Sequential Hypothesis Testing based Polling Interval Adaptation in Wireless Sensor Networks for IoT Applications

Sungryoul Lee
The Attached Institute of Electronics and Telecommunications Research Institute (ETRI),
Yuseong P.O.Box1, Daejeon, Korea
[e-mail: srlee0525@nsr.re.kr]
*Corresponding author: Sungryoul Lee

Received August 2, 2016; revised November 28, 2016; accepted December 27, 2016;
published March 31, 2017

Abstract

It is well known that duty-cycling control by dynamically adjusting the polling interval according to the traffic loads can effectively achieve power saving in wireless sensor networks. Thus, there has been a significant research effort in developing polling interval adaptation schemes. Especially, Dynamic Low Power Listening (DLPL) scheme is one of the most widely adopted open-looping polling interval adaptation techniques in wireless sensor networks. In DLPL scheme, if consecutive idle (busy) samplings reach a given fixed threshold, the polling interval is increased (decreased). However, due to the trial-and-error based approach, it may significantly deteriorate the system performance depending on given threshold parameters. In this paper, we propose a novel DLPL scheme, called SDL (***S**equential hypothesis testing based **D**ynamic **LPL***), which employs sequential hypothesis testing to decide whether to change the polling interval conforming to various traffic conditions. Simulation results show that SDL achieves substantial power saving over state-of-the-art DLPL schemes.

Keywords: Low power listening, sequential hypothesis testing, wireless sensor networks

https://doi.org/10.3837/tiis.2017.03.008 **ISSN : 1976-7277**

1. Introduction

Recent advances in wireless communications and low-cost electronic devices make it possible to materialize Internet of Things (IoT) which enables to support the interaction between things through Internet access anytime and anywhere. A number of things perform environmental monitoring, i.e., body, automotive and consumer electronics, and share sensing data with others. Furthermore, by integrating Web services and Cloud computing, new types of IoT applications involve smart home, smart care, automation and provide entertainment, security, telecommunication [1].

In fact, wireless sensor networks (WSNs) is one of the most important elements for IoT applications. As shown in **Fig. 1**, connecting WSNs to the Internet is possible to connect heterogeneous systems and provide common services such web service. For successful realization of IoT applications, how to save the power of sensor is one of main issues in WSNs. Sensor are generally equipped with small battery and cannot be replaced nor recharged due to environmental constraints. Therefore, since energy of sensor is a limited resource, power saving mechanism at sensor must be considered to prolong network lifetime as much as possible.

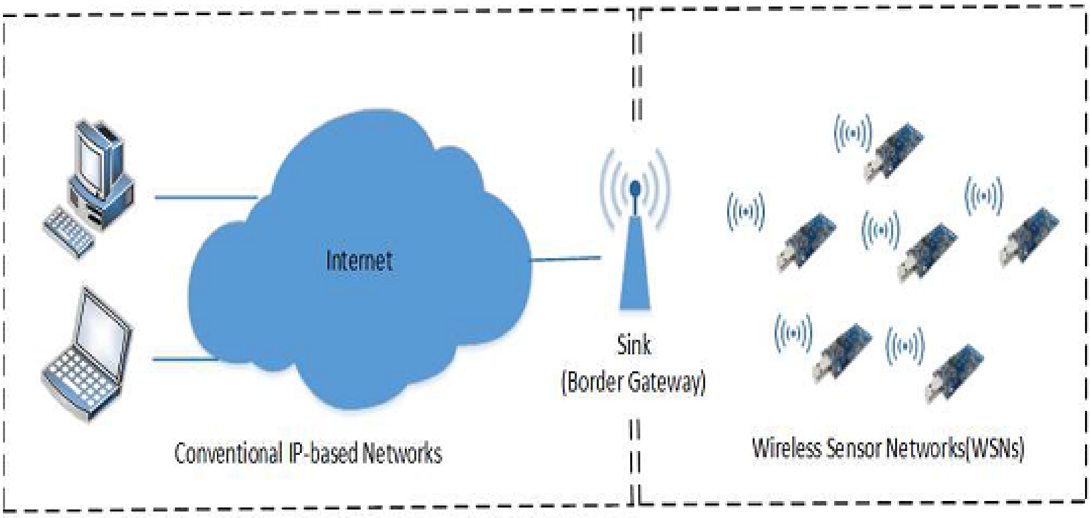


Fig. 1. WSNs-based IoT

In recent few years, duty-cycling has provided a great opportunity to mitigate energy consumption in WSNs; the sensor alternates between sleep mode and wake-up mode periodically to save power. In wake-up mode, the sensor consumes high power as in transmit or receiving mode while being idle. Therefore, the most effective approach is duty-cycling where the sensor is put in sleep mode to mitigate energy consumption of idle listening. Note that idle listening means the time spent in wake-up mode without receiving any radio packets. Duty-cycling can be basically classified into two categories: synchronous and asynchronous. Especially, asynchronous duty-cycling MAC protocol [2, 3] based on *Low Power Listening* (LPL) scheme is very attractive from the implementation point of view due to not requiring any time-synchronization such as S-MAC [5]. In LPL scheme, with data packet to transmit, a sender first transmits a series of short preambles longer than polling interval to guarantee asynchronous rendezvous between the sender and its receiver. Since the receiver samples the channel every polling intervals to check for activity, it can detect a short preamble when being waking-up. If channel is busy, the receiver stays awake for preparing following communication. Otherwise, it goes back to sleep until the next polling interval.

