subimage computed by calculating the trends along the rows, followed by computing the fluctuations along the columns; (c) $LH$ is the subimage computed by calculating the trends along the columns, followed by computing the fluctuations along the rows; (d) $HH$ is the subimage computed by calculating the fluctuations along the columns, followed by computing the fluctuations along the rows.

In the rest of the paper, we will use, for our numerical results, the Haar wavelet [8]. The Haar wavelet is indeed the simplest wavelet function, and enables a rapid implementation while capturing all the advantages of DWT.

4 Compressing a Random Field

4.1 Problem Formulation

Let us associate the surrounding context to a physical phenomenon to be monitored. This physical phenomenon is modeled as an isotropic random field $X$, defined on a suitable probability space $(\Omega, \mathcal{F}, P)$ [23]. Assuming a two-dimensional space, the process $X$ can be written as $X(x, y, t)$, $(x, y)$ representing a location and $t \in [0, +\infty)$ the time index.

We assume the Space/Time Random Field (S/TRF) $X$ to be stationary and ergodic, which eliminates the possibility of any spatial or temporal trend. While this assumption may look restrictive, it is generally possible to separate the trend component from the stationary one, and deal with them separately. In this work, we will concentrate on the stationary component, which accounts for the variability of the physical phenomenon to be monitored.

Let us assume $N^2$ sensors to be deployed in a regular fashion (so that their topology forms a grid) over an $L \times L$ square playground, as depicted in Fig. 2 for $N = 5$. Without loosing in generality, let us further assume $N$ to be a power of 2. The complete sensor status, representing the contextual information, is stored in the Sensor Map (SM), a square matrix of size $N \times N$:

$$SM = \begin{pmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ s_{1,N} & s_{2,N} & \cdots & s_{N,N} \end{pmatrix},$$  

where $s_{i,j}$ represents a sample of the random field at position $(i, j)$ over the $N \times N$ square grid. As we are not interested in monitoring historical trends of data, we neglect for the moment the temporal domain and simply assume that a value of the sensor map corresponds to the most recent reading from the corresponding sensor. We are now interested in understanding how well DWT can be used to compress the SM matrix.

For the sake of simplicity, we limit our analysis to the case of the field $X$ constituting a Gaussian random field with zero mean and unit variance. The process $X$ is therefore completely specified (in the statistical sense) by its covariance matrix $K$.

As it is commonly assumed for environmental processes [23], we adopt separable space-time models, so that for two locations $s, s'$ and two time epochs $t, t'$ the process normalized covariance can be written as:

$$\rho(s, t; s', t') = \rho_1(s, s') \rho_2(t, t'),$$  

where $\rho_1(s, s')$ and $\rho_2(t, t')$ are the normalized covariance of the spatial and temporal components, respectively. Further, we have assumed that the spatial component of the covariance follows an exponential (isotropic) decay:

$$\rho(\zeta) = \exp\left(-\frac{\zeta}{\lambda}\right),$$  

where $\zeta$ is the (Euclidean) distance between two points on the grid and the parameter $\lambda$ describes the rate at which correlation varies. The parameter $\lambda$ is usually referred to as the “correlation length” or “correlation scale” of $\rho$. We assume also to sample the temporal component of (4) with a sufficiently large period, so that we can consider the space/time random field as a stochastic image, where time is seen as a quality index of each spatial component in terms of freshness of data.

We have used stochastic simulations [2], [11] for generating alternative, equally probable, realizations of the space/time random field. By adopting a grid topology for the sensors, each realization represents a possible stochastic image of the space/time distribution of the random field, and satisfies the constraints imposed on the covariance model.

Varying the correlation length results in a different entropy [9], and, thus, in a different compressibility of the stochastic image. The entropy decreases while increasing the correlation length [9].

\[1\text{It is worth recalling that the gaussian RF presents the highest entropy of any random field with zero mean and covariance matrix } K.\text{ Hence, since we are interested in compressing the SM and since to lower entropies correspond a better compressibility, the gaussian assumption represents an upper bound.}\]
4.2 Compressing the Random Field Through Standard Wavelet Techniques

An important property of the wavelet transform is the compaction of energy. Although the wavelet decomposition maintains the total energy of the original signal, typically, the trend component accounts for a large percentage of the energy of the transformed signal. This results in the magnitude of its coefficients being significantly larger than the magnitude of the detail ones.

