2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

Integration of WSN, IoT and Cloud Computing in Distributed Monitoring System for Aging Persons in Active Life

Oana Chenaru, Viorel Mihai, Dan Popescu, *Member, IEEE,* and Loretta Ichim, *Member, IEEE*

Abstract—**The paper presents a multi-level architecture for AAL applications, separating sensor, fog computing and cloud processing levels and assigning a specific role for each of them. This increases efficiency in data acquisition, integration, response time and long-term analysis. The system was designed scalable, being capable to integrate data from different IoT and WSN networks and provide support for distributed application implementation. The paper details the functionality of each level and tools used in the implementation of a body sensor network and indoor environmental monitoring of a single patient. Indoor comfort and temperature variation are used as examples for the user application stored at the fog computing level. An ARIMA model is used at the cloud level for temperature data prediction.**

I. INTRODUCTION

In the medical sector, a great attention is given to improving the quality of life for elderly people wishing to live independently at home, without the constant presence of a career by their side. It is important for these people to have confidence in a prompt reaction of support, in case of need, from careers, medical staff or relatives, without affecting their private life. Typical applications refer to monitoring a person's actions and health status to identify and alert in case of dangerous situations, providing support in following a medical treatment, or access to different services like groceries or cleaning. Along with the constant increase of Wireless Sensor Networks (WSN) and Internet of Things (IoT) solutions we can develop new applications to support this behavior while minimizing the medical risks involved.

Intelligent monitoring networks, with reconfiguration, self-diagnosis and self-learning capabilities have been approached in the last years in both scientific and industrial sectors. From the technical point of view, there are several proposals on the quantity and type of measurements, the processing levels and the services or functions needed in an Ambient Assisted Living (AAL) application. There are many papers which focus only in processing data from (infra-red) video cameras [1] or only from accelerometers [2], but the information obtained covers a limited set of applications. This set includes the identification of emergency situations (like falling), characterization of specific behavior or daily routines to identify possible deviations. In addition to this, video surveillance is perceived as a violation of personal space and removes the benefits of an independent living [2]. The use of this information alone will mostly alert regarding a possible or already critical medical condition and call for fast and specific actions from the caregivers of medical staff. It does not support a preventive approach.

A more appropriate architecture in AAL applications is to include a larger set of sensors, monitoring elderly's vital signs (heart rate, body temperature, and brain activity besides the accelerometer) combined with indoor and outdoor environmental measurements. Fast and accessible self-alerting sensors should also be included. This will allow advanced data processing in a contextual analysis, as the vital signs normal levels can be affected by the environmental conditions or previous activities.

Such an architecture is illustrated in Fig. 1. It combines WSNs used for sensors data acquisition and transmission with IoT for making this data available in a remote server or cloud site and accessible by involved actors. This approach promotes centralizing all data storage and processing in the cloud, requiring a high number of resources, and increasing inefficiency in case fast actions are needed [4]. The main disadvantage of this approach is the increase of collected and transmitted data, leading to low autonomy of wireless devices, increased latency, and increased workload at the centralized level.

Current architectures that analyzed AAL applications from a context-aware perspective use different algorithms as a possible implementation solution. In [3] the authors chose rule-based reasoning for the implementation of the context model. Also starting from an ontology model, in [5] the authors use a probabilistic reasoning that enables model adaptation at runtime. Other solutions like video-based monitoring has been extensively analyzed because of the high precision in activity recognition [6]. The main drawback in these solutions is the possibility to detect only ongoing situations, without having the possibility to apply long term prediction algorithms. A step forward in this direction is provided in [7], where the authors propose a context-aware model able to predict future states from current and past observations.

*This work has been funded by University Politehnica of Bucharest, through the "Excellence Research Grants" Program, UPB – GEX 2017. Identifier: UPB- GEX2017, Ctr. No. 25/2017 (INOVATION).

Oana Chenaru is Lecturer with the Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Splaiul Independentei nr. 313 Sector 6, București, 060042, Romania (phone: +4021 402 91 05; e-mail: oana.chenaru@gmail.com).

