



HAL
open science

MPaaS: Monitoring values Prediction as a Service for energy consumption optimization purpose

Ahmed Jemal, Marwa Hachicha, Riadh Ben Halima, Ahmed Hadj Kacem, Khalil Drira, Mohamed Jmaiel

► **To cite this version:**

Ahmed Jemal, Marwa Hachicha, Riadh Ben Halima, Ahmed Hadj Kacem, Khalil Drira, et al.. MPaaS: Monitoring values Prediction as a Service for energy consumption optimization purpose. 2013. hal-00872716v1

HAL Id: hal-00872716

<https://hal.science/hal-00872716v1>

Preprint submitted on 15 Oct 2013 (v1), last revised 22 Mar 2014 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MPaaS: Monitoring values Prediction as a Service for energy consumption optimization purpose

Ahmed JEMAL^{1,2,3} Marwa HACHICHA¹ Riadh BEN HALIMA^{1,2,3} Ahmed HADJ KACEM¹ Khalil DRIRA^{2,3}
Mohamed JMAIEL¹

¹University of Sfax, ReDCAD Laboratory, B.P. 1173, Sfax-Tunisia

²CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

³Univ de Toulouse, LAAS, F-31400 Toulouse, France

jmlhmd@gmail.com, marwahachicha@gmail.com, riadh.benhalima@enis.rnu.tn, ahmed.hadjkacem@fsegs.rnu.tn, khalil@laas.fr, mohamed.jmaiel@enis.rnu.tn

Abstract— The Wireless Sensor Networks (WSNs) deployment introduces many issues and challenges mainly in terms of energy independence. In this context, we adopted a self-adaptation strategy composed of four steps (Monitor, Analyze, Plan and Execute) to manage Quality of Service (QoS) [1]. This paper focuses on the first step which consists in monitoring and sending QoS values such as the value of power remaining in the battery of each sensor. We notice that the transmission process is very expensive in terms of energy and reduces the battery lifetime. In this work, we propose a probabilistic approach that estimates a part of these QoS monitoring values and therefore economizes their transmission energy and extends the sensor battery life. Our approach is based on the hidden Markov chain and the fuzzy logic and composed of two steps: (i) learning which allows apprehending the WSNs behavior and (ii) predicting which estimates QoS monitoring values. A WSN application deployed in a datacenter is studied as an illustration. The carried out experiments over AZEM WSN simulator show its feasibility and efficiency.

Keywords— Wireless sensor network; probabilistic approach, hidden Markov chain; value of monitoring; prediction.

I. INTRODUCTION

WSNs are composed of small wireless nodes that monitor and control the environment. Then, the collected information is transmitted to the base station that can be a computer or a mobile smart phone to process their analysis and exploitation [2].

Sensor networks can sense a wide range of measurement such as temperature, humidity, pressure, detection of movement. Consequently, the WSNs overcome several field of application such as medical applications, environmental, datacenter, smart building and military.

However, the architecture of WSN is influenced by several constraints especially the energy consumption due to the fact that nodes are battery powered and generally it is impossible to recharge or change them given that sensors are deployed in a large scale and in inaccessible areas.

Sensor node's lifetime depends on battery lifetime. Therefore, the energy is the most precious resource that affects the sensors lifetime and thus the sensor network.

An autonomic enabled-architecture according to the IBM control loop [3] should be taken in order to adapt WSNs applications and increase the WSNs lifetime. The autonomic loop is composed of four steps, namely: Monitoring which collects QoS parameters values, Analysis which checks the gathered values and identifies a possible degradation of the network state, Planning which plans actions to adapt the WSNs behavior and Execution which enforces them. The first step relies on monitors able to transmit periodically measurements such as the value of power remaining in the sensor battery. The transmission of these QoS values consumes a great part of the residual battery energy. Optimizing this process enable to reduce the energy consumption of sensors and maximize their lifetime and therefore the life of WSNs applications.

In this paper, we focus on predicting a part of these values. Our challenge is to maintain lifetime duration of WSNs as maximum as possible. We propose MPaaS: Monitoring values Prediction as a Service which uses the hidden Markov chains and the fuzzy logic to optimize the energy consumption. In MPaaS approach, we start by learning WSNs behavior which allows us to predict a part of the QoS monitoring values. It will avoid the transmission process of the predicted values, and therefore increase the sensor butterfly life.

