# *Adaptable data models for scalable ambient intelligence scenarios*

Article

Accepted version

A. De Paola, G. Lo Re, F. Milazzo, M. Ortolani

In Proceedings of the International Conference on Information Networking (ICOIN), 2011, pp. 80-85



Publisher: IEEE

http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5723138

# Adaptable Data Models for Scalable Ambient Intelligence Scenarios

Alessandra De Paola, Giuseppe Lo Re, Fabrizio Milazzo and Marco Ortolani
DINFO - Department of Computer Engineering (Univ. of Palermo)
Viale delle Scienze, ed 6. – 90128 Palermo, Italy
Email: {depaola, lore, milazzo, ortolani}@unipa.it



***Abstract*—In most real-life scenarios for Ambient Intelligence, the need arises for scalable simulations that provide reliable sensory data to be used in the preliminary design and test phases. This works present an approach to modeling data generated by a hybrid simulator for wireless sensor networks, where virtual nodes coexist with real ones. We apply our method to real data available from a public repository and show that we can compute reliable models for the quantities measured at a given reference site, and that such models are portable to different environments, so as to obtain a complete, scalable and reliable testing environment.**

***Index Terms*—Ambient Intelligence, Hybrid Simulation, Wireless Sensor Networks, Environmental Data Modeling.**

## I. Introduction and Motivations

Ambient Intelligence is a branch of AI that focuses on adapting the environmental conditions to maximize the user's comfort, and aims to do so transparently by applying methods and ideas borrowed from such fields as pervasive and ubiquitous computing. The underlying assumption is the availability of tools for extensive and timely monitoring of the environment under observation, as well as the construction of predictive models that reliably reproduce the behavior of the physical phenomena of interest. A sensing and communication infrastructure that is increasingly gaining popularity in this context is the Wireless Sensor Network (WSN) technology [1], thanks to its versatility and to the possibility of carrying on limited computations on board of the nodes.

A common approach to assessing the validity of AmI systems is to develop a full functional prototype of the intelligent application, and to actually deploy it into the environment. This is for instance the solution adopted for iDorm [2], a prototype for a student dormitory that allows the simulation of different everyday life activities. AmI applications usually require the creation of predictive models from sensed data; for instance, in the Neural Network House [3] a neural network system was used to forecast future environment state and users' occupancy.

The intrinsic drawback of AmI tests based on actual deployments is that it may prove costly in complex environments, such as entire buildings; moreover, it does not allow to test application scalability, nor to evaluate the application behavior across different configurations.

An alternative approach consists in simulating the whole control loop, from sensing the physical phenomena of interest, to performing artificial reasoning, and finally to modifying the environmental conditions. The Intelligent Home [4], for instance, is a simulated testbed intended as a support for the development of multi-agent systems. However, while the application logic is in general easily reproducible, it is difficult to capture the runtime overall behavior of the whole system. Moreover, early detection of design errors, and fine tuning of critical factors, such as the position and number of sensor nodes in the various areas of the test site, may avoid subsequent, presumably expensive, re-deployment.

Here, we specifically consider WSN-based systems which thus require reliable simulators for the underlying hardware infrastructure. Although several simulation frameworks exist for WSNs which provide controlled, and reproducible environments for tests, they are not guaranteed to deliver fully reliable results, especially when the application logic is heavily sensitive to the actual sensor readings. In order to minimize the difficulties in porting simulated sensor networks to actually deployed systems, it may be advisable to use "real code" simulation tools, that run identical code in simulation and deployment, such as TOSSIM [5], the TinyOS simulator. TOSSIM translates hardware interrupts into discrete simulator events, which are guaranteed to be handled in the correct order; unfortunately this is not sufficient to provide timing guarantees. The authors of [6] have recently proposed a somewhat similar approach that suggests the introduction of "sensor network checkpoints" between simulation and testbed so that rollbacks may be executed to restore the network state to previous conditions. The use of *hybrid* simulators has been proposed as a way to generate reliable, and easily scalable scenarios by the interaction of virtual sensor nodes with actual ones. The use of simulated nodes allows to limit the deployment to just a minimal set of real nodes, which may serve as realistic data model generators to steer the behavior of their virtual counterpart.