We now move our attention to the repartition of energy between the trend and detail components for various levels of decomposition. Varying the level of decomposition consists in stopping the iterative DWT decomposition at a certain level $k$. As an example, in Fig. 1 we could stop the decomposition at the first iteration, with the detail component obtained with a $8B$ band-pass filter and the trend with a $\frac{L_0}{2} + 2B$ low-pass filter, or further iterate the process to deeper levels.

Let us assume $64 \times 64$ sensors to be regularly distributed over a $1280 \times 1280 \ m^2$ square playground, and the random field to be characterized by a correlation length $\lambda = ND_0 = 1280 \ m$. In Fig. 3, we reported the repartition of energy for various levels of decomposition. The results are obtained averaging $50$ simulations. As it is intuitively clear, most of the energy is concentrated in the trend component. This effect is particularly evident for low decomposition levels. When we attempt to squeeze the energy into ever smaller time intervals (higher band-pass filters) some energy inevitably leaks out.

![Figure 3. Energy repartition between the trend and details components at various decomposition levels, in the case of $64 \times 64$ sensors regularly distributed over a $1280 \times 1280 \ m^2$ playground, and a random field with correlation length $\lambda = 1280 \ m$.](image)

In general, wavelet-transformed images are then compressed by means of Thresholding Schemes, which consist in comparing each DWT coefficient with a fixed threshold $T_h$, and in keeping only the coefficients above the defined threshold, while setting to 0 the remaining ones $^2$. The processed signal results then composed by a high number of 0s, and can be compressed by means of standard compression methods (i.e., run-length encoding, etc.) $^{[25, 19]}$. Obviously, by selecting a large threshold value, a high compression rate is achieved at the cost of a corrupted reconstructed image. The main objective in wavelet compression is selecting a good threshold value, depending on the specific application scenario considered, in order to ensure either a given compression rate, or a chosen quality of the reconstructed image.

We now consider the percentage of detail coefficients that are set to zero at each level of decomposition by applying the described thresholding scheme. We select a threshold so that at least 95% of the energy of the original sensor map is kept, applying the method in $^{[25]}$. In Fig. 4, the number of zeros in the case of a $64 \times 64$ sensors regularly deployed over a $1280 \times 1280 \ m^2$ playground are presented in the case of a correlation length of $320 \ m$ and $1280 \ m$. Results are obtained averaging $50$ simulations. Clearly, different correlation lengths lead to different compression ratios, since the correlation length influences the resulting entropy of the random field. As we can see from Fig. 4, the first levels of decomposition are characterized by an extremely high percentage of null values in the detail coefficients and this percentage increases with larger deployments (i.e., larger values of $N$) or higher correlation lengths $^{[6]}$.

![Figure 4. Percentage of zeros in the detail components applying adaptive thresholding, in the case of $64 \times 64$ sensors regularly distributed over a $1280 \times 1280 \ m^2$ playground, and a random field with correlation length $\lambda = 1280 \ m$.](image)

4.3 The Multiresolution Data Compression Scheme

From the previous analysis it possible to conclude that, when compressing a sufficiently high correlated random field and for sufficiently large deployments ($N \gg 1$) $^{[6]}$, only a minimum amount of energy of the original signal will be lost when removing the detail coefficients. This is particularly true for low level of decomposition (Fig. 4). This suggested us an alternative approach that we termed Multi-Resolution Data Compression (MRDC) scheme. The MRDC scheme consists in removing all the details coefficients up to a certain level of decomposition, and in main-
taining only the trend component of the DWT transform. The level of decomposition reflects the compression ratio of the sensor map (and therefore the memory required for storing the compressed image), since the deeper is the decomposition level, the more are the coefficients that will be dropped. As an example, let us consider an \( N \times N \) image, with \( N \) power of 2. The lowest compression ratio can be obtained by performing a single level DWT of the SM, and maintaining the resulting trend component: the trend matrix is a \( \frac{N}{2} \times \frac{N}{2} \) matrix. Conversely, the highest compression ratio can be obtained by iterating the DWT down to the maximum decomposition level, which is \( \log_2 (N) \) for \( N \) power of 2. In this case, the trend matrix consists of a single coefficient, which represent the maximum compression achievable with the MRDC scheme. Obviously, the deeper the level of decomposition, the higher the compression rate at the cost of a lower quality of the reconstructed image.