Selecting the optimal polling interval is vital to maximize the performance of LPL scheme. As LPL scheme use the fixed polling interval, several works [4, 6, 7] have introduced Dynamic LPL (DLPL) scheme to effectively response to the varying traffic conditions. The idea behind DLPL scheme is to change the polling interval after consecutive idle (or busy) channel sampling at the current polling interval. If there are U consecutive idle polling, the polling interval is increased. On the other hand, D consecutive busy polling induce a polling interval decrease. Main advantage of this approach is that it can achieve good performance despite its simple design and implementation.

DLPL scheme has some critical issues. Most importantly, an open issue is how to optimally set the parameters U and D of DLPL scheme. Apparently, DLPL scheme has a purely heuristic nature, and hence it cannot work properly in stable traffic conditions since it increases (or decreases) the polling interval upon consecutive idle (busy) sampling. In other words, it may produce the deterioration in performance since it tends to be too aggressive or too conservative in adapting the polling interval. Polling interval adaptation is crucial to duty-cycled WSNs, and therefore many researches [4, 8, 9, 10] have paid to develop novel adaptation mechanisms. However, to the best of our knowledge, there is little work so far that investigates the effectiveness of its mechanism without strobe overhead. This motivates us to propose a new DLPL scheme, called SDL, which relies on a combination of sequential hypothesis testing and energy efficiency to find the best polling interval for the current traffic loads. We conduct extensive simulations and show that SDL achieves substantial power saving over state-of-the-art DLPL schemes.

The contributions of this paper are as follows:

- DLPL scheme searches the best polling interval and it, by its trial-and-error nature, experiences periodical polling interval fluctuations. It should be emphasized that the main goal of this paper is to address this problem and to provide insight to the design of better polling interval adaptation in duty-cycled WSNs.
- In this paper, we introduce the new metric to measure energy efficiency in channel sampling. It enables decision-makers to change the current polling interval. Also, we believe that our main contribution is the proposal of a novel algorithm, SDL, based on sequential hypothesis testing. It only use the previous results of channel sampling to decide whether to change the polling interval, thus is compatible with DLPL Scheme. Simulation results show that our algorithm offers considerable performance improvements over state-of-the-art DLPL schemes.

The remainder of this paper is organized as follows. We review related work in Section 2, and also present an overview of sequential hypothesis testing in Section 3. We propose a novel DLPL scheme, SDL, using sequential hypothesis testing to dynamically adjust the polling interval conforming to traffic conditions in Section 4. Simulation results are illustrated in Section 5, while conclusion is given in Section 6.