Viorel Mihai is PhD student with the Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Splaiul Independentei nr. 313 Sector 6, București, 060042, Romania (phone: +4021 402 91 05; e-mail: mihai.tc83@gmail.com)

Dan Popescu is Professor with the Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Splaiul Independentei nr. 313 Sector 6, București, 060042, Romania (phone: +4021 402 91 05; e-mail: dan_popescu_2002@yahoo.com).

Loretta Ichim is with Faculty of Automatic Control and Computer Science, Politehnica University, Bucharest, Romania (e-mail: loretta.ichim@upb.ro).

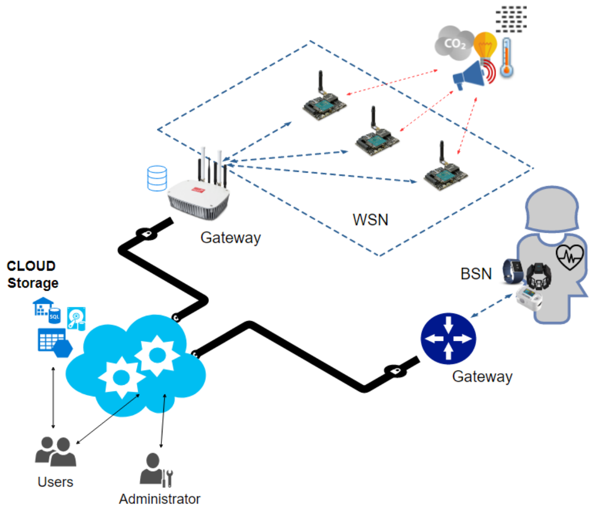

Figure 1. Example of a typical AAL architecture

The presented papers focus on multi-level sensor architectures where processing takes place only at the upper level. This increases the data transmission congestion and delay at the upper level and the forces processing on large quantities of data. Increased efficiency in data transmission was obtained by adopting multi-level processing models, where initial processing at the local level enable a reduced workload at the upper levels [8]. It was also shown that implementation of simple prediction functions at the sensor level increases the system autonomy by optimizing the power consumption [10].

Our work describes a new approach on data transmission that uses Fog Computing concepts to define a multi-level processing (MLP) architecture. We propose a local preliminary processing of time-sensitive data and use of a cloud application for long-term analysis that will be responsible for data storage and context characterization. Our aim is not only to covert data into relevant information, but also to obtain environmental and behavioral models. We can then use the cloud level for historical data storage and we can analyze if there is a tendency of change in environmental quality, patient's health or occurrence of changes in patient's behavioral pattern.

The rest of the paper is structured as follows. Section II describes the proposed multi-level processing architecture, empathizing their role and implemented functions. Section III details the application setup and the tools used for the implementation. It also shows the obtained results for air quality monitoring in an indoor living space and how that information is presented to the user for easy understanding. Section IV concludes this paper and provides directions for future work.

I. MLP Architecture for AAL Applications

In this section we describe in detail our proposed MLP architecture. We defined three levels, with specific functions implemented at each one: sensor processing level (SPL), data analysis level (DAL) and cloud prediction level (CPL) (Fig. 2). The aim is the development of an integrated and intelligent context characterization and change prediction framework. It should be able to provide detailed information on environmental conditions, ongoing actions of the monitored person and predict his needs for support. This means it will enable not only alerting in case of an ongoing critical situation, but also warning in case of changes in the environmental parameters, early detection of symptoms and chronic disease prevention.