The benefits deriving from the MRDC mechanism are twofold. First, it allows for a great level of control over the memory resources allocated for storing the compressed image. In fact, at each level of decomposition the trend component has a well defined number of coefficients; second, it presents an extremely low-complexity implementation, and can therefore run on processing-constrained devices. This allows us to easily trade off information accuracy with resources, depending on the specific application scenario. Ideally, the processed signal is then transmitted and reconstructed on the receiver side. The reconstructed signal will be an estimate \( \hat{S} \) of the original stochastic image \( S \). In fact, while reconstructing the signal, all the details components that were previously removed are set to zero, and this obviously results in a degradation of the reconstructed image. In order to quantify this degradation we considered the distortion \( D \), defined as:

\[
D = \frac{1}{N^2} \sum_{i=0}^{N} \sum_{j=0}^{N} (S_{i,j} - \hat{S}_{i,j})^2,
\]

where \( S_{i,j} \) and \( \hat{S}_{i,j} \) are, respectively, the elements \((i, j)\) of the original and reconstructed sensor map, and \( N \) is the size of the SM matrix. In Fig. 5, the distortion of the reconstructed signal \( \hat{S} \) of a \( 64 \times 64 \) sensors image, with sensors equally spaced by \( D_S = 20 \) m and distributed over a \( 1280 \times 1280 \) m² playground, is depicted. The setting encompasses a variable correlation length \( \lambda \) and a variable level of compression \( \text{ComprLev} \). Results are obtained running 50 simulations and considering the corresponding 98% confidence interval. The \( \text{ComprLev} \) represents the decomposition level down to which details coefficients are discarded. As it is intuitively clear, for any fixed correlation length \( \lambda \) the distortion increases while increasing the compression level of the image (thus reducing the memory resources needed for storing the SM), since we are discarding a higher number of DWT coefficients. Additionally, for a given compression level, increasing the correlation length results in a higher correlation of the sampled random field and, thus, in a higher significance of the trend (low-pass) component. This affects the capacity of the Haar DWT to represent the RF, and is reflected in a reduced distortion of the reconstructed signal.

5 A Multi-Resolution Data Management Scheme

5.1 Scheme Description

Let us assume \( N^2 \) sensing devices to be uniformly deployed over a \( L \times L \) square playground, with the aim of providing a precise estimation of the random field \( X \) when asked to. Let us further assume the playground to be logically divided into \( T \times T \) square tiles, each tile containing \( l = \frac{N^2}{T^2} \) sensors. \( M \) mobile nodes are moving over the playground and reading sensor nodes when in mutual communication range. Each sensor reading is constituted by the tuple: \(<\text{value}, \text{time}, \text{pos}>\), with \text{value} being the reading of the sensor, \text{time} the relative reading time \(^4\), and \text{pos} the sensor location \(^5\). The gathered information is then stored in the internal memory of the mobile users’ portable device.

Multi-Resolution Data Management Scheme

In order not to exhaust the limited storage resources of mobile nodes, information is managed according to a Multi-Resolution Data Management (MRDM) scheme, which implements a lossy and location-dependent degrading storage model. Starting from its current position, each mobile node builds a quad-tree hierarchy of the stored information, compressing different tiles along the quad-tree with

\(^4\)This does not require all nodes to be synchronized to a common clock, since we are only interested in storing the reading age, which can be calculated at any time instant by subtracting the timestamped value from the local clock.

\(^5\)Sensors’ position can be programmed at installation phase, or, given a limited sensors communication range, simply assumed as the mobile user position.
different compression ratios. Finer representations will be kept for localized information, while only coarse approximations will be left for remote regions. Hence, each tile is wavelet transformed and compressed separately. Different compression ratios are achieved by varying the wavelet decomposition level, as detailed in Sec. 3. This is briefly summarized in Fig. 6, where the scale ranges from black to white, with black corresponding to the maximum resolution (no compression), and white to a coarser approximation of the original signal (maximum compression), and the mobile user is positioned in the left corner tile.

