Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 32 (2014) 1154 – 1159

Procedia
Computer Science

International Workshop on Wireless Networks and Energy Saving Techniques (WNTEST-2014)

Energy saving in WSN using monitoring values prediction

Ahmed JEMAL[a,b,c], Marwa HACHICHA[a], Riadh BEN HALIMA[a,b,c,] *, Ahmed HADJ KACEM[a], Khalil DRIRA[b,c], Mohamed JMAIEL[a]

[a]*University of Sfax, ReDCAD Laboratory, B.P. 1173, Sfax-Tunisia*
[b]*CNRS; LAAS; 7 av. du Colonel Roche, F-31400 Toulouse, FRANCE*
[c]*Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; F-31400 Toulouse, France*

Abstract

The Wireless Sensor Networks (WSNs) deployment introduces many issues and challenges mainly in terms of energy independence. In this context, we adopted the IBM control loop which is 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 sending periodically QoS values such as the value of power remaining in the battery of each sensor. We notice that the transmission process is very costly 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. Our approach is based on the hidden Markov chain and the fuzzy logic. It is composed of two steps: (i) learning which allows apprehending the WSNs behavior and (ii) prediction which estimates QoS monitoring values. A WSN application deployed in a datacenter is studied as an illustration. The carried out experiments over AZEM[1] WSN simulator show that the gain varies from 25% to 75% of the battery energy.

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

1. Introduction

WSNs are composed of nodes that monitor and control the environment. The collected information is transmitted to the base station to process their analysis and exploitation[2]. However, the architecture of WSN is influenced by

* Corresponding author. Tel.: +216-74-275-088; fax: +216-74-666-578.
E-mail address: riadh.benhalima@enis.rnu.tn

[1] An extended version of AvroraZ simulator: http://www.redcad.org/members/benhalima/azem/.

1877-0509
doi:10.1016/j.procs.2014.05.547

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.

An autonomic enabled-architecture according to the IBM control loop[3] should be adopted in order to adapt WSNs applications and increase the WSNs lifetime. The autonomic loop is composed of four steps, namely: Monitoring, Analysis, Planning and Execution. 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 allows reducing the energy consumption of sensors and maximizing their lifetime and therefore the life of WSNs applications.

In this paper, we focus on predicting a part of these values. We propose MPaaS: Monitoring values Prediction as a Service which uses the hidden Markov chains (HMC) and the fuzzy logic to optimize the energy consumption without too much computational overhead. 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 this paper is organized as following. In section 2, we introduce the related work. 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 rate error. The last section concludes the paper.

2. Related work

Several researches in WSN have looked at various ways of saving energy. In particular, S. Goel et al.[4] proposed a mechanism called Prediction-based monitoring for energy efficient monitoring. This approach focuses only on identifying correlation in monitoring data, eliminating their transmission and predicting them at the monitored node.

The work proposed by P.Hu et al.[5] is an estimation model based on the HMC to predict the energy level of a sensor node. The proposed 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 limits. It does not predict the value of power remaining in the battery of sensor. Also, the algorithms used are very expensive and complex[6]. Additionally, it does not focus on optimizing the process of the QoS values transmission.

3. Proposed approach

As shown in Fig. 1, our approach is based on two main steps: learning and prediction and aims at estimating a part of monitoring values to save sensing and transmission energy.

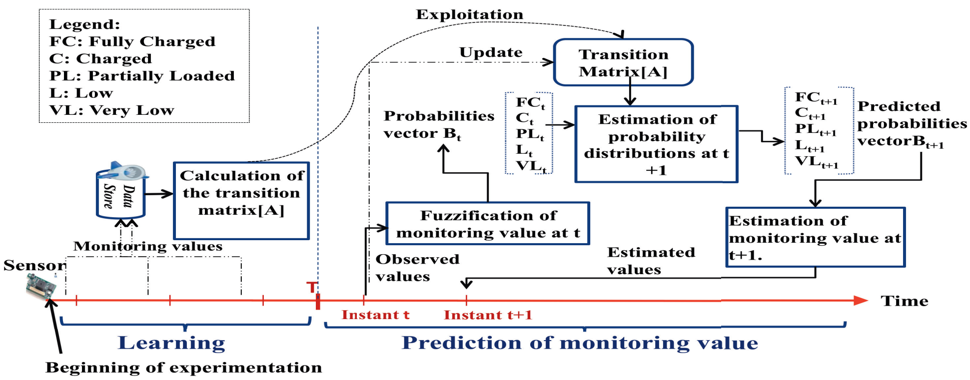


Fig. 1.Proposed approach

3.1. Learning step

The first step in our approach starts by storing collected values from embedded monitors in a Google cloud datastore. 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** = {*VEnergy*: value of power remaining in the battery.}, **B** is the current probability distributions of observing *VEnergy* at different states at t instant: $\mathbf{B_t}$ = {P (FC), P (C), P (PL), P (L), P (VL)}.