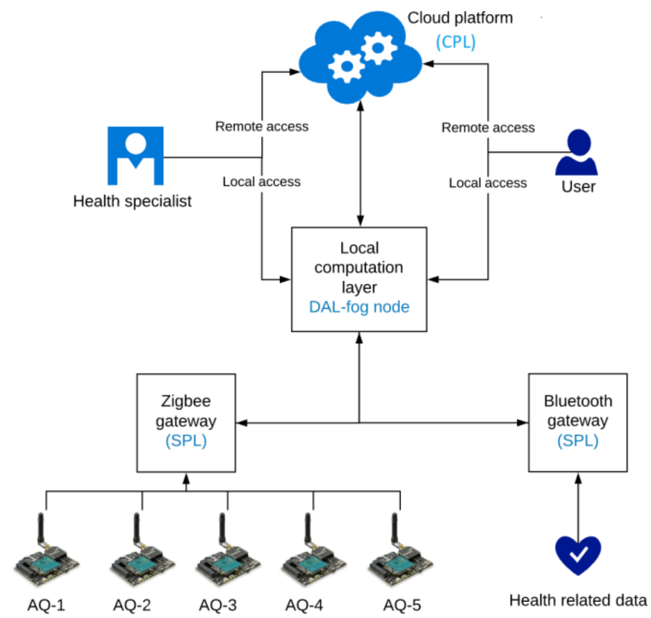

Figure 2. Proposed MLP architecture

A. *Sensor Processing Level (SPL)*

SPL interacts directly with the environment and it is responsible for data acquisition from a limited set of sensors, in a single context. A BSN node, for example, can receive data from both temperature and heart rate, or an environmental node can receive data regarding room temperature, pressure, and humidity. In a mesh WSN network, it is represented by the base station, capturing data from the entire network. SPL temporarily stores gathered, performs data validation and scaling and sends it to DAL at predefined intervals. Data packets communication protocol follow the rules defined in the implementation of the WSN or IoT network.

Optimization of data transmission is achieved at this level by implementing a simple data estimation algorithm that evaluates the average of last values in a limited data set and compares current value with it, as a predicted value. If the read value corresponds with the predicted value, considering a certain threshold, then the data is not transmitted to the next level. An alive message ensures DAL of an active communication link. The dimension of the data set and threshold used in the prediction algorithm depend on the characteristics of each measured parameter.

B. Data Analysis Level (DAL)

DAL has typical role of data aggregation and transmission to cloud, but also implements fog computing mechanisms to increase AAL application efficiency. DAL gathers data received from SLP and applies simple analysis and rule-based reasoning methods for fast reaction to critical situations.

At each sampling period t, a new value for each parameter is stored. Data analysis regards obtaining information from real-time data, or from analysis of short-term data. The mean value can be computed for each parameter of the air quality or BSN data set to illustrate the central tendency. Considering a process data set $X = [X_1, X_2 \ldots X_n], X \in \mathbb{R}^n$ of n samples, we can define the mean value with the following equation [9]:

$$\overline{X}_n = \frac{\sum_{i=1}^{n} X_i}{n} \quad (1)$$

For continuous real-time evaluation of the mean value, we identified the recursive formula:

$$\overline{X}_{n+1} = \frac{\sum_{i=1}^{n+1} X_i}{n+1} = \overline{X}\frac{n}{n+1} + \frac{X_{n+1}}{n+1} \quad (2)$$

The standard deviation σ_n describes the variability of the data values. It is computed based on the variables sample mean with (3):

$$\sigma_n = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \overline{X}_n\right)^2} \quad (3)$$

To identify the recursive computing method for σ, we defined S_n as the variable component of the standard variation:

$$S_n = \sum_{i=1}^{n}\left(X_i - \overline{X}_n\right)^2 \quad (4)$$

Eq. (3) becomes:

$$\sigma_n = \sqrt{\frac{1}{n-1} S_n} \quad (5)$$

S_{n+1} can be expressed depending on S_n as:

$$S_{n+1} = \sum_{i=1}^{n}\left(X_i - \overline{X}_n + \overline{X}_n - \overline{X}_{n+1}\right)^2 + \left(X_{n+1} - \overline{X}_{n+1}\right)^2 \quad (6)$$

Considering that:

$$\sum_{i=1}^{n}\left(X_i - \overline{X}_n\right) = 0 \quad (7)$$

the recursive formula for S_n was identified as:

$$S_{n+1} = S_n + (X_{n+1} - \overline{X}_{n+1})^2 + n(\overline{X}_n - \overline{X}_{n+1})^2 \quad (8)$$