Clearly, by adopting a tiled management of the stored information, we are trading off the efficiency of the Discrete Wavelet Transform with the level of flexibility that is needed for applying different compression ratios to information deriving from different spatial regions. Treating the entire sensor map as a single image would lead to better (but flat) compression ratios, without the possibility of separately compressing/decompressing portions of the overall SM.

The overall information is stored in the Information Map (IM), which is a $T \times T$ tuple matrix containing all the sensor readings stored by a mobile user. Except for the tile where the mobile user is moving in, the following fields are associated to each tile contained in the IM:

- **Readings Number (RN):** represents the number of sensor readings over which the DWT is calculated;
- **Readings Timestamp (RT):** when reading a sensor, each mobile node stores the reading time, so that it is possible to keep track of the age of each reading. The RT is an average over the timestamp of the sensor readings, and represents the “freshness” of the tile information;
- **Compression Ratio (CR):** is the compression ratio at which the tile is compressed;
- **DWT coefficients (DC):** contains the coefficients of discrete wavelet transform of the tile, in accordance with the compression ratio $CR$ applied.

No compression is applied instead to the tile where the user is moving in. For this tile only, the complete information $(<value, time, pos>)$ on each sensor reading is kept. Indeed, this is the most localized information and, thus, the most important for context-aware services running on users’ portable device.

The Information Map Summary (IMS) is defined as the information map purged from the DWT coefficients DC. Hence, the IMS contains a summary of the information stored in the IM (the DWT coefficients constitute the largest portion of the IM storage), and it will be used, as explained in the following, by mobile nodes for a preliminary exploratory phase of the stored information.

The introduced MRDM scheme requires a Memory Usage (MU) that asymptotically scale as $\Theta(N \log_2 N)$ [6]. This represents the tradeoff between the storage resources and the quality of the information, and can be efficiently mapped to the requirements of the specific application scenario.

Mobile nodes are exchanging their IMs according to a simple P2P protocol, whenever in mutual communication range [6]. When merging the received IM, the most updated information is kept (greater Readings Timestamp and Readings Number).

### 5.2 Performance Evaluation

In order to evaluate the performance of the proposed multi-resolution data management scheme, we have run extensive simulations utilizing a freely available simulation tool [1]. We considered a square playground of size $L = 2.56$ km, over which $128 \times 128$ sensors are uniformly deployed in a grid fashion. A variable number $M$ of mobile nodes are moving over the playground, and are applying the multi resolution data management scheme, for storing, and exchanging at every communication occasion, data read from sensors. The MRDM scheme is implemented by means of Haar discrete wavelet transform, with a $8 \times 8$ structure. Since each tile contains 256 sensors, the MRDM scheme consists of 4 levels of decomposition at most. In our implementation, we applied the MRDM down to level 3, keeping 4 DWT coefficients at the highest compression ratio.

Mobile nodes are moving according to a Random Waypoint (RWP) mobility model [4] with a pausing time uniformly distributed between $(0, 10)$ s, and a speed uniformly distributed in $(10, 15)$ m/s. Nodes’ initial location is drawn according to the steady-state distribution [18], thus reproducing a “perfect simulation”. This eliminates the time needed for reaching a stationary regime.

It is assumed an ideal channel, where mobile nodes can communicate if their mutual distance is less than 75 meters, and sensors are read if located at a distance less than 10 meters. Nodes apply a CSMA/CA for resolving collisions, and communicate at a rate of 11 Mb/s. The Weighted Distortion (WD) is the metric utilized for evaluating the performance of the system, and is defined
as:

\[ WD = \frac{1}{M N^2} \sum_{i=1}^{M} \sum_{j=1}^{N^2} w_{i,j} (R_{i,j} - S_j)^2, \]  

(7)

where \( R_{i,j} \) is the value stored by node \( i \) on sensor \( j \), \( S_j \) is the value of sensor \( j \). \( w_{i,j} = \exp(-\gamma d_{i,j}) \), with \( d_{i,j} \) being the spatial distance of node \( i \) by sensor \( j \) and \( \gamma \) the weighting memory, is an exponentially decaying correcting factor, which accounts for the weight that context-aware applications apply to information: higher for local data, lower for data originating from remote locations (the lower \( \gamma \), the larger the impact of remote data on the WD). The weighting memory is application-dependent, and tunes the level “of information locality” that the specific application scenario can tolerate.