This paper presents a proposal for the creation of effective models in the context of a hybrid simulation, in order to ease the design and testing phases of WSN-based AmI applications controlling large sites, such as entire office buildings. In this context, scalability is a fundamental requirement with respect to the number of users and also to the number of sensory devices and of environments under observation.

We have tested our approach to scalable modeling as an addition to our hybrid simulator for WSNs, ATOSSIM [7]. The present work will discuss our method for extracting reliable models for data sensed in one specific environment taken as reference; such models are used as a realistic data source for virtual nodes; moreover we will show how they can be ported to different sites so that a complex scenario may be realistically simulated. Actual deployment of a limited number of sensor nodes is required only for the reference site, so the approach is both scalable and cost-effective.

The remainder of the paper is organized as follows. Section II presents the basic ideas behind hybrid simulation for WSNs, Section III describes our technique for adaptable modeling, Section IV presents experimental results showing the practical feasibility of our approach, and finally Section V reports our conclusions.

## II. Addressing Scalability through Hybrid Simulation

Hybrid simulation allows real world entities to interact with simulated ones, which is extremely useful during the design of large-scale testbeds for AmI applications. Our goal is the creation of a virtual testbed for simulating the perceptive component of an AmI application, whose behavior is strictly related to the actual trend of the physical quantities; the testbed will be formed by a hybrid simulated WSN where a limited set of real nodes is augmented with a larger set of simulated ones.

The hybrid simulator we propose here is aimed to ease testing scale-sensitive AmI applications by allowing for virtual deployment of a large amount of sensor nodes with the freedom to choose the shape and size for the setting, thus providing a simple way to test the application behavior in qualitatively different environments. The adherence to reality is granted by the inclusion of real nodes, whose sensed data are used to generate predictive models for the actual physical quantities; flexibility is also taken into account by adapting the obtained models to simulate the behavior of virtual nodes.

Our hybrid simulator has been realized as an augmented version of TOSSIM, but, unlike this, it allows some of the virtual nodes to be logically bound to real ones; such hybrid (*shadow*) nodes represent the projection of real ones into the simulation and may be regarded as wrappers whose main purpose is to act as interfaces toward their real counterparts, while appearing identical to other virtual nodes from the simulator point of view. The main function of shadow nodes is to collect sensed data from the real world, and to re-route communication from virtual nodes to actual ones.

The coexistence of virtual and real nodes in a hybrid simulation also poses some additional problems; for instance, it is important that the simulation execution time reliably mirrors real execution time, especially in order to preserve causality; some kind of coordination must then be ensured between virtual and real nodes. Our hybrid simulator provides acceptable performances in terms of soft real time constraints, and provides probabilistic end-to-end delay guarantees. Preliminary experiments have been carried on to verify this aspect, and experimental assessment of our simulator in terms of timing accuracy may be found in [7].

In the context of an AmI application, the soft real time constraint is totally acceptable with respect to simulation reliability. Sensing rates in the order of milliseconds or greater are generally totally acceptable for sensor nodes for the kind of physical quantities involved in AmI, so they cannot possibly interfere with the realism of a simulation run.

## III. Models and Metamodels

The proposed AmI hybrid simulator supports the definition of models for environment physical quantities to be monitored by the higher-level AmI application. This allows to simulate sensory readings in places where no sensory devices are actually deployed, thus enabling the simulation of environmental scenarios wider than those actually at researchers' disposal. It is necessary for the adopted models to be tunable, so that they can be instantiated according to real past sensory readings, and adjusted with respect to on-line incoming ones. They also need to be sufficiently generic so that they can be applied to a different monitored area than the one for which they have been built.

Although the overall hybrid simulation mechanism is generic and valid regardless of the specific physical quantity, the particular mathematical model, learning algorithm and mechanism to port a model from a monitored area to another need to be selected, also taking into account the typical observed trends. More specifically, some criteria have to be met for our approach to produce reliable models:

- for each point in space, sensed data are temporally correlated so that it is possible to predict future values with sufficient precision;
- for each time instant, sensed data are spatially correlated, i.e. sensory readings in a given point can be estimated from the readings of close sensors;
- for each monitoring area, the temporal and spatial trend of the considered physical quantity can be derived from those of a similar, close monitored areas.