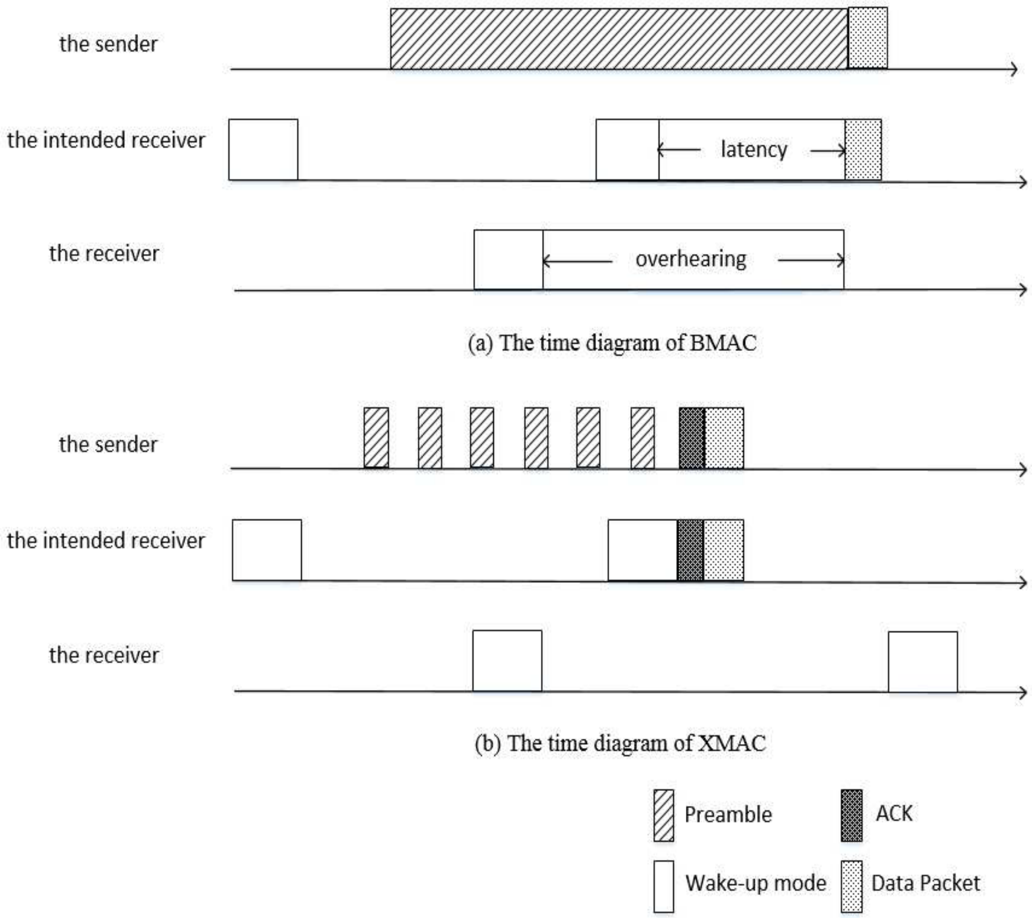


Fig. 2. The time diagram of B-MAC and X-MAC

2. Related Work

There have been remarkable studies on asynchronous MAC protocol in duty-cycled WSNs. Asynchronous MAC protocol uses a preamble to rendezvous between a sender and its receiver, where the receiver transmits ACK indicating that it is ready for data packet reception. Also, to adapt to traffic loads, a sensor dynamically adjusts the polling interval with feedback such as previous sampling results, queue length and number of successfully received data packets. In this section, we go into details these studies as related work.

B-MAC [2], which first introduced the concept of *Low Power Listening* (LPL) scheme, used the long preamble for synchronization between a sender and a receiver. In B-MAC, prior to transmission of data packet, the sender transmits a long preamble to notify the receiver of incoming data packet (see **Fig. 2(a)**). However, basic LPL scheme incurs overhearing problem at non-intended receivers and also introduced excessive latency at each hop. To address these problems, a short preamble was used in [3] to allow the receiver to determine whether or not it should be active. As shown as **Fig. 2(b)**, X-MAC [3] introduced a series of short preamble containing address of the destination to avoid overhearing problem on non-intended receivers. Also, the authors adopted the early ACK to notify the sender that the receiver is ready to receive a data packet. Thus, the burst of short preamble can be stopped by the early ACK. A similar approach such as [12] and [13] was to replace the short preambles with the chucks containing the number of remaining chucks before transmitting the data packet. By counting the remaining chunk, the receiver can known how long it has to keep sleep before receiving the data packet.

Ever since LPL scheme emerged, polling interval adaptation has been a hot research topic in recent years and a number of algorithms [4, 8, 9, 10] have been proposed. Most of these algorithms guess the traffic conditions based on previous results of channel sampling. This approach is referred to as Dynamic LPL (DLPL) scheme. Note that it follows the mechanism identified in Section 1. DLPL scheme is generally compliant with LPL scheme and very simple to implement, and hence, is widely-adopted in duty-cycled WSNs. BoostMAC [4] was one of the most well-known DPL scheme based on AIMD (Addictive Increase Multiplicative Decrease). Specifically, the sensor increases the polling interval in addictive manner if it finds the channel idle. Otherwise, the polling interval is decreased multiplicatively by busy polling. A similar approach is the Maximally Traffic-Adaptive MAC (MAXMAC) [8] which defined the fixed thresholds to decrease the polling interval according to the number of received data packets. Also, in [9, 10], the authors proposed the adapting the MAC schedule to node and network conditions to improve performance under the wide range of traffic conditions and for both unicast and broadcast packets.

Although well-intentioned, there is a fundamental issue when designing DLPL scheme, i.e., how to determine parameters U and D. For example, BoostMAC [4], which employs $U = 1$ and $D = 1$, has very reactive adaptation property. This approach may experience unnecessary polling interval fluctuations and hence it significantly deteriorates the performance. To address this issue, previous works [14, 15] proposed an adaptive polling interval control through the queue management to achieve high performance under various traffic conditions. The queue state of sensor is a good indicator for the network status, and thus the sensor can adapt the polling interval by inferring traffic conditions. In [11], the authors proposed two adaptive solutions to meet the target rate of data packets. In first approach, the sensor increases/decreases its polling interval based on the number of successfully received data packets. The second one uses control theory to adjust the polling interval to accommodate the number of data packets to be sent or to save the power. These mechanisms are conservative in polling interval changes compared to DLPL scheme. However, they cannot timely adapt to traffic loads and select the polling interval over-conservatively.

Sequential hypothesis testing [16] is a suitable technique for making a decision in real-time. Therefore, it has been used to tackle some decision problems in different networks. Two recent studies [17, 18] focused on to develop new sequential schemes for detecting portscan [17] and replica node [18]. The idea in [19] was similar to our work in terms of developing a control mechanism based on sequential hypothesis testing. In [19], the authors employed sequential hypothesis testing to improve the rate adaptation mechanism of IEEE 802.11 WLANs.