To illustrate the application of our approach, we used the WSNs datacenter monitoring case study. Our purpose is to monitor the battery of each sensor and estimate the value of power remaining in order to increase the whole application lifetime.

The reminder of the paper is organized as following. In section 2, we briefly introduce the related work, the basic concept of hidden Markov chain and the fuzzy logic. In section 3, we detail our approach and explain how to rationalize energy consumption. Section 4 describes the case study, and the experimentation results. We evaluate the performance of our approach in section 5 while calculating the prediction error. The last section concludes the paper.

II. RELATED WORK

In this section, we examine the state of the art about the energy prediction model in WSN. Then, we present the main concepts related to the hidden Markov chain and the fuzzy logic.

A. Energy predicting model in WSN

The work proposed by [4] is an estimation model based on the hidden Markov chain (HMC) to predict the energy level of a sensor node. The entire process contains two main parts: a first part to train the protocol-specific HMC via the *Baum -Welch* algorithm and a second part to predict energy levels via *Viterbi* algorithm.

This approach suffers from several disadvantages. It doesn't predict the value of power remaining in the battery of sensor. Also, the algorithms used are very expensive and complex [5]. Additionally, it doesn't focus on optimizing the process of the QoS values transmission.

B. The Hidden Markov chain

The hidden Markov chains [6] are designed to model a system with two stochastic processes. The first represents the evolution of observable information while the second is characterized by a not visible states and transition probabilities. The hidden Markov chain can represent the transition from a hidden state to another. This feature allows it to predict the behavior of an object or a sequence.

A hidden Markov chain is characterized by the following elements:

- N is the number of hidden state in the model. We denote the different accessible state as: $S = \{S1, S2, \dots, Sn\}$ and qt the state at t instant.
- M is the number of distinct observations per state. Observation symbols correspond to the physical output of the modeled system. We denote the different symbols as: $V = \{V1, V2, \dots, Vm\}$.
- A is the state transition probability distribution $A = [a_{ij}]$, where

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i] \quad 1 \leq i, j \leq N \quad (1)$$

- B is the observation symbol probability distribution in state j, where

$$b_j(k) = P[V(t) = V_k | q_t = S_j] \quad 1 \leq j \leq N, 1 \leq k \leq M \quad (2)$$

- Π is the initial state distribution, $\Pi = [\pi_i]$, where

$$\pi_i = P[q_1 = S_i] \quad (3)$$

In our approach, we used the fuzzy logic to calculate the observation symbol probability distribution B. It consists in the membership probability of the remaining battery power of sensor to different states namely Fully Charged (FC), Charged (C), Partially Loaded (PL), Low (L) and Very Low (VL). More details will be given in Section 3.

C. The fuzzy logic

Fuzzy logic [7] introduced by Zadeh in 1965, is an extension of the theory of binary sets. It allows the representation and processing of imprecise or uncertain data.

Fuzzy logic is based on degrees of truth rather than the usual true or false binary logic. So, it proposes a gradual transition of an object between two states whereas classical set theory allows only full transition. Fuzzy logic system is based on three operations: first, the fuzzification which transforms a crisp value to fuzzy. Second, the inference which makes rules to deduce a fuzzy output set from linguistic input memberships. Third, the defuzzification which transforms a fuzzy set to a crisp value.

III. PROPOSED APPROACH

As shown in Fig. 1, our approach is based on two steps: learning and predicting and aims to estimate a part of monitoring values to save sensing and transmission energy. Therefore, it increases the lifetime duration of wireless sensor network.