The scenario described in this work considers such physical quantities as temperature, relative humidity, and light exposure, as they represents typical environmental features considered by AmI applications. In most indoor environments, those quantities typically present a periodic pattern, with a similar trend for all sensor nodes deployed throughout the monitored area; intuitively, this is a good hint that it may be feasible to learn mathematical models able to describe and predict them, and such models may also be reused and applied to other environments with similar physical characteristics as the one considered as reference. Clearly, each physical quantity requires individual modeling, i.e. different models will be independently derived for temperature, humidity, and light respectively; in the following, the function representing one such model will be indicated by $f(x, y, t)$.

The model must be formulated so as to capture information about the physical location of the sensors as well as potential temporal correlation over several measurements. This can be highlighted by explicitly considering a different function $f_{x_i,y_i}(t)$ per each node $i$, which represents the predictive model for the measurements of node $i$ with validity of 24h. We aim to extract a global model providing measurements for each point in the reference site in order to "feed" the virtual sensor nodes with realistic values; we thus have to infer a function $f(x, y, t)$ defined $\forall(x, y) \in S$ from all the $f_{x_i,y_i}(t)$ functions, where $S$ is the reference site.

To this aim, $f(x, y, t)$ may be obtained via simple interpolation of the values predicted by the local $f_{x_i,y_i}(t)$ functions. The method chosen for interpolation should possess the following properties:

1) $f(x, y, t)$ is smooth, i.e. it should be continuous and belong to class $C^\infty$;
2) the interpolated values should fall within the same range as the values predicted at each point in time.

The first property reflects the fact that the monitored physical quantities do not naturally present discontinuities in indoor environments. Even though light exposure may occasionally contradict this assumption in localized points (spots), this is usually not particularly relevant to AmI applications; moreover, strict modeling of this phenomenon would prevent from formulating a general approach valid for all the considered physical quantities at the same time.

The second property aims to ensure that $f(x, y, t)$ does not present extreme values devoid of any physical interpretation; in particular, the property ensures that no unrealistic values are generated in points where no sensors are present.

Because of these considerations, we used a normalized linear combination of the $f_{x_i,y_i}(t)$ functions as a spatial interpolator, as shown by the following equation:

$$f(x, y, t) = \frac{\sum_{i=1}^{N} w_i f_{x_i,y_i}(t)}{\sum_{i=1}^{N} w_i}, \tag{1}$$

where the summation is computed over a set of $N$ nearby deployed sensors, $w_i = e^{-d_i}$, and $d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$, so that each value given by this model is computed by taking into account all the deployed sensors, but with higher importance given to the ones closer to the considered point.

Models learnt from past measurements and current measurements are used to construct a predictor of the various environmental functions. A spatial Interpolator then merges the different predictor functions through Eq. 1 and builds the environmental model $f(x, y, t)$.

The next step in our approach consists in determining how to adapt the environmental models built for one known site (*reference* site) to different areas presenting similar characteristics (*target* sites), as may be the case for different rooms in the same office building. A naïve superimposition of a known model to a different site may not work seamlessly, because of the differences in shape, size, light exposure and so on between the reference and the target areas. We assume here, however, that the considered physical phenomena present some underlying similarity that is preserved across different sites. We will show in Section IV that this is indeed the case for the data considered in our experiments.

In order to port a model from one site to another, it is necessary to determine the set of geometrical parameters that characterize the corresponding transformation; it is convenient to distinguish between two kinds of parameters:

- *intrinsic* parameters, which control the shape of the model functions $f_{x_i,y_i}(t)$ for the reference site;
- *extrinsic* parameters, which control how the model computed for the reference site may be mapped onto a different target site, and basically summarize all the required geometrical transformations (translations, rotation, stretching, and so on) on the models.