3. Preliminary

Now, we briefly introduce some definitions underlying Sequential Hypothesis Testing (SHT), a well-known statistical inference, used in our proposed algorithm. SHT is one of the widely used techniques for operations research and industrial engineering. SHT rests on two hypotheses H_0 and H_1, which means that a statistical testing is performed to compare their corresponding probability distributions $P(x_n|H_0)$ and $P(x_n|H_1)$ for sequence of i.i.d.

observations X_n, respectively. To find evidence that H_0 or H_1 is true, we can compute the likelihood ratio as follow.

$$\rho(n) = \rho(n-1)\,\frac{P[x_n|H_1]}{P[x_n|H_0]} \tag{1}$$

with $\rho(1)$ defined as

$$\rho(1) = \frac{P[x_1|H_1]}{P[x_1|H_0]} \tag{2}$$

Intuitively, if the likelihood ratio is large, we can accept H_1. Otherwise, if the likelihood ratio is small, we can accept H_0. However, False Positives (False Negatives) can occur when accepting H_1 (H_0) with probability α (β). To terminate the sequence test with enough confidence, SHT computes two constants A and B after making observation, $x_1, x_2, \cdots, x_n$. The decision is made as follows:

Case 1. $\rho(n) \geq A$, then accept H_1,
Case 2. $\rho(n) \leq B$, then accept H_0,
Case 3. $B < \rho(n) < A$, then take another observation.

By Wald [16], this test can provide the same level of precision with as the test by using the precise value for A and B where $A = \frac{1-\beta}{\alpha}$ and $B = \frac{\beta}{1-\alpha}$ given α and β, respectively.

Table 1. System parameters

Symbol	Meaning	Value
P_{tx}	Power in transmitting	52.2mW
P_{rx}	Power in receiving or listening	56.4mW
P_{sleep}	Power in sleeping	3μW
t_{cls}	Average carrier sense time	2ms
t_B	Time to transmit or receive a byte	32μs
L_{data}	Data packet length	50B
$L_{preamble}$	Preamble packet length	15B
L_{ack}	ACK packet length	11B
S_p	Time to transmit the preamble at the sender	0.48ms
S_a	Time to listen the ACK at the sender	0.352ms
S_d	Time to transmit the data at the sender	1.6ms
R_l	Wake-up time at the receiver	10ms
R_a	Time to transmit the ACK at the receiver	0.352ms
R_d	Time to receive the data at the receiver	1.6ms

4. Sequential hypothesis test based Dynamic LPL scheme

In this section, we present the details of our polling interval adaptation scheme, **SDL** (***S****equential hypothesis testing based* ***D****ynamic* ***LPL***). SDL is to guess incoming traffic loads at the receiver and to select the appropriate polling interval incurred minimal communication overhead. To achieve this goal, SDL adopts a new metric, the expected number of received packets per energy unit, to track energy efficiency at the receiver. Also, to ensure compatibility with DLPL scheme, SDL uses only the previous result of channel sampling to decide the next polling interval.

Algorithm 1 SDL

```
Initialization :
For $1 \le i \le K$, compute $X_i$ using Eqs.(3)–(4)
Set $n = 1$, $\rho = 1$
Do for $n$th observation :
if channel is idle then
    $\rho = \rho\gamma$
    if $\rho \ge A$ then
        Accept $H_1$, and switch to the next larger polling interval, $C_{i+1}$.
        Set $\rho = 1$.
    end if
else
    $\rho = \rho(1-\gamma R_i^*)/(1-R_i^*)$
    if $\rho \le B$ then
        Accept $H_0$, and switch to the next shorter polling interval, $C_{i-1}$.
        Set $\rho = 1$.
    end if
end if
```

Now, we consider the sensor as the receiver which monitors channel sampling results and determines whether an unknown hypothesis is H_1 or H_0 based its observations. And the sensor decides whether to increase or decrease the polling interval, respectively, based on the test results of the hypothesis. We assume that the observations on each hypothesis are independent and identically distributed (i.i.d). We also assume that the packet arrival follows the Poisson distribution arrival model.

Let us assume that there are K polling intervals, $C_1 < C_2 < . . . < C_K$. And, we denote R_i, $1 \le i \le K$, as the idle rate of channel sampling with polling interval C_i, and E_i, $1 \le i \le K$, as energy required to transmit and receive a packet with polling interval C_i, respectively. E_i is obtained from Eq. (3) where E_s and E_r indicate power consumption at a sender and a receiver, respectively. The system parameters are shown in **Table 1**.

$$\begin{aligned} E_i &= E_s + E_r \\ &= (P_{tx}S_p + P_{rx}S_a)\left(\frac{C_i + R_l}{2(Sp + Sa)}\right) + P_{tx}S_d \\ &\quad + P_{sleep}C_i + P_{rx}R_l + P_{tx}R_a + P_{rx}R_d \end{aligned} \tag{3}$$

Let X_i denote the multiplicative inverse of E_i. This value gives a numerical interpretation to the expected number of received packets per energy unit when idle rate of channel sampling is zero.

$$X_i = \frac{1}{E_i} \tag{4}$$

SDL is an open-loop adaptation algorithm. It keeps track of the results of channel sampling to determine whether to change next polling interval that would offer better energy efficiency. the expected number of received packet per energy unit at next polling interval exceeds that at current polling interval.