To model the transition between these states, we present the transition matrix **A** associated to the HMC as follow:

$$\left\| \begin{matrix} P_{FCFC} & P_{FCC} & P_{FCPL} & P_{FCL} & P_{FCVL} \\ P_{CFC} & P_{CC} & P_{CPL} & P_{CL} & P_{CVL} \\ P_{PLFc} & P_{PLC} & P_{PLPL} & P_{PLL} & P_{PLVL} \\ P_{LFC} & P_{LC} & P_{LPL} & P_{LL} & P_{LVL} \\ P_{VLFC} & P_{VLC} & P_{VLPL} & P_{VLL} & P_{VLVL} \end{matrix} \right\|$$

$P_{ij=}\dfrac{n_{ij}}{\sum_j n_{ij}}$ is the probability of transition from state i at t instant to the state j at t +1 instant, where n_{ij} is the number of sensors with battery level in state i. For instance, PFCPL means the probability of transition from fully charged state at t instant to partially loaded state at t+1 instant.

3.2. Prediction step

Now suppose that we are at t instant 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 instant, we estimate it.

3.2.1. Fuzzification of monitoring value at t instant

Initially, we compute the current probability distributions of observing *VEnergy* at different states via fuzzification process. To describe the variable domain of *VEnergy*, we have used five triangular membership functions defined through five items (threshold1, threshold2, threshold3, threshold4, threshold5), as shown in Fig. 2 (a). The variable domain is divided equally to five sub-domains following the standard designation in fuzzy logic of five items triangular membership[7]. These functions are associated with battery states: (FC - C - PC - L – VL). In fact, simple functions are used to build membership function because using complex function causes a high computational cost and they do not add more precision.

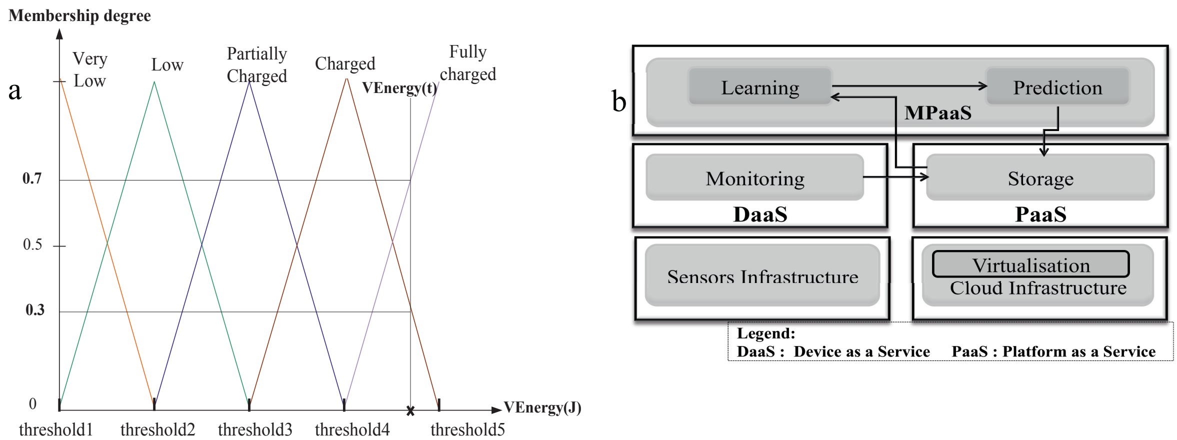


Fig. 2. (a) General form of our discourse universe; (b) The deployment architecture

3.2.2. Estimation of probability distributions at t+1 instant

The estimating probability distribution $\mathbf{B_{t+1}}$ at t+1 instant is equal to the multiplication of the probability distribution obtained in the previous step $\mathbf{B_t}$ with the transition matrix **A**.

$$B_{t+1} = B_t.A \tag{1}$$

3.2.3. Estimation of the monitoring value at t+1 instant

The last phase of the second step is the estimation of the monitoring value at t+1 instant which is calculated based on this equation [8]:

$$E_{t+1} = \sum_{i=1}^{M} (Pi * Ei) \tag{2}$$

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

4. Illustrations

In what 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.

4.1. 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. 2 (b). Our implementation relies on two infrastructures namely: embedded sensors infrastructure and cloud infrastructure. Sensors infrastructure enables us to collect monitoring values arising from monitors deployed on sensor. 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.

4.2. Case study

To illustrate our approach, we used a case study based on the deployment of a WSN in a datacenter. The sensors deployed in the datacenter are battery powered to get an optimal availability in case of a power interruption 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 subtracting 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 periodic, event-based and hybrid applications.