### A. Estimating the Intrinsic Parameters

In our view, intrinsic parameters model the shape of the environmental functions representing the observed physical quantities within the area of the reference site. We have chosen to employ fitting methods based on mixture of Gaussians.

A generic function modeling the readings of sensor $i$ may be approximated by a series of $M$ Gaussian functions depending on the previous sensors readings:

$$f_{x_i,y_i}(t) = \sum_{j=0}^{M-1} w_j \mathcal{N}(t|\mu_j, \sigma_j^2), \tag{2}$$

where the spatial coordinates of sensor $i$ are not explicitly indicated. A training set of sensory measurements allows to estimate all the $w_j$, $\mu_j$, $\sigma_j^2$, i.e. the intrinsic parameters for our model of that particular physical quantity, by mean square error minimization [8].

Our approach also takes into account the natural periodicity of environmental phenomena, so the model for a given sensor learnt for a time interval $\Delta t_{old}$ will be used also as predictor for time interval $\Delta t_{new}$ after proper scaling and translation:

$$f_{x_i,y_i}^{new}(t) = \alpha(t) \left[ f_{x_i,y_i}^{old}(t) - \theta(t) \right] + \theta(t). \tag{3}$$

Having computed the intrinsic parameters of the models of the sensor, we are able to implement a predictor for each of them; such predictor should not modify the overall shape of the underlying environmental function, although at the same time it should adapt to transient climatic changes (e.g. a rainy day when the average light exposure is lower than usual, or a sudden drop in temperature).

Moreover, the reliability of the predictions needs to be constantly assessed also in order to adapt the behavior of the sensor nodes; for instance, for the sake of energy saving, the sensing rate may be lowered and transmissions consequently reduced when the prediction model is deemed sufficiently precise, so that it may be provide the required values instead of actual readings.

In our case, we estimate the parameters for the mixture of Gaussians through readings collected in the previous time interval; the pseudocode for iteratively computing the intrinsic parameters for the current time interval is shown in Algorithm 1, where the threshold $\tau$ represents the maximum

**Algorithm 1** Incremental estimate of intrinsic parameters.

**Parameters initialization:**

$T \leftarrow 1$ ▷ sensing rate

$f_{x,y}^{old}(t) \leftarrow \sum_{j=0}^{M-1} w_j \mathcal{N}(t|\mu_j, \sigma_j^2)$ ▷ $\Delta t_{old}$ Gaussian fitting

$S_r \leftarrow \emptyset, S_t \leftarrow \emptyset$ ▷ daily set of readings and relative time instant

$read\_time \leftarrow t_0$ ▷ beginning of readings

$\alpha \leftarrow 1$ ▷ compression

$\theta \leftarrow 0$ ▷ translation

$\tau \leftarrow \tau^\star$ ▷ prediction error threshold

**Read-Prediction cicle:**

**loop**

  **while** $read\_time > current\_time$ not operate **end while**

  $S_r \leftarrow S_r \cup \{z(t)\}$ ▷ update readings set

  $S_t \leftarrow S_t \cup \{t\}$ ▷ update time set

  $\hat{z}(t) \leftarrow \alpha[f_{x,y}^{old}(t) - \theta] + \theta$ ▷ prediction at time t

  **if** $|\hat{z}(t) - z(t)| < \tau$ **then** $T \leftarrow 2 * T$

  **else** $T \leftarrow 1$

  **end if**

  $S_p \leftarrow \emptyset$ ▷ initialize prediction set

  **for all** $t \in S_t$ **do** ▷ compute prediction set

    $S_p \leftarrow S_p \cup \alpha[f_{x,y}^{old}(t_i) - \theta] + \theta$

  **end for**

  $E_D(\alpha', \theta') = \sum_{z \in S_r, \hat{z} \in S_p} \{z_i - \hat{z}_i\}^2$ ▷ square error of predictions

  $(\alpha, \theta) \leftarrow \mathbf{argmin}_{\alpha', \theta'} E_D$ ▷ gradient descent algorithm

  $read\_time \leftarrow read\_time + T$ ▷ next reading time

**end loop**

tolerable prediction error, and it must be specific to any given physical quantity. If the model passes the reliability test (i.e. $\tau > \tau_f$), the sensing rate is halved at each iteration, otherwise it it reset to the default so that the model may be quickly fixed.