For current polling interval, C_i, let the next larger polling interval be C_{i+1}. The criteria for change C_i to C_{i+1} is defined as following

$$X_{i+1} > (1 - R_i)X_i \tag{5}$$

By calculating the derivative of Eq. (5) with respect to R_i,

$$R_i^* = 1 - \frac{X_{i+1}}{X_i} \tag{6}$$

This means that when idle rate of channel sampling at C_i is larger than R_i^* , the expected number of received packets per energy unit at C_i become lower than that at C_{i+1}. Thus, SDL switches to C_{i+1}. In Eq. (5), we assume that R_{i+1} is zero, but it is likely non-zero in practice. To prevent underestimation of idle rate of channel sampling, we adopt a tunable parameter, γ , as follows.

In SDL, H_1 is defined as H_1: $R_i > \gamma R_i^*$ which indicates that energy-efficiency at polling interval C_{i+1} has been weighed to more than that at polling interval C_i, and H_0 is the alternative hypothesis, H_0: $R_i \leq R_i^*$. To determine which of them is true, the sensor monitors the results of channel sampling and compares the likelihood ratio, ρ , to two thresholds A and B.

In SDL, the sensor observes independent channel sampling results, and calculate the likelihood ratio how likely a hypothesis is true. If the sensor accepts H_1, it switches to the next larger polling interval, C_{i+1}. To decrease the polling interval, SDL must make a decision whether to switch to the next short polling interval, C_{i-1}. In our previous work [7], we studied the effect of threshold parameter, U and D, on the performance of DLPL scheme, and showed that the aggressive decrease of polling interval has the advantage of energy conservation compared to conservative approach. Thus, in our design, the sensor immediately switches to the short polling interval, C_{i-1}, after accepting H_0.

SDL updates the likelihood ratio, ρ, which is obtained by using γ. γ can be adjusted depending on how sensitive the reaction should be to dynamic change in incoming traffic loads. Note that SDL uses 1.7 as γ to response quickly to moderate changes in traffic loads.

