Analysis on Event Delay in Event-Driven Wireless Sensor Networks

Rong Zhou
Department of Computer Science & Technology
Shanghai Normal University
Guilin Road 100#, Shanghai, China
Email: zrskyline@gmail.com

Luqun Li
Department of Computer Science & Technology
Shanghai Normal University
Guilin Road 100#, Shanghai, China
Email: liluqun@gmail.com

Jun Zhang
Department of Computer Science & Technology
Shanghai Normal University
Guilin Road 100#, Shanghai, China

***Abstract*—In this paper, we consider an event-driven wireless sensor network (WSN) scenario, in which the events arrive with a certain probability in a time of interval. Firstly, we analyze the event delay in a single node in an M/G/1 vacation queueing model. Then, considering the limited buffer space, channel contention and the possibility of a link failure, we derive the total average event delay through a route path. Finally, we build an OMNeT++ simulation and verify our theoretical research results. This research work should provide a guidance for understanding the event delay and future designing more flexible event-driven WSN routing protocols.**

Keywords-Wireless sensor networks (WSNs), Event-driven, Event delay, Vacation queueing model

I. INTRODUCTION

Wireless sensor networks (WSNs) constitute a major trend in modern networking, which have many applications that are difficult for traditional network such as deep space communication, digital content delivery in rural areas with under-developed infrastructure, wildlife and habitat monitoring [1, 2]. However, some intrinsic limitation, e.g., the limited available power, weight, and memory size, and also the uncertain ad hoc deployment, may seriously affect the network performance. Typically, all sensor nodes in a WSN are always scheduled between working state and sleeping state for reducing energy consumption. As an efficient and effective way, event-driven techniques are developed to minimize the energy consumption and guarantee timely delivery as it works only if an event of interest arrives [3]. Like the traditional network, event delay is also a very important metric for the quality of service (QoS) of a network, which is thought to be influenced by the nodal behavior.

Compared to the traditional wired and wireless networks, however, event-driven WSNs have two major distinctive characteristics, namely, unpredictable environment conditions and specific nodal service. These constraints increase the complexity of the efforts that measure the performance of the event-driven WSNs. For example, large amounts of redundant sensors are deployed into monitoring region to guarantee timely detecting the event of interest, and it is impractical to deploy sensor nodes at predefined locations. So, the network topology and data communication path may be changed according to the varied environmental condition, the analytical methods applied on the traditional network are not suitable for such networks. Also, as an event comes from individual nodes can be unreliable; it is desirable to receive the event from multiple sensor nodes. Even so, it is pretty much possible that a node may not receive any events in a long time. That is to say, event arrival is a discrete randomness process, which determined by several factors such as network coverage, topology, and wakeup scheme. When there is not an event to be received, the node should be switched into sleeping state because it is in idle state [4]. A sleeping node can be activated into working state within an event arrival probability. This specific nature of WSNs is a key fundamental factor in event-driven techniques and thus can significantly influence the event delay in the whole network. Different event delay in a node can lead to different battery energy consumption and network latency. The related research works did not do comprehensive study on event delay. In this paper, we will focus on the event delay in event-driven WSNs. We assume an event arrives within a certain probability p ($0 \le p < 1$) in a time interval and build an M/G/1 vacation queueing model based on queuing theory to analyze the event delay in a node that change with varying p. Moreover, considering the limited buffer size, channel contention and the possibility of a link failure, we derive the total average event delay through an end-to-end route path. The research results should provide guidance for us to design more flexible event-driven routing protocol.

II. RELATED WORK

Many related research works on event delay in the event-driven WSNs have been done in past years. Wang et al [5] assumed an event can be detected successfully with a probability p in working state, and used a probabilistic approach to analyze the detection event delay under the *any-sensor-detection* model and the *k-sensor-detection* model respectively. Bouabdallah et al [6] analyzed the latency-lifetime tradeoff in the WSN with multiple source nodes, and thus derived the optimal number of source node under several limited factors. Gribaudo