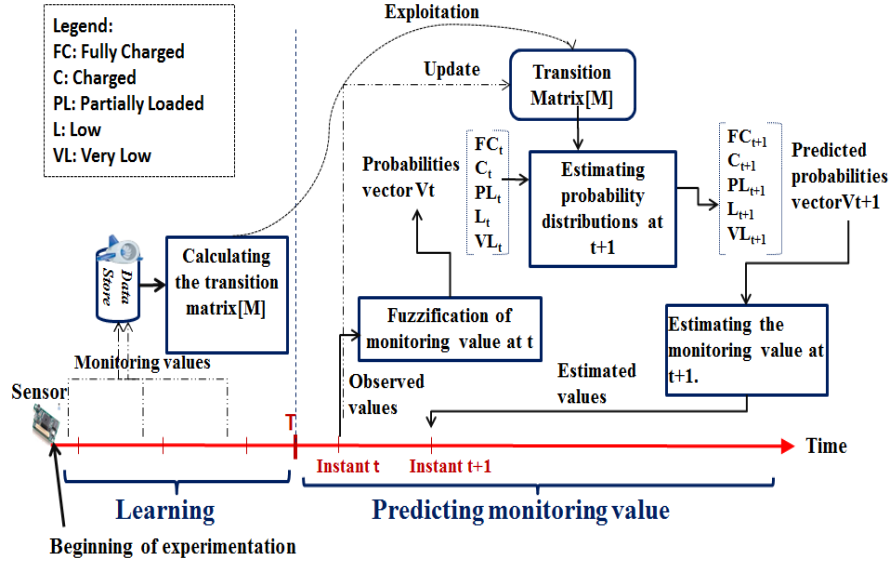


Fig. 1. Proposed approach

A. Learning step

The first step in our approach starts by storing collected values from embedded monitors during the monitoring phase in a database. Then, we calculate the transition matrix that we use later in the second step. Collecting sufficient numbers of monitoring values is necessary to ensure the stabilization of transition matrix.

We denote by S the set of states of the battery: $S = \{FC, C, PL, L, VL\}$, V is the observed variable: $V = \{V_{Energy}: \text{value of power remaining in the battery.}\}$, B is the current probability distributions of observing V_{Energy} at different states at t instant: $B_t = \{P(FC), P(C), P(PL), P(L), P(VL)\}$. The set S is composed of five states which are close to the reality and fit with our prediction model.

- FC: The fully charged state means that the battery is totally charged enabling a high performance functioning.
- C: The charged state denotes that the battery is charged enabling a normal functioning.
- PL: The partially load state denotes that the battery is partially charged but it shows a degradation of the energy level, so it is a precaution state.
- L: The low state denotes that energy level in the battery is low.
- VL: The very low state denotes a critical state of the battery energy level.

To model the transition between these states, we present the transition matrix A associated to the hidden Markov chain as follow:

$$A = \begin{bmatrix} P_{FCFC} & P_{FCC} & P_{FCPL} & P_{FCL} & P_{FCVL} \\ P_{CFC} & P_{CC} & P_{CPL} & P_{CL} & P_{CVL} \\ P_{PLFC} & P_{PLC} & P_{PLPL} & P_{PLL} & P_{PLVL} \\ P_{LFC} & P_{LC} & P_{LPL} & P_{LL} & P_{LVL} \\ P_{VLFC} & P_{VLC} & P_{VLPL} & P_{VLL} & P_{VLVL} \end{bmatrix}$$

P_{ij} is the probability of transition from state i at t instant to the state j at $t+1$ instant, and is expressed as [8]:

$$P_{ij} = \frac{n_{ij}}{\sum_j n_{ij}} \quad (4)$$

Where n_{ij} is the number of sensors which batteries switch from state i at t instant to state j at $t+1$ instant and $\sum_j n_{ij}$ is the total number of sensors with battery level in state i .

B. Predicting step

Now suppose that we are at t instance and we received monitoring value from the monitors deployed on sensor. Instead of asking the monitor for sensing and transmitting the value at $t+1$ instance, we estimate it.

The predicting step is composed of three phases, namely: fuzzifying the monitored values at t instance, estimating the probability distributions at $t+1$ instant and estimating the monitoring value at $t+1$ instant.

1) *Fuzzification of monitoring value at t instance*

Initially, we compute the current probability distributions of observing V_{Energy} at different states via fuzzification process [9].

To describe the variable domain of V_{Energy} , we have used five triangular membership functions defined through five items (threshold1, threshold2, threshold3, threshold4, threshold5), as shown in Fig. 2. The variable domain is divided equally to five sub-domains following the standard designation in fuzzy logic of five items triangular membership. These functions are associated with battery states: (FC - C - PC - L - VL).