### *B. Estimating the Extrinsic Parameters*

Models computed for the reference site must then be extended to be ported to environments with slightly different characteristics. For each of the environmental functions relative to the considered measurements, we assume that the value in at least one specific point $\mathbf{p} = (x, y, t)$ is known; our aim is to compute the transformation that maps the coordinate space of the reference site into a new coordinate space for the target site; each point $\mathbf{p} = (x, y, t)'$ would then be mapped onto $\mathbf{P} = (X, Y, T)'$. Such mapping may be formally defined as $\mathbf{P} = \mathbf{M_S M_\phi p}$, where $\mathbf{M_S}$ depends on the scaling parameters $s_x$ and $s_y$ relative to the spatial dimensions of the reference and target sites, while $\mathbf{M}_\phi$ depends on the parameter $\phi$ controlling the spatial rotation (useful, for instance when considering different light exposure between the reference and the target sites).

The overall transformation matrix may thus be written as $\mathbf{M} = \mathbf{M_S M_\phi}$,and the new environmental function for the target site will thus be given by:

$$F(\mathbf{P}) = \beta(t) \cdot f(\mathbf{p}) = \beta(t) \cdot f(\mathbf{M^{-1}P}), \quad (4)$$

where the $\beta(t)$ weight (generally, time-dependent) stretches

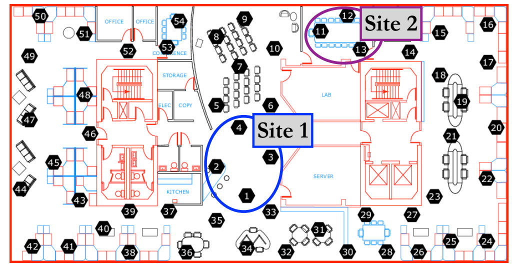


Fig. 1. Maps of the sensor field from Intel Berkeley Research lab [9]; the highlighted areas are those considered in our experiments.

the transformed function to better fit the target environment, and needs to be estimated as will be explained in the following.

In order to estimate optimal settings for the extrinsic parameters we might use a simple, empirical approach, or an automated method based on the measurement of a small set of sensory data from the target environment. For the rotation and scaling parameters the former method was preferred, as support from the human operator seemed reasonable; the required effort is minimal and basically consists only in computing the size and orientation of the target environment, in order to produce the mapping function.

As regards the estimate of $\beta(t)$, on the other hand, we decided to exploit a few sensory readings obtained by placing in the target environment a minimal set of sensor nodes to be used as *probes*. The weight $\beta(t)$ is a proportionality factor that fits measurements predicted by the *Spatial Interpolator* for the reference environment into the probe measurements sensed in corresponding points of the target environment, and basically acts as a time-dependent stretching factor.

The $\beta(t)$ stretching factor is estimated incrementally. Starting from the measured value $F_{X_i,Y_i}(\bar{t})$ for sensor $i$ in location $(X_i, Y_i)$ at time $\bar{t}$ in the target environment, and knowing the mapping matrix $\mathbf{M}$ we infer the prediction for the corresponding point $f_{x_i,y_i}(\bar{t})$ in the reference site. Assuming that the actual measurement in the target site differs from the one estimated through this mapping by a stretching factor $\bar{\beta} = \beta(\bar{t})$, i.e. $F_{X_i,Y_i}(\bar{t}) = \bar{\beta} f_{x_i,y_i}(\bar{t})$, we estimate $\beta(t)$ at various time instants by minimizing the following error function with respect to $\bar{\beta}$:

$$E(\bar{\beta}) = \sum_{\mathbf{P} \in S} \left\{ F_{X_i,Y_i}(\bar{t}) - \bar{\beta} f_{x_i,y_i}(\bar{t}) \right\}^2 \quad (5)$$

where the summation runs over all the probe points $\mathbf{P} = (X_i, Y_i, \bar{t})'$ in the target region $S$.