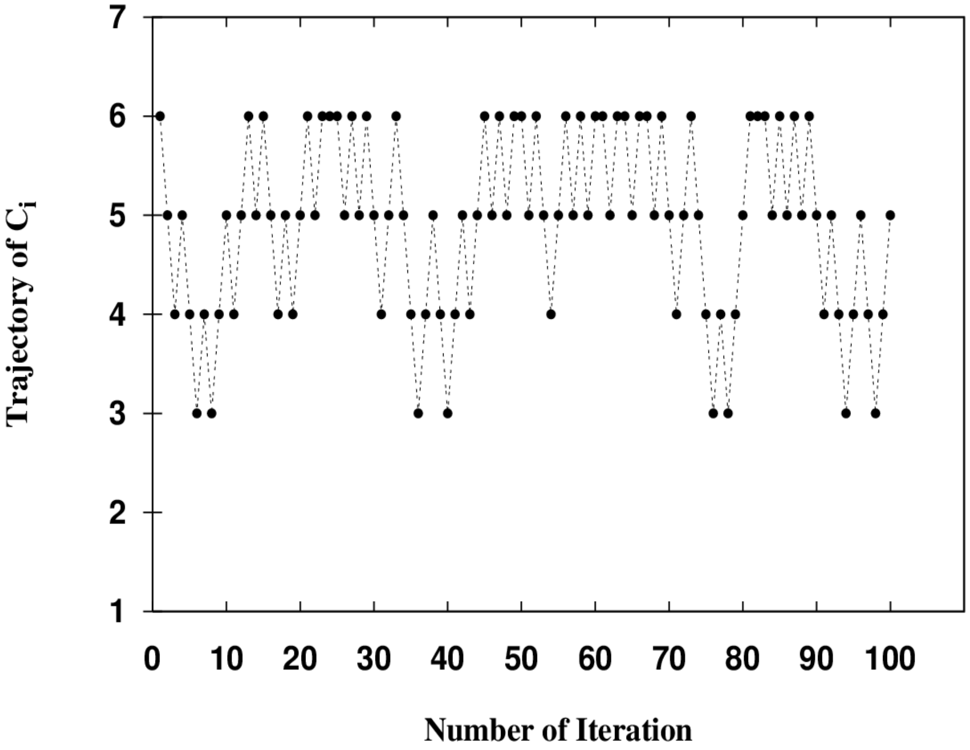


Fig. 3. Trajectory of polling interval in DLPL scheme

Discussion The intuition behind our approach is that DLPL scheme based on consecutive idle (or busy) polling may not be the right choice statistically. In **Fig. 3**, we show the trajectory of polling interval for first 100 iterations. Note that the iteration means the polling iteration of the sensor. At the starting time of each iteration, the sensor determines the polling interval, and the polling interval is not changed during each iteration. X-axis presents the number of iterations and Y-axis presents the polling interval index. Note that simulation parameters will be explained in detail in Section 5. We have following observations. First, idle polling can be misleading and triggers incorrect polling interval increase. As shown in **Fig. 3** it is difficult for DLPL scheme to reach the target polling interval due to frequent idle polling when initial polling interval is large. Thus, the sensor may stay on the large polling interval for a long time, and have poor performance. Second, frequently fluctuation makes it hard for DLPL scheme to stay the optimal polling interval. Trial-and-error based mechanism of DLPL scheme may help polling interval adaptation quickly, but it aggravates fluctuation in case of stable traffic conditions. In this paper, to tackle these issues, we propose a novel polling interval adaptation based on sequential hypothesis testing.

5. Performance Evaluation

In this section, we evaluate the performance of SDL against state-of-the-art DLPL schemes: BoostMAC and Basic DLPL scheme. BoostMAC [4] employs an AIMD (Additive Increase and Multiplicative Decrease) mechanism to control the polling interval of the sensor. If the channel is idle, BoostMAC switches C_i to C_{i+1}. On other hand, if channel is busy, it decreases C_i to $C_{i/2}$. Basic DLPL scheme uses (U, D) = (1, 1) and AIAD (Addictive Increase Additive Decrease) mechanism to control the polling interval. Note that SDL operates at only receiver-side. And hence the sensor which acts as the sender does not affect the polling interval operation.

We implemented these schemes in C language. With the detailed packet level simulations, we employ a single sender-receiver pair in order to monitor receiver's polling interval its variable traffic load. We assume that a sensor nodes has seven polling intervals ($K = 7$) of 20, 40, 80, 160, 320, 640, 1280 (*msec*). Also, packet generation at the sender follows the Poisson process with rate one (*packet* / *sec*). The simulations are done for 1000 seconds Also, both of α and β are set to be 0.05 (5%).

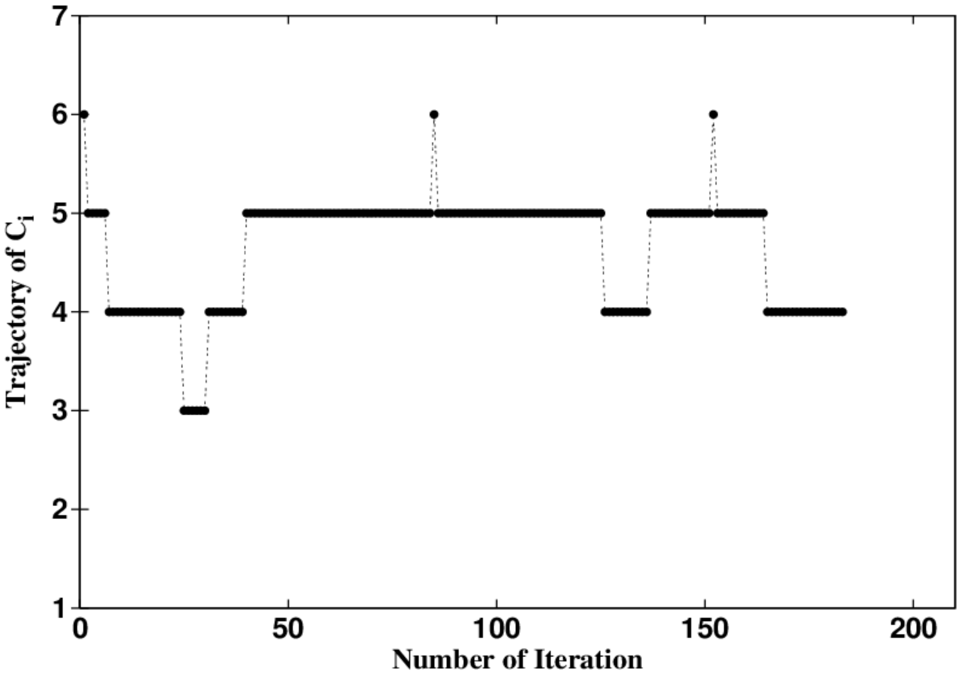
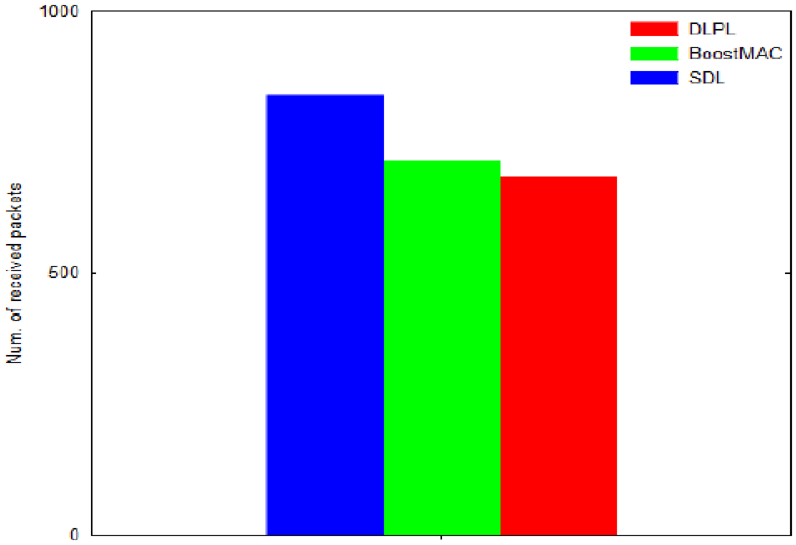
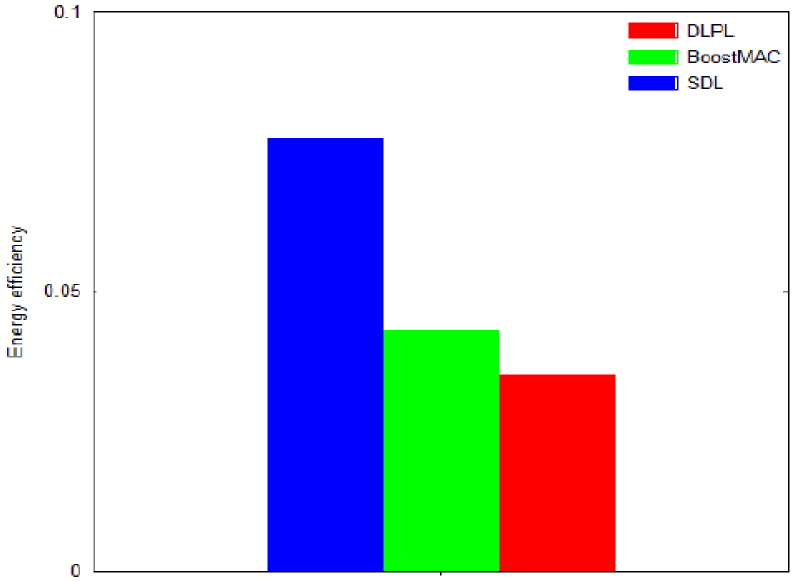