The mathematical definitions [10] of these five functions are:

$$\mu_{xFullyCharged}(i) = \begin{cases} 0 & \text{if } x_i > \text{threshold5} \\ \frac{x_i - \text{threshold4}}{\text{threshold5} - \text{threshold4}} & \text{if } \text{threshold4} \leq x_i \leq \text{threshold5} \\ 0 & \text{if } x_i < \text{threshold4} \end{cases}$$

$$\mu_{xCharged}(i) = \begin{cases} 0 & \text{if } x_i < \text{threshold3} \\ \frac{\text{threshold5} - x_i}{\text{threshold5} - \text{threshold4}} & \text{if } \text{threshold4} \leq x_i \leq \text{threshold5} \\ \frac{x_i - \text{threshold3}}{\text{threshold4} - \text{threshold3}} & \text{if } \text{threshold3} \leq x_i \leq \text{threshold4} \\ 0 & \text{if } x_i > \text{threshold5} \end{cases}$$

$$\mu_{xPartiallyCharged}(i) = \begin{cases} 0 & \text{if } x_i < \text{threshold2} \\ \frac{\text{threshold4} - x_i}{\text{threshold4} - \text{threshold3}} & \text{if } \text{threshold3} \leq x_i \leq \text{threshold4} \\ \frac{x_i - \text{threshold2}}{\text{threshold3} - \text{threshold2}} & \text{if } \text{threshold2} \leq x_i \leq \text{threshold3} \\ 0 & \text{if } x_i > \text{threshold4} \end{cases}$$

$$\mu_{xLow}(i) = \begin{cases} 0 & \text{if } x_i < \text{threshold1} \\ \frac{\text{threshold3} - x_i}{\text{threshold3} - \text{threshold2}} & \text{if } \text{threshold2} \leq x_i \leq \text{threshold3} \\ \frac{x_i - \text{threshold1}}{\text{threshold2} - \text{threshold1}} & \text{if } \text{threshold1} \leq x_i \leq \text{threshold2} \\ 0 & \text{if } x_i > \text{threshold3} \end{cases}$$

$$\mu_{xVeryLow}(i) = \begin{cases} 0 & \text{if } x_i < \text{threshold1} \\ \frac{\text{threshold2} - x_i}{\text{threshold2} - \text{threshold1}} & \text{if } \text{threshold1} \leq x_i \leq \text{threshold2} \\ 0 & \text{if } x_i > \text{threshold2} \end{cases}$$

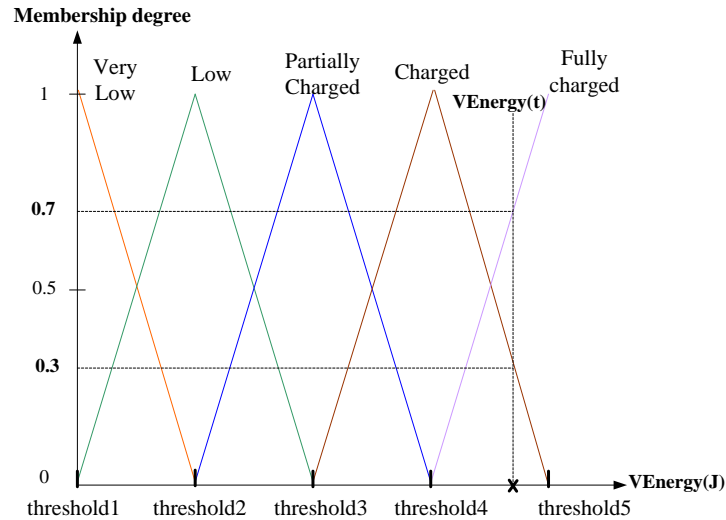


Fig. 2. General form of our discourse universe

We note that each received monitoring value is stored in the database which allows us to continuously update the transition matrix.

2) Estimating probability distributions at t+1 instance

The estimating probability distribution μ_{m+1} at t+1 instance is equal to the multiplication of the probability distribution obtained in the previous step μ_m with the transition matrix A. This equation is adopted from the work in [11]:

$$\mu_{m+1} = \mu_m \cdot A \quad (5)$$

3) Estimating the monitoring value at t+1 instance

The last phase of the second step is the estimation of the monitoring value at t+1 instance which is calculated based on this equation [12]:

$$E_{t+1} = \sum_{i=1}^M (P_i) \cdot E_i \quad (6)$$

Where M is the number of states (five states), P_i is the probability that a sensor will enter in state i at t+1 instance and E_i denotes the energy remaining in the battery of sensor when it is 100% in state i (threshold1, threshold2, threshold3, threshold4, threshold5)

IV. ILLUSTRATION

In follows, we describe the implementation details of our approach. Then, we detail the case study based on the deployment of a WSN in a datacenter and we present experimental results.