## IV. Case Study and Experimental Results

We tested our approach on real data from the public repository available from [9], which contains readings collected from 54 sensors deployed in the Intel Berkeley Research lab between February 28th and April 5th, 2004, via a network of Mica2Dot sensor nodes equipped with weather boards measuring temperature, relative humidity, and light.

Temperature is measured in degrees Celsius, humidity is temperature-corrected relative humidity, ranging from 0 to 100%, and light is in Lux (a value of 1 Lux corresponds to moonlight, 400 Lux to a bright office, and 100,000 Lux to full sunlight.) For our purposes, we disregarded 2 nodes with insufficient readings; we analyzed measurements relative to the readings for dates between Mar, 1 2004 and Mar, 7 2004; moreover, in order to compare different quantities we normalized the results to their corresponding effective ranges, namely $18°C$ ($[17°C - 35°C]$) for temperature, 70% ($[20\% - 90\%]$) for relative humidity, and 1600 Lux ($[0 - 1600]$) for light.

Figure 1 depicts the actual settings of the nodes, and highlights the two areas that we chose as representative for a reference and target sites; although they are part of a bigger open-space, in the following we will first prove that for our purposes they may be considered as two separate sites, and that our hypotheses hold, and then proceed to assess the performance of our approach to modeling.

### A. Hypotheses validation

Considering the readings for the three considered physical quantities for the 52 nodes in the database, Figure 2 plots the autocorrelation computed in a time range of 24h and averaged across 7 days. Each plot in the figure shows, for the corresponding quantity, a very high value of autocorrelation in the considered time span, which then may be safely assumed as a base period for temperature, humidity, and light. Not surprisingly, light shows a greater variance with respect to both temperature and humidity (whose plots are almost superimposed and constantly close to 1).

According to the deployment map reported in Figure 1, we selected some sets of close nodes, which we assume to represent our basic sites; in order to have the simulation generalize to a complex scenario, we have assumed that it is possible to choose one of them as a reference site, over which we compute our models that will then be ported to similar sites with no need for additional deployment of sensor nodes.

Here we first validate the assumption that it is possible to individuate such groups, i.e. that physically close nodes indeed sense similar values for each quantity they measure.

Figure 3 reports the results of one experiment carried on the group indicated as "Site 1" (nodes 1, 2, 3, 4); the plots show that each quantity has a very low standard deviation (normalized with respect to the range spanned by the values of each of the considered quantities); note that all values are well below 10%, meaning that, within that group, sensed values across nodes are consistently close to each other.

### B. Model Validation

Considering the groups of sensor nodes highlighted in Figure 1, we will now consider "Site 1" as the reference site, and "Site 2" as the target site; we thus have to show that we can reliably model the sensory readings within the reference site, and that such model is portable, i.e. can provide reliable predictions, to nodes of the target site.

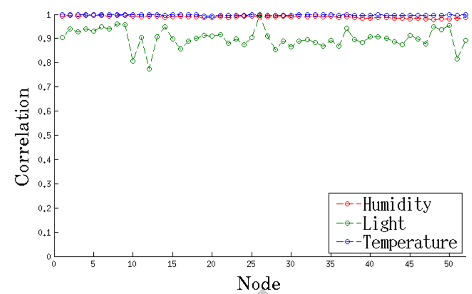


Fig. 2. Autocorrelation for all nodes for temperature, humidity, and light in a time range of 24h.

We started by building a model for each of the nodes of "Site 1" as described in Section III-A. Figure 4 shows the performance of our models as predictors, by plotting the mean value of the absolute error in the considered time range, also showing the maximum and minimum values for $3h$ slots. It is evident that the performances are satisfactory, as the mean error is again constantly lower than 10% for temperature, humidity and light.