As a first step, we considered a static scenario, with sensor nodes taking values according to a static Random Field \( RF = C + X_0 \), with \( C = 4 \) and \( X_0 \) a sample of the Gaussian random field. In Fig. 7, the weighted distortion \( WD \) is depicted in the case of a variable number of mobile nodes, a correlation length of 1280 and no weighting memory (\( \gamma = 0 \)). It can be observed that a higher number of mobile nodes leads to a better performance of the system. An initial phase is needed for the mobile nodes to first read the sensors, and then diffuse the gathered information. This is reflected in the distortion being constant for the first 100 s, and dropping down afterwards. Fig. 7 presents also the lower bound, calculated as:

\[ LB = D_0 + \sum_{i=1}^{log(N)} (D_i \cdot 3 \cdot 4^{i-1}), \]  

(8)

where \( D_i \) is the distortion at decomposition level \( i \) obtained by averaging 100 Matlab simulations. It is possible to observe that, in the long run, the distortion approaches the lower bound, although it is never reached. This fact can be easily explained as a consequence of missing information: it is very likely that nodes, when leaving (and compressing) a tile, will have only the information on a subset of sensor nodes. Hence, only this partial information will be compressed and determine the distortion of the reconstructed image. Conversely, the \( LB \) is calculated under the assumption that information on all sensors inside the tile to be compressed is available.

We have then introduced a dynamic scenario, with sensor nodes taking values according to:

\[ RF = C + \sin(2\pi T_f t) + X, \]  

(9)

with \( C \) being constant, \( \sin(2\pi T_f t) \) a periodic deterministic temporal component and \( X \) a Gaussian random field (with zero mean, unit variance and an exponentially decaying, in the space domain, covariance) updated every \( T_{RF} \) seconds. For our simulations, we set \( T_f \) to 1 day, \( C \) to 4, the exponential decay factor of the random field to 1280 and the weighting memory \( \gamma \) to 0.001.\(^6\)

\(^6\)By setting \( \gamma = 0.001 \), we are weighting approximately \( \frac{1}{3} \) information originating from sensors that are 600 m away from the current position of a mobile node.

In Fig. 8, the \( WD \) is depicted in the case of 100 and 600 mobile nodes, and a random field varying every \( T_{RF} = 600 \) s. After an initial bootstrap phase, needed for the MRDM scheme to adjust to the average value \( C \), the system is able to track the RF changes. Clearly, given the limited time available for adjusting to new RF values, the WD is constantly fluctuating, with peaks corresponding to RF updates.

Finally, Fig. 9 presents the WD in the case of 600 mobile nodes moving according to the RWP mobility model and a RF update time \( T_{RF} \) of 60, 600 and 6000 s. Clearly, the lower \( T_{RF} \) the less the time available for the system to read diffuse sensor information. This is evidently clear from Fig. 9.
6 Conclusions and Future Work

As Weiser’s vision of an ubiquitous halo of embedded devices supporting our daily activities is taking place, totally new challenges arise for allowing future pervasive services to gain of this opportunity.

In this work, we addressed the scalability problem deriving from the management of context data originating from sensors embedded in the environment. We have defined Multi-Resolution Data Management that takes on the resolution of data for its storage resources. The mechanism has been implemented and evaluated by means of a simulative study. Results show that for a sufficiently large number of mobile nodes, the proposed scheme is capable of tracking the random field changes, while significantly reducing the storage and communication resources required on mobile nodes.

Future work will be devoted to a deeper investigation of the proposed algorithms, and to the implementation of the multi-resolution data management scheme over a real testbed.

References

[1] OMNeT++ discrete event simulation system.