A. Architecture of our implementation

To illustrate our approach, we developed a service oriented web application using Google App Engine platform. The deployment architecture of our implementation is illustrated in Fig. 3.

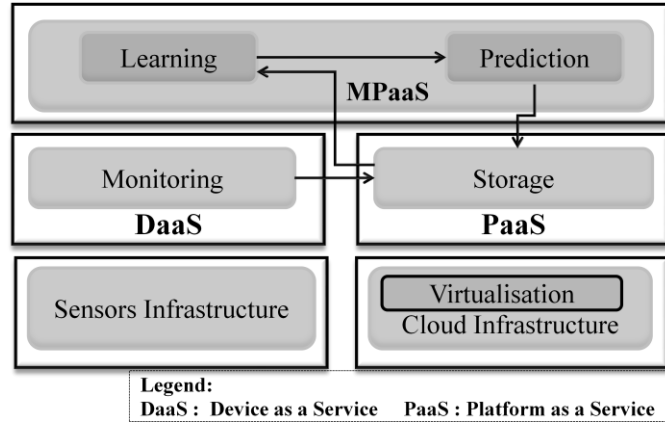


Fig. 3. The deployment architecture

Our implementation relies on two infrastructures namely: embedded sensors infrastructure and cloud infrastructure.

We need a sensors infrastructure because our approach is defined in the context of self-adaptation strategy in WSN applications. This infrastructure enables us to collect monitoring values arising from monitors deployed on sensors. These monitors consist of a service oriented device level that we call Device as a Service (DaaS). The collected values are stored in the Google Cloud database (called *DataStore*). These data enable computing transition matrix that is used in the computing of estimated monitoring values. To perform our experimentation, we used AZEM WSNs simulator to emulate sensor infrastructure.

The storage of monitoring values and computing estimated values operate in cloud infrastructure and was implemented at PaaS level.

We have implemented the following PaaS services in our application: storage of monitoring values, getting monitoring values in defined time interval, computing the transition matrix, fuzzification of monitoring value at instant t, setting monitoring frequency, getting monitoring frequency and computing estimated monitoring values.

B. Case study

To illustrate our approach, we used a case study based on the deployment of a WSN in a datacenter [13]. The datacenter includes many machines and servers for the storage and the processing of data. Administrators must ensure the continuity, availability of these servers and security of data and of the datacenter physical environment.

In addition, datacenter consume between 1.1% and 1.5% of worldwide electricity [14], therefore many solutions are implemented to reduce this increased consumption and monitor the environment such as the deployment of a WSN in a datacenter in order to save energy. Many kinds of sensors can be deployed in the datacenter; we mention sensors for measuring humidity, temperature, state of charge of uninterruptible power supply batteries and movement detectors. These sensors communicate collected data to the gateway that in turn sends to a base station via Internet to enable the control and the management of such environment.

For instance, a temperature and humidity sensors monitor servers and send alarm upon unusually temperature rise, smoke and fire detection. They can also stop the redundant cooling unit in order to save energy and in case of exceeding of the temperature threshold a ventilator or an air conditioner is triggered automatically. A movement detectors sensor sends alarm upon an unauthorized access to the data center or inappropriate activities of personnel.

In case of a power interruption, the servers must continue their work through uninterruptible power supply batteries that are able to bridge power failures and supply equipment in the data center to ensure their availability. Ensuring the availability and effectiveness of these batteries, sensors are used to monitor and measure their state of charge.

The sensors deployed in the datacenter are battery powered to get an optimal availability in case of a power interruption (sensors, fire and smoke detectors, intrusion sensor must work to monitor the environment), to reduce energy consumption and to provide a greater flexibility and operational speed by deploying these sensors in different and difficult accessibility place without being worried of wiring constraint. Thus, the life duration of a sensor is highly dependent on the life of its battery, so monitoring the state of charge of the battery of sensors, optimizing energy consumption and taking preventive measures and decisions are very important. Our approach aims to achieve these goals by predicting the monitoring values and subtract the cost of sending these values. To test our approach, we carry out experiments using our WSNs simulator AZEM on sensors deployed in a data center that incorporates the following types of WSN applications [15]:

- Periodic applications: Sensors send monitoring data periodically such as sending measures of state of charge of UPS batteries.
- Event-based applications: sensors send monitoring data when an event occurs such as sending alarm in case of unauthorized access in datacenter.
- Hybrid application: Some applications need a mixture of both periodic and event-based data monitoring such as a sensor that monitors the temperature of a server in the datacenter periodically and besides it sends alarms when a specific event occurs.