4.3. 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 and AZEM is the used WSN simulator. We use three energy models, namely µAMPS Specific Model [9], Mica2 Specific Model [10] and Mica2 Specific Model with actual measurement [11].

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 dealing with periodic application, this means that sensor sends the monitoring value at t instant, and we estimate the monitoring value at t+1 instant as shown in Fig. 3 (a). In the second case we carried out these experiments with an estimation frequency equal to two-thirds of monitoring values as shown in Fig. 3 (b). Finally, we perform experiments with an estimation frequency equal to three-quarters of monitoring values dealing with Hybrid application (see Fig. 4). Realizing these experiments on three energy models with maintaining the same application type, we notice that estimated values are around the curve of the received monitoring values. That's proving the efficiency of these estimated values.

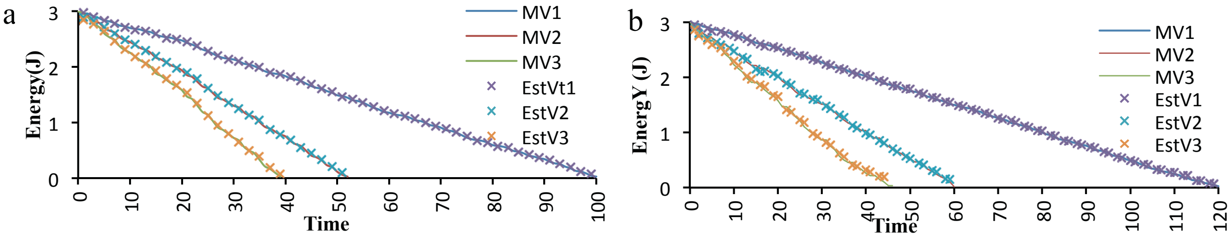
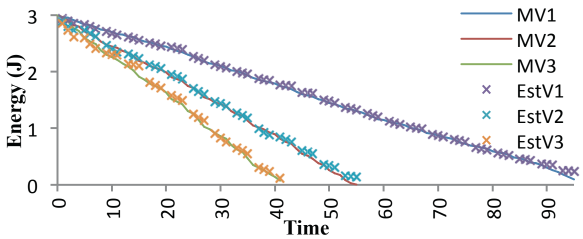


Fig. 3. (a) Estimation result of half of values in P application; (b) Estimation result of two-third of values in Evt application



Legend
MV1: Monitoring values following µAMPS Specific Model
MV2: Monitoring values following Mica2 Specific Model
MV3: Monitoring values following Mica2 Specific Model with actual measurements
EstV1: Estimated values following µAMPS Specific Model
EstV2: Estimated values following Mica2 Specific Model
EstV3: Estimated values following Mica2 Specific Model with actual measurements

Fig. 4. Estimation result of three-quarters of values in H application

5. Evaluation

In order to determine the accuracy of our approach, we calculate the estimation error defined as:

$$Error = | MV - EstV | \tag{3}$$

Where *MV* is the real monitoring value and *EstV* is the estimated value. We also compute the average error for each application and for each energy model. For example following µAMPS 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 periodic applications types is less than the average error in the other two ones, since energy consumption is almost stable over time due to the absence of random 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 1, we save 50% of the cost of the transmission process with the prediction of the half of monitoring values. For instance, according to µAMPS Specific Model, the required energy to send a message is equal to 1.04 µJ. To send 100 messages of one bit, the sensor consumes 104 µJ (1.04 µJ * 100, see Table 4). With our approach, when we estimate the half of the sensed values, the sensor consumes only 52 µJ (1.04 µJ * 50). When we estimate the two-thirds of the sensed values, the sensor consumes 34.32 µJ (1.04 µJ * 33). In this case we save 66% of the cost of the transmission process. When we estimate three-quarters of the monitoring values, we can save 75% of the transmission cost. In addition, there is no additional power cost for the sensor because we separate the application logic of WSN from the prediction logic ensuring that the prediction layer is not bound to the application layer. In this way the implementation of our method does not effect on the sensors energy.

Table 1. The energy consumption of a sensor with and without estimation.

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

6. 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 monitoring 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 predicts a part of the sensed values. The learning process is continuously updated in order to get better prediction results. 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 proves the correctness of the proposed 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. Samir Goel, Tomasz Imielinski, “Prediction-based Monitoring in Sensor Networks: Taking Lessons from MPEG”, ACM computer Review, vol. 31, no. 5, 2001.
5. 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.
6. 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.
7. Nasseri, H., “Fuzzy numbers: Positive and nonnegative”, International Mathematical forum”, vol. 3, no. 36, pp. 1777-1780, 2008.
8. P. Brémaud, “Markov chains. Gibbs fields, “Monte Carlo simulation and queues”, Springer, ISBN 0-387-98509-3, 1998.
9. 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.
10. 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.
11. 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.