Fig. 4. Trajectory of polling interval in SDL

To illustrate the process of polling interval adaptation by SDL, **Fig. 4** provides the trajectory of the polling interval in a simulation. In SDL, the metric whether to change the polling interval is energy-efficiency. If the current polling interval cannot provide best performance, the sensor switches to a next larger polling interval. As shown in **Fig. 4**, the sensor can quickly adapt the polling interval and maintain it. Therefore, SDL can handle the polling interval fluctuation properly.



(a) Number of received packets



(b) Energy efficiency

Fig. 5. Performance evaluation

Discussion In SDL, α and β determine the confidence level of sequential hypothesis testing. Also, A and B are determined by α and β (i.e. $A = (1-\beta)/\alpha$, $B = \beta/(1-\alpha)$). SDL can provide the same level of precision by using the precise value for A and B. In SDL, ρ is updated based on results of previous channel sampling. For example, if 5 consecutive channel is idle, ρ is 1.7 * 1.7 * 1.7 * 1.7 * 1.7 = 14.19857, (initially $\rho = 1$, and $A = 14$). Therefore, SDL accept H_1, and hence switches to the next larger polling interval. For another example, if α and β are set to be 0.15% (85% confidence level), sensor changes the next larger polling interval when 4 consecutive idle.

Fig. 5(a) shows that the number of received packets of different polling interval adaptation schemes at the receiver. We observe that BoostMAC and basic DLPL scheme suffer from performance degradations due to the incompetency of capturing the polling interval fluctuations. Note that BoostMAC and basic DLPL scheme incur frequent the polling interval fluctuations due to their aggressive approach as previous mentioned in Section 1. Also, it is clear to see that BoostMAC and basic DLPL scheme provide worst performance with a slight difference, and SDL is much more than other two because SDL can provide more chance to receive the packets from the sender.

As shown in **Fig. 5(b)**, SDL is more energy efficient than BoostMAC and basic DLPL scheme while the number of received packets is approximately the same. BoostMAC and basic DLPL scheme are likely to decrease the polling interval rather than to increase it, idle listening occurs more frequently. On other hand, since SDL reacts to the result of energy efficiency measured in the current polling interval, it can select the appropriate polling interval to maintain the highest possible energy efficiency.

6. Conclusion

In this paper, we focus on the problem of polling interval adaptation, which directly affects throughput, delay and energy consumption in duty-cycled WSNs. Basic DLPL scheme suffers from polling interval fluctuation due to the trial-and-error based increase/decrease mechanism. In order to tackle this problem, we propose a novel DLPL scheme, called SDL, for duty-cycled WSNs. The basic idea of SDL is to estimate the results of channel sampling, and adaptively change the polling intervals using sequential hypothesis testing. Simulation results show that SDL increase energy efficiency compared with state-of-the-art DLPL schemes.