C. Experimental results

Our experiments are ensured using those properties: the sent message size is equal to 10 000 bit, the initial energy value for each sensor is equal to 3 joules. We use three energy models, namely μ AMPS Specific Model [16], Mica2 Specific Model [17] and Mica2 Specific Model with actual measurement [18]. We carried out experiments for periodic (P), event-based (Evt) and hybrid (H) applications.

First, we perform these experiments with an estimation frequency equal to the half of monitoring values, this means that sensor sends the monitoring value at t instant, and we estimate the monitoring value at $t + 1$ instant. Realizing these experiments on three energy models with maintaining the same application type, we obtain graphics that contain three curves as shown in Fig. 4 dealing with periodic application.

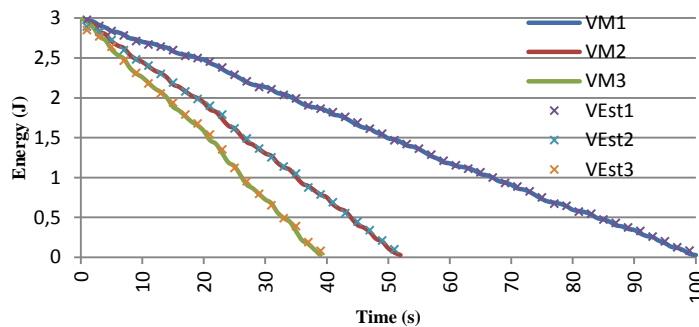


Fig. 4. Estimation result of half of values in P application

We notice that estimated values are around the curve of the received monitoring values. That's prove the efficiency of these estimated values

In second case we carried out these experiments with an estimation frequency equal to two-thirds of monitoring values as shown in Fig. 5 that deals with Event-based applications. We note that the estimated values still surrounds the received monitoring values curve which allows us to deduce that our approach tolerates the two-thirds estimation frequency.

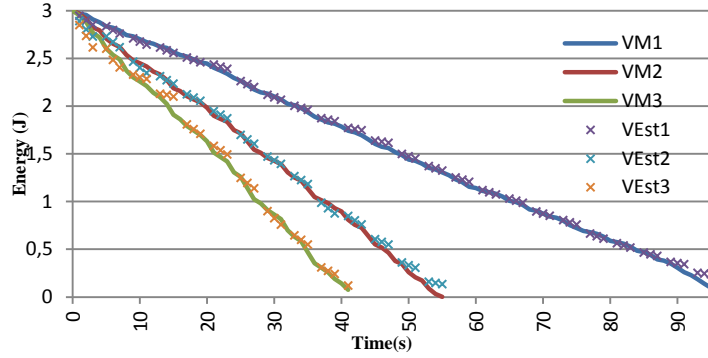


Fig. 5. Estimation result of two-third of values in Evt application

Finally, we perform experiments with an estimation frequency equal to three-quarters of monitoring values. Results are shown in Fig. 6 regarding Hybrid application.

With three-quarters frequency estimation, we notice that sometimes, estimated values move away from the received monitoring values curve. This is caused by the error propagation due to the integration of estimated values that may contain estimation error in the database used to generate the transition matrix influencing thereby the second and the third generated values.