To increase load efficiency at cloud level data we want to send data only when variation of parameter X at current step $n+1$ exceeds the previous received value according to (9):

$$X_{n+1} - \overline{X}_n \geq \sigma_n \quad (9)$$

The reasoning uses data received from SPL, a definition of actions based on conditions, set of limits that the parameters must comply with and a set of if-then reasoning rules. Each parameter is evaluated against specific limits for risk situation identification. These limits are adjustable by CPL to fit the behavior of each monitored person. The same way, parameters in the reasoning rules can be adjusted by the superior level. For example, if the variation of a BSN parameter, like heart-rate, has a growing rate higher than a specified value max_{HR}, while the context parameters follow a different or a constant variation, than an alarm is triggered.

A representation example of the DAL reasoning is illustrated in Fig. 3. In combines parameter specific analysis with quality indicators that take into consideration several parameters and reasoning that may include one or more rules evaluation. Mean and standard deviation values are updated at the end of parameter analysis. A cloud buffer is used to optimize data transmission to the CPL level, so that larger data sets are forwarded at a tc interval.

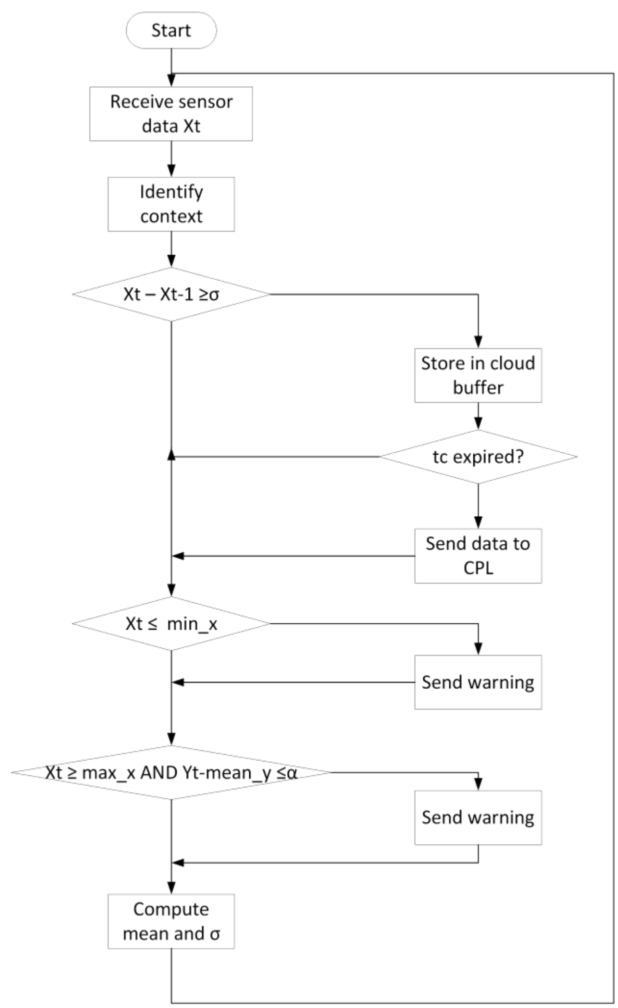

Figure 3. Flowchart example of a parameter analysis at DAL level

C. Cloud Prediction Level (CPL)

CPL represents an advanced processing level that implements environmental parameters prediction and generates alarms or warnings in case they are expected to exceed recommended limits. We perform a cross-correlation analysis between temperature and humidity to identify possible sensors malfunction.

The ARIMA (Autoregressive Integrated Moving Average) model [11] was implemented for each time series modeling. For the implementation of the model, its components were defined as a sum of autoregressive terms (AR) with moving average terms (MA) [11]. Assuming that each variable is a weighted linear combination of p past observations, adding a random error at each step. The mathematical representation for this aspect, known as *AR(p)* model, is:

$$X_n = c + w_1 X_{n-1} + w_2 X_{n-2} + \ldots + w_p X_{n-p} \tag{10}$$

The moving average model, *MA(q),* can be used to represent a time series as sum of q weighted past errors:

$$X_n = \varepsilon_n + \mu_1 \varepsilon_{n-1} + \mu_2 \varepsilon_{n-2} + \ldots + \mu_p \varepsilon_{n-q} \tag{11}$$