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet of Things Journal*. vol.1, no. 1, pp.22–32, 2014. Article (CrossRef Link)

[2] J. Polastre, J. Hill and D. Culler, “Versatile low power media access for wireless sensor networks,” in *Proc. of ACM Sensys*, 2004. Article (CrossRef Link)

[3] M. Buettner, G. V. Yee, E. Anderson and R. Han, “X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks,” in *Proc. of ACM Sensys*, 2006. Article (CrossRef Link)

[4] K. Stone and M. Colagrosso, “Efficient duty cycling through prediction and sampling in wireless sensor networks,” *Wireless Commun. and Mobile Computing*, vol. 7, no. 9, pp.1087–1102, 2007. Article (CrossRef Link)

[5] W. Ye, J. Heidemann and D. Estrin, “An energy efficient mac protocol for wireless sensor networks,” in *Proc. of IEEE INFOCOM*, 2002. Article (CrossRef Link)

[6] S. Lee, J. Park and L Choi, “AMAC: Traffic-adaptive sensor network MAC protocol through variable duty-cycle operations,” in *Proc. of IEEE ICC*, 2007. Article (CrossRef Link)

[7] S. Lee, J. Choi, J. Na and C. Kim, “Analysis of dynamic low power listening schemes in wireless sensor networks,” *IEEE Comm. Letters*, vol. 13, no. 1, pp.43–45, 2009. Article (CrossRef Link)

[8] P. Hurni and T. Braun, “Maxmac: A maximally traffic-adaptive MAC protocol for wireless sensor networks,” *Wireless Sensor Networks*, vol. 5970, pp. 289–305, 2010. Article (CrossRef Link)

[9] C. Merlin and W. Heinzelman, “Network-aware adaptation of MAC scheduling for wireless sensor networks,” in *Proc. of DCOSS(Poster Session)*, 2007.

[10] C. Merlin and W. Heinzelman, “Schedule adaptation of low-power listening protocols for wireless sensor networks,” *IEEE Trans. on Mobile Computing*, vol. 9, no. 5, pp. 672–685, 2010. Article (CrossRef Link)

[11] C. Merlin and W. Heinzelman, “Duty cycle control for low-power listening MAC protocols,” *IEEE Trans. on Mobile Computing*, vol. 9, no. 11, pp.1508–1521, 2010. Article (CrossRef Link)

[12] M. Avvenuti, P. Corsini, P. Masci and A. Vecchio, “Increasing energy eficiency of a preamble sampling MAC protocol for wireless sensor networks,” in *Proc. of IEEE Mobile Computing and Wireless Comm. Int’l Conf.*, 2006. Article (CrossRef Link)

[13] K. Wong and D. Arvind, “SpeckMAC: Low-Power Decentralised MAC Protocol Low Data Rate Transmissions in Specknets,” in *Proc. of Second IEEE Int’l Work shop Multi-Hop Ad Hoc Networks: From Theory to Reality (REALMAN ’06)*, 2006. Article (CrossRef Link)

[14] H. Byun and J. So, “Queue-based adaptive duty cycle control for wireless sensor networks,” in *Proc. of int’l conference on Algorithms and architectures for parallel processing*, 2011. Article (CrossRef Link)

[15] H. Byun and J. Yu, “Queue management based duty cycle control for end-to-end delay guarantees in wireless sensor networks,” *IEEE Trans. on Mobile Computing*, vol. 12, no. 6, pp. 1214–1224, 2013. Article (CrossRef Link)

[16] A. Wald, Sequential Analysis, J. Wiley & Sons, New York, 1947. Article (CrossRef Link)

[17] J. Jung, V. Paxson, A. W Berger and H. Balakrishnan, “Fast Portscan Detection Using Sequential Hypothesis Testing,” in *Proc. of IEEE Symp. Security and Privacy*, 2004. Article (CrossRef Link)

[18] J. Ho, M. Wright, S. K. Das, “Fast Detection of Mobile Replica Node Attacks in Wireless Sensor Networks using Sequential Hypothesis Testing,” *IEEE Trans. on Mobile Computing*, vol. 10, no. 6, pp.767–782,2011. Article (CrossRef Link)

[19] Y. Rong , A. Teymorian , L. Ma , X. Cheng and H. Choi, “A novel adaptation scheme for 802.11 Networks,” *IEEE Trans. Wireless Commun.*, vol. 8, no. 2, pp.862–870, 2009. Article (CrossRef Link)

Sungryoul Lee received his B.S. and M.S. from Dept. of Computer Science, Sogang University, Korean, in 2001 and 2003, respectively, and his Ph.D. from the School of Electrical Engineering and Computer Sciences, Seoul National University, Korea, in 2010. Now, he works for the Attached Institute of ETRI in Korea. His research interests include Power Saving Technologies for Wireless Networks, Network Security, Information Security.