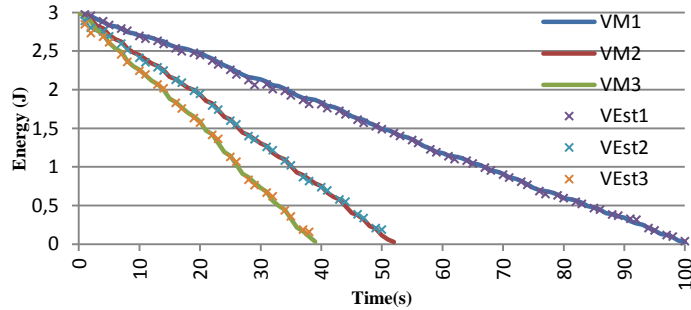


Fig. 6. Estimation result of three-quarters of values in H application

Legend	
VM1:	Monitoring values following μ AMPS Specific Model
VM2:	Monitoring values following Mica2 Specific Model
VM3:	Monitoring values following Mica2 Specific Model with actual measurements
VEst1:	Estimated values following μ AMPS Specific Model
VEst2:	Estimated values following Mica2 Specific Model
Vest3:	Estimated values following Mica2 Specific Model with actual measurements

V. EVALUATION

In order to determine the accuracy of our approach, we calculate the estimation error. This error is defined as the difference between real monitoring value and estimated value in absolute value. We also compute the average error for each application and for each energy model.

Table I, Table II and Table III summarize the average error of monitoring values estimation for each type of application. For example following μ AMPS Specific Model, for periodic application, the average error is equal to 0.0135, for event-based application, the average error is equal to 0.0246, while for hybrid application, the average error is equal to 0.0171.

In conclusion, we note that the average error in the periodic applications types is less than the average error in the other two ones, since energy consumption is almost stable over time due to no random events or events that affect the behavior of the battery and therefore the transition matrix is more stable and the estimation is better.

We remark that as the number of estimated values increases, the average error increases. We also note that when there is a sudden change in the behavior of the battery, the error of estimation increases at this instant. However, through continuous learning, the transition matrix is updated each instant and our prediction approach adjusts itself and therefore the error rate decreases.

Our approach increases the life duration of sensors battery by reducing the cost of transmitting monitoring values. As shown in Table IV, we save 50% of the cost of the transmission process with the prediction of the half of the monitoring values. For instance, according to μ AMPS Specific Model, the required energy to send a message is equal to 1.04 μ J. To send 100 messages of one bit,

the sensor consumes 104 μJ ($1.04 \mu\text{J} * 100$, see Table IV). With our approach, when we estimate the half of the sensed values, the sensor consumes only 52 μJ ($1.04 \mu\text{J} * 50$, see Table I). When we estimate the two-thirds of the sensed values, the sensor consumes 34.32 μJ ($1.04 \mu\text{J} * 33$, see Table II). In this case we save 66% of the cost of the transmission process. When we estimate three-quarters of the monitored values, we can save 75% of the transmission cost (see Table III).

TABLE I. THE ESTIMATION ERRORS AVERAGE OF HALF OF THE SENSED VALUES

Application type Model	Periodic	Event-Based	Hybrid
μAMPS Specific Model	0.0135 J	0.0246 J	0.0171 J
Mica2 Specific Model	0.021 J	0.025 J	0.022 J
Mica2 Specific Model with actual measurement.	0.032 J	0.038 J	0.034 J

TABLE II. THE ESTIMATION ERRORS AVERAGE OF TWO-THIRDS OF THE SENSED VALUES

Application type Model	Periodic	Event-Based	Hybrid
μAMPS Specific Model	0.0169 J	0.027 J	0.019 J
Mica2 Specific Model	0.024 J	0.026 J	0.025 J
Mica2 Specific Model with actual measurement.	0.03 J	0.046 J	0.032 J

TABLE III. THE ESTIMATION ERRORS AVERAGE OF THREE-QUARTERS OF THE SENSED VALUES

Application type Model	Periodic	Event-Based	Hybrid
μAMPS Specific Model	0.025 J	0.0311 J	0.026 J
Mica2 Specific Model	0.040 J	0.046 J	0.044 J
Mica2 Specific Model with actual measurement.	0.043 J	0.058 J	0.056 J

TABLE IV. THE ENERGY CONSUMPTION OF A SENSOR WITH AND WITHOUT ESTIMATION

Frequency Model	Without estimation	Half of values	Two-third of values	Three-quarter of values
μAMPS Specific Model	104 μJ	52 μJ	34.32 μJ	26 μJ
Mica2 Specific Model	27 μJ	13.5 μJ	8.91 μJ	6.75 μJ
Mica2 Specific Model with actual measurement.	460.2 μJ	230.1 μJ	151.68 μJ	115.05 μJ