The *ARMA(p, q)* model is obtained by summing the *AR(p)* and *MA(q)* models:

$$X_n = c + \varepsilon_n + \sum_{i=1}^{p} w_i X_{n-i} + \sum_{i=1}^{q} \mu_i \varepsilon_{n-i} \tag{12}$$

Considering the lag operator with the property that $LX_n = X_{n-1}$, the *ARMA(p, q)* model can be rewritten as:

$$(1-\omega_1 L)(1-\omega_2 L)\cdots(1-\omega_p L)X_n =$$

$$= (1-\varphi_1 L)(1-\varphi_2 L)\cdots(1-\varphi_p L)\varepsilon_n \tag{13}$$

ARIMA(p, d, q) is a generalization for this model applicable also to non-stationary data. It can be obtained by representing the *ARMA(p, q)* model considering the lag polynomials L:

$$\left(1-\sum_{i=1}^{p} w_i L^i\right) X_n = \left(1+\sum_{i=1}^{q} \mu_i L^i\right)\varepsilon_t \tag{14}$$

Assuming that L has a unit root factor of multiplicity d, eq. (11) can be rewritten to define the *ARIMA(p, d, q)* model as:

$$\left(1-\sum_{i=1}^{p'} w_i L^i\right)(1-L)^d X_n = \left(1+\sum_{i=1}^{q} \mu_i L^i\right)\varepsilon_t \tag{15}$$

where $p' = p - d$.

To evaluate the performance of the forecasted model, the initial data set will be separated into a training set and an evaluation set. A comparison will be performed between the values from the evaluation set with the ones predicted using the ARIMA model. Performance quantification is obtained by computing the mean squared error (*MFE*) as:

$$MFE = \frac{1}{n}\sum_{i=1}^{n}(X_i - f_i)^2 = \frac{1}{n}\sum_{i=1}^{n}\varepsilon_i^2 \tag{16}$$

where ε_i is the prediction error at step i, f_i is the forecasted value and X_i is the actual value. This measure captures the average deviation of the forecasted values from the actual ones and shows the direction of the error. In a good forecasting model MFE → 0.

II. System Implementation and Results

An example of implementation considering the proposed architecture is illustrated in Fig. 4.

A. SPL application

The SPL level included one ZigBee gateway and one Bluetooth gateway. The Zigbee gateway collected and performed initial processing on all environment related data. The examined area was divided in five points of interest, each corresponding to different expected air quality levels. Following this consideration, the sensors measurements were more relevant in terms of local data collection. Four of the monitoring points were situated indoor and one was outdoor. The outdoor measurements provided relevant data to involved actors regarding indoor/ outdoor environment discrepancy.

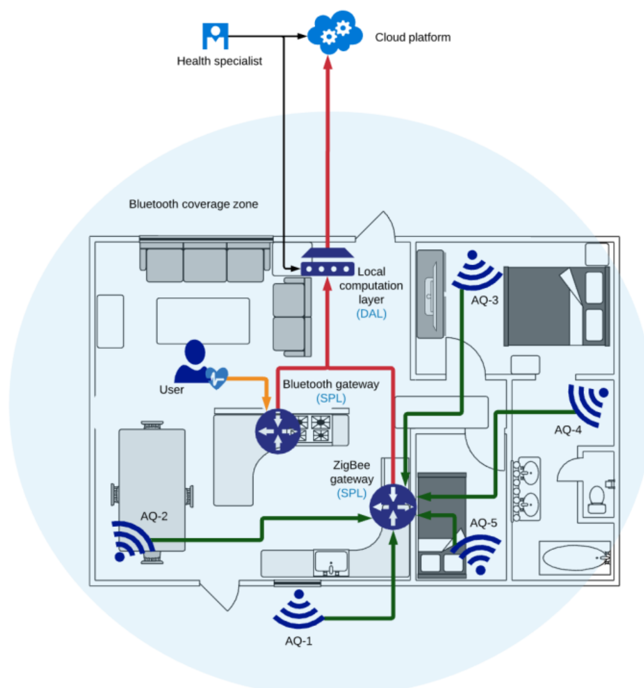

Figure 4. Integration of WSN, BSN and cloud computing in an AAL application

One air quality WaspMote node was considered for each zone of interest (noted from AQ-1 to AQ-5). WaspMote was developed by Libellium [12] and, in our configuration, it included six sensors: temperature, humidity, CO2, NO2, VOC and CO. The WaspMotes where deployed for maximum area coverage. Data was transmitted to the central