In order to assess the validity of our models as estimators of sensor readings in generic locations within the considered test site, we used all available nodes in "Site 1" (nodes 1, 2, 3, 4 ) and performed some tests in a "leave-one-out" fashion, i.e. we built the model with data from $N-1$ nodes and computed the error on readings from the $N^{th}$ node. In our case, the tests were conducted using 3 nodes for training and 1 for testing, for the same time interval as specified above, where the first day was used for training. The absolute error (averaged across all runs of "leave-one-out") was 0.0238 for temperature, 0.0178 for humidity, and 0.0285 for light.

Finally, we assessed the feasibility of porting the models to a different site. We assume that the target site has much fewer actually deployed nodes; in our case, "Site 2" contains nodes 11, 12 and 13, but only node 11 is considered for training, whereas nodes 12, and 13 are used for testing; hence, by using the readings from node 11 as probes, we adapted our models from "Site 1" to the new environment, following the method described in Section III-B. Figure 5 shows the estimation errors for all the considered quantities, for both testing nodes. Again, errors are every low, thus proving that models computed on "Site 1" and adapted to "Site 2" by using probes from just one node do provide reliable estimates in the two considered test points. It must be noted, however, that while errors relative to temperature and humidity measurements are below 25%, and 10% respectively, errors for light measurements are considerably higher; this is due to intrinsically lower correlation for Light, which makes it harder to port the model from one site to another, and more importantly, to the fact that the exact topology of "Site 2" was unavailable to us, so we were not able to fine tune the spatial rotation parameter $\phi$.

## V. Conclusion

This work described an approach to modeling data generated by a hybrid simulator for WSNs in the context of an AmI

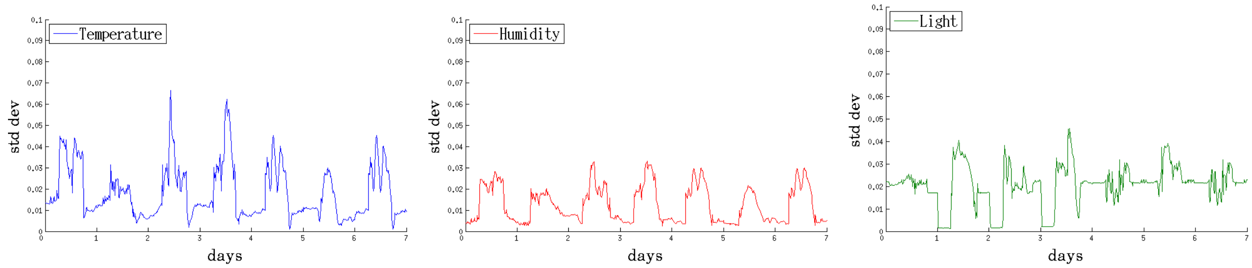


Fig. 3. Assessment of similarity across spatially close measurements.

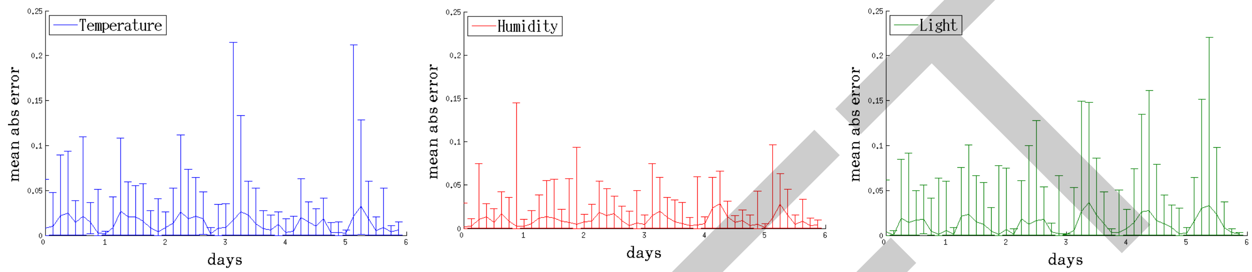


Fig. 4. Assessment of the model as a predictor.