VI. CONCLUSION

In this paper, we presented the MPaaS approach, which allows optimizing energy consumption of WSNs. The optimization process is based on the prediction of monitored values, that enables to economize their sensing and the transmission energy cost. Our approach relies on rigorous reasoning over the hidden Markov chain and the fuzzy logic. It is composed of two steps. The first allows learning the behavior of the WSNs application energy consumption model. The second uses the hidden Markov chain to predict a part of the sensed values. The learning process is continuously updated in order to get better prediction result. The use case of WSN deployed in a datacenter is used to illustrate the feasibility of our approach. The carried out experiment shows that we can increase battery lifetime and the negligible computed estimation error prove the correctness of the propose approach. Our future work will focus on deploying and assessing our approach in a real and a large scale environment.

REFERENCES

- [1] A. Jemal and R. Ben Halima, "A QoS-Driven Self-Adaptive Architecture For Wireless Sensor", Networks, IEEE International Conference on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), Hammamet, Tunisie, pp. 125-130, 2013.
- [2] Akyildiz, I. F.; Kasimoglu, I. H "Wireless sensor and actor networks: research challenges." Ad Hoc Networks, vol. 2, no. 4, pp. 351- 367,2004.
- [3] Jeffery,O.K, Chess, D. M., "The vision of autonomic computing."Computer, vol. 36, no. 1, pp. 41- 50, 2003.
- [4] P. Hu, Z. Zhou, Q. Liu, F. Li, "The HMM-based Modeling for the Energy Level Prediction in Wireless Sensor Networks", In Proceedings of the 2007 2nd IEEE Conference on Industrial Electronics and Applications (ICIEA '07), pp. 2253–2258, May 2007.
- [5] Santosh S. Vempala and Georgia Tech, "A Random Sampling based Algorithm for Learning Intersections of Halfspaces", Journal of the ACM, vol. 57, no. , 6, pp. 111–124, 2010.
- [6] Lawrence R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition", Readings in speech recognition, pp. 267–296, 1990.
- [7] Mamdani, E.H., Assilian, S, "An experiment in linguistic synthesis with a fuzzy logic controller". Int. J. Man Mach. Studies, vol. 7, no. 1, pp. 1–13, 1975.
- [8] G. Timofeeva, N. Timofeev, "State estimation problem for markov chain model", In proceeding of the 5th International Conference on Physics and Control (PhysCon 2011) , Leon, pp 1-4, 2011.
- [9] R. Ben Halima, " Design, implementation and testing of a bus architecture for self-healing distributed applications based on web services ", Ph.D. thesis, universit  de Toulouse et de l'universit  de Sfax, Mai 2009.
- [10] Nasser, H., "Fuzzy numbers: Positive and nonnegative", International Mathematical forum", vol. 3, no. 36, pp. 1777-1780, 2008.
- [11] P. Br maud, "Markov chains. Gibbs fields, "Monte Carlo simulation and queues", Springer, ISBN 0-387-98509-3, 1998.
- [12] F.M. Dekking C. Kraaikamp, H.P. Lopuhaa, L.E. Meester.A., "Modern Introduction to Probability and Statistics", Springer, ISBN 1-85233-896-2, 2005.
- [13] "Wireless Sensors Improve Data Center Energy Efficiency: Technology Case Study Bulletin", Technology Case Study Bulletin, Federal Energy Management Program, 2010.
- [14] <http://www.greenit.fr/>
- [15] M. Fonoage, M. Cardei, and A. Ambrose, "A QoS Based Routing Protocol for Wireless Sensor Networks", IEEE International Performance Computing and Communications Conference, pp. 122-129, 2010.

- [16] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, A. Chandrakasan, "Physical Layer Driven Protocol and Algorithm Design for Energy-Efficient Wireless Sensor Networks", Seventh Annual ACM SIGMOBILE Conference on Mobile Computing and Networking, Rome, Italy, pp 272-286, 2001.
- [17] J. Polastre, J. Hill, D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks", In Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (SenSys), Baltimore, Maryland, USA , pp 95-107, 2004.
- [18] V. Shnayder, M. Hempstead, B. Chen, G. Werner Allen, and M. Welsh, "Simulating the Power Consumption of LargeScale Sensor Network Applications", In Proceedings of the Second ACM Conference on Embedded Networked Systems, Baltimore, Maryland, pp. 188-200, 2004