Zigbee gateway, responsible for WSN management and data transmission to a database on a local server.

For the BSN network Shimmler instrumentation [13] was considered. All health-related data was collected via Bluetooth at DAL level. The architecture allowed body monitoring for one inhabitant. As future work, the solution will be scaled to multiple persons monitoring. This sample implementation considered suitable coverage area for the Bluetooth network.

B. DAL implementation

The server performs DAL processing as presented in section III. It can send warnings or alert messages to a list of caregivers and health specialists. The local functionality can be accessed through a web-based graphic user interface (GUI). It allows users to monitor all local information. This functionality was developed in Node-red [14], a user-friendly graphic programing environment. It can provide backend data aggregation functionality but also frontend dashboard support. One key feature is the cross-platform capability. The GUI can be accessed via mobile or personal computer. Fig. 5 presents a graphic view of measured data, temperature and relative humidity. The comfort zone can be modified according to the inhabitant's preferences or to their medical condition. Similar visualizations are available for the other investigated values.

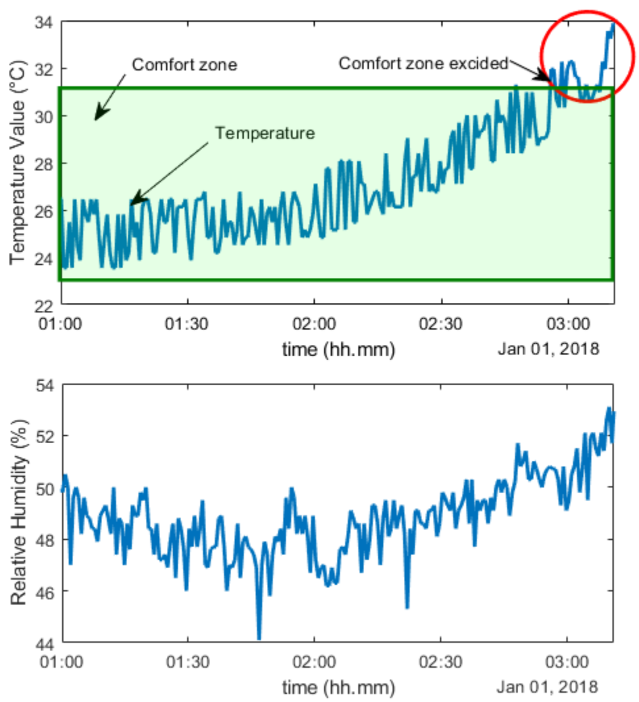

Figure 5. Temperature and umidity measurements

Additional functionalities are available, as shown in Fig 6. The GUI present the comfort level according with the ASHRAE Standard 55 using the PMV method [15]. The implementation of this functionality was based on CBE Thermal Comfort Tool [16]. It provides a method to quantify the thermal comfort domain considering 6 factors: air temperature, mean radiant temperature, relative humidity, air speed, metabolic rate and clothing insulation. The result was overlaid with measured data.

C. CPL implementation

CPL modules were implemented using Microsoft Azure. Each 30 minutes, data stored in the local DAL buffer was requested by the cloud for the historical storage and predictive analysis. Data storage used SQL databases.

The prediction was implemented in "Functions" module available in Microsoft Azure, using Python because of existing resources for data management, mathematical analysis and trend display. Computing the prediction model took place after receiving each new data set. It implemented in the statsmodels library the functions needed for executing and evaluating the ARIMA model.

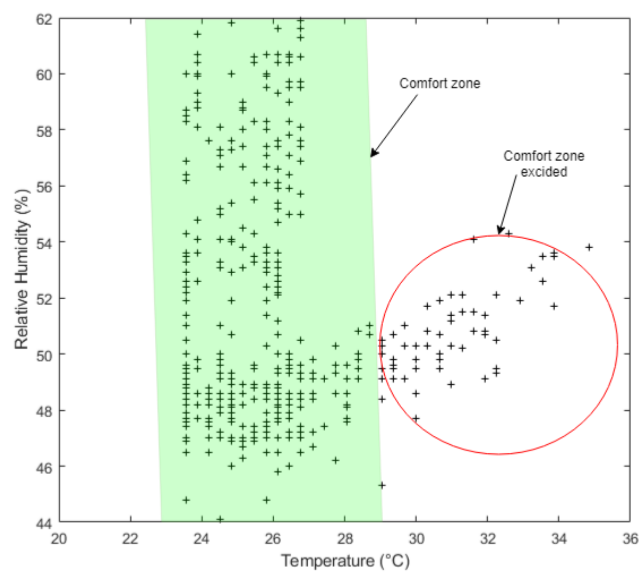

Figure 6. Temperature and umidity – comfort level acording with the thermal environment and is assessed by subjective evaluation (ANSI/ASHRAE Standard 55)

The execution time and the efficiency of the prediction depend on the stationary behavior of the data set, the lag order, the degree of differencing and the moving average order. Fig. 7 shows the obtained results in the initial set for temperature prediction. Blues represents the real data, and red the predicted values. Our initial data set used 50 values for model training and 300 values for computing predicted values. We used an (5, 1, 0) model and obtained a MFE of 1.518. The model parameters were chosen based on the analysis of the temperature autocorrelation graph and non-stationary characteristics.

Increasing the data set to 500, on the same model, lead to a small improvement in the prediction error, with a MFE of 1.261, but the execution time increased. Results from the prediction models are used to send warnings to DAL, which will, in turn, inform respective caregivers. Results of predicted values are stored in the database and are available to end-users as dashboards.

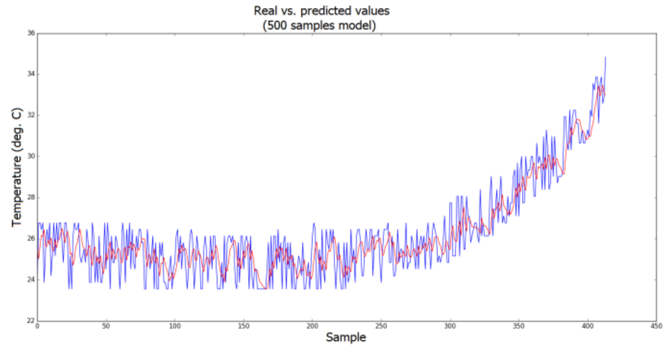

Figure 7. Temperature prediction results

III. Conclusions

The paper presented a novel three-level architecture that integrates WSN, IoT and cloud computing capabilities in an AAL application. At bottom, the SPL layer clusters all sensors. The DAL level is considered as a local interface between sensors, cloud platform and the involved actors such as users or health specialists. The DAL server backend consists of local algorithms for data aggregation and event triggered alarms. The graphic user interface available at this level allows real time monitoring of data gathered from air quality and health sensors. The cloud level, CPL, allows data storage, data correlation and prediction. It generates useful information for users and specialists. Using this layer, the health specialist can provide an anticipative reaction to health issues considering also the air quality level. The paper provides an implementation solution for advanced AAL application implementation. The novelty is not only in the different processing levels, but also in the implementation that provides solutions for ensuring a fast reaction speed in case of emergency, together with the access to complex prediction algorithms.

As future work we intend to scale this system at multiple levels. First, we need to ensure multiple data connection for BSN sensors to provide health monitoring to more than one person. Close to this, the algorithms will allow personalized health information for each individual, correlated with specific air quality data from his relative localization in the considered area. Secondly, we intend to scale this system in the concept of smart house philosophy. The DAL level can gather relative large quantity of data from multiple sources in a wired or wireless network. Also, the developing environment, node-red, allows fast integration and possibility to personalize the application according with user's needs. In the end, this architecture will be an interface between the final user/ users and specialist in different domain (health, economy etc.). By leveraging more complex data aggregation, correlation and prediction algorithms, we intend to support a better quality of life for the inhabitants without compromising privacy regulations.

Acknowledgment

This work has been funded by University Politehnica of Bucharest, through the "Excellence Research Grants" Program, UPB – GEX 2017. Identifier: UPB- GEX2017, Ctr. No. 25/2017 (CAMIA).

References

[1] G. Castellano, S. D. Villalba, A. Camurri, "Recognising human emotions from body movement and gesture dynamics", ACII'07 Proceedings of the 2nd International Conference on Affective Computing and Intelligent Interaction, pp.71–82, 2007.

[2] J.Ye, S. Dobson, S. McKeever, "Situation identification techniques in pervasive computing: A review", Pervasive and Mobile Computing, pp. 33 – 66, 2012.

[3] F. Paganelli, D. Giuli, "An ontology-based system for context-aware and configurable services to support home-based continuous care", IEEE Transactions on Information Technology in Biomedicine, vol. 15, pp. 324– 333, 2011.

[4] S.A. Moraru, L. Pemiu, D. Ungureanu, F. Sandu, A. Moşoi, and D.M. Kristaly, "Integrating wireless sensors into cloud systems for ambient assisted living", in IEEE, Control and Automation (MED), 2017 25th Mediterranean Conference on, ISSN: 2473-3504, pag. 1106-1112, INSPEC Accession Number: 17046337, IEEE Xplore.

[5] F. Ongenae, M. Claeys, T. Dupont, W. Kerckhove, P. Verhoeve, T. Dhaene, F. De Turck, "A probabilistic ontology-based platform for self-learning context-aware healthcare applications", vol. 40, pp. 7629–7646, 2013.

[6] Y. Zhu, N.M. Nayak, A.K. Roy-Chowdhury, "Context-aware activity recognition and anomaly detection in video", IEEE J. Sel. Top. Signal Process. 7, pp. 91–101, 2013.

[7] Abdur Forkan, Ibrahim Khalil, Zahir Tari, Sebti Foufou, Abdelaziz Bouras, "A context-aware approach for long-term behavioural change detection and abnormality prediction in Ambient Assisted Living", in Pattern Recognition, vol. 48, pp. 628 – 641, 2015.

[8] K. Wang, Y. Shao, L. Shu, G. Han, C. Zhu, "LDPA: A Local Data Processing Architecture in Ambient Assisted Living Communications", IEEE Communications Magazine, 2015.

[9] M. P. Barde, P. J. Barde, "What to use to express the variability of data: Standard deviation or standard error of mean?", Perspectives in Clinical Research, Vol 3(3), pp. 113–116, 2012.

[10] B. Risteska Stojkoska, K. Trivodaliev, "Enabling Internet of Things for Smart Homes Through Fog Computing", 25th Telecommunication Forum (TELFOR), 2017.

[11] G.P. Zhang, "Time series forecasting using a hybrid ARIMA and neural network model", Neurocomputing Journal, Vol. 50, pp. 159–175, 2003.

[12] Libellium product guide, available online at: http://www.libelium.com/products/plug-sense/ (accessed 20.02.2018).

[13] Shimmersensing product guide, available online at: https://www.shimmersensing.com, (accessed 20.02.2018).

[14] NodeRed IDE, available online at: https://nodered.org/ (accessed 20.02.2018).

[15] Thermal Environmental Conditions for Human Occupancy, [*ANSI/ASHRAE Standard 55-2013]* ISSN 1041-2336

[16] Hoyt Tyler, Schiavon Stefano, Piccioli Alberto, Cheung Toby, Moon Dustin, and Steinfeld Kyle, 2017, CBE Thermal Comfort Tool. Center for the Built Environment, University of California Berkeley, http://comfort.cbe.berkeley.edu