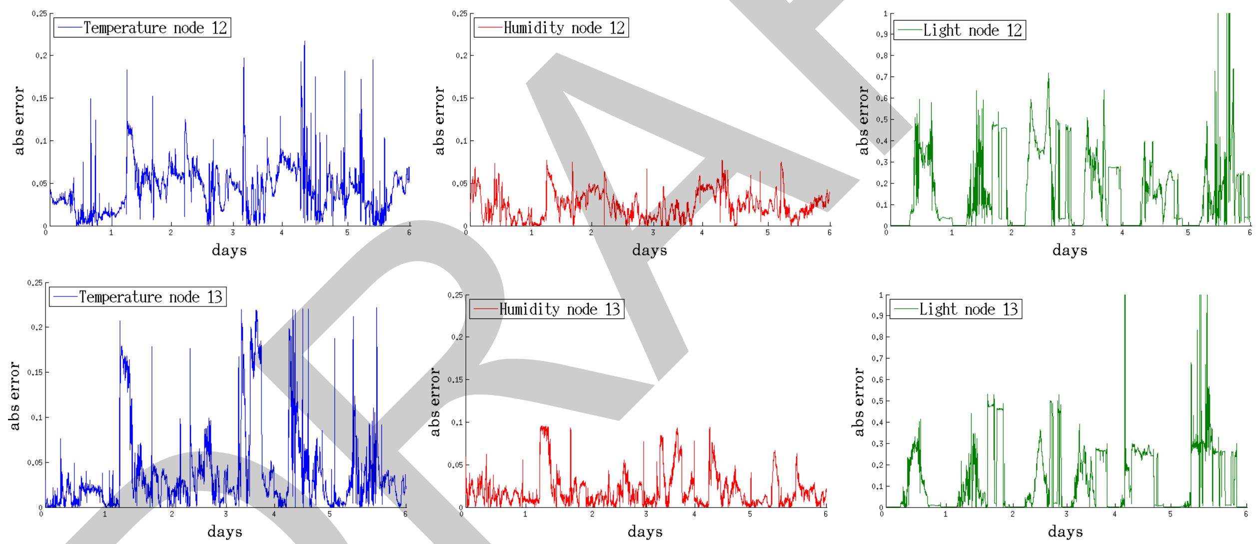


Fig. 5. Estimation error when porting models to a target site. (The y-axis range for Light is $[0, 1]$, as opposed to $[0, 0.25]$ used for Temp. and Hum.

scenario. We addressed issues related to the assessment of the produced predictive models, as well as to their generalization to unknown environments in order to show that it is possible to generate scalable and reliable simulations for the sensory layer of an AmI system. Our experiments, conducted on a publicly available repository, show the validity of the proposed approach.

## References


[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communication Magazine*, vol. 40, no. 8, pp. 102–114, August 2002.

[2] A. Holmes, H. Duman, and A. Pounds-Cornish, "The iDorm: Gateway to heterogeneous networking environments," in *International ITEA Workshop on Virtual Home Environments, Paderborn, Germany (February 2002)*, 2002, pp. 1–8.

[3] M. Mozer, "The Neural Network House: An Environment hat Adapts to its Inhabitants," in *Proc. AAAI Spring Symp. Intelligent Environments*, 1998.

[4] V. Lesser, M. Atighetchi, B. Benyo, B. Horling, A. Raja, R. Vincent, T. Wagner, P. Xuan, and S. Zhang, "The intelligent home testbed," in *Autonomy Control Software Workshop*, 1999.

[5] P. Levis, N. Lee, M. Welsh, and D. Culler, "Tossim: Accurate and scalable simulation of entire tinyos applications," in *the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003)*. ACM, November 2003.

[6] N. Finne, J. Eriksson, N. Tsiftes, T. Voigt, A. Dunkels, and F. Österlind, "Sensornet checkpointing: enabling repeatability in testbeds and realism in simulations," in *Proceedings of EWSN 2009: 6th European Conference on Wireless Sensor Networks*, Cork, Ireland, 2009.

[7] A. Lalomia, G. L. Re, and M. Ortolani, "A hybrid framework for soft real-time wsn simulation," in *13th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications*, October 2009.

[8] A. Dempster, N. Laird, D. Rubin, *et al.*, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[9] S. Madden, "Intel lab data," http://db.csail.mit.edu/labdata/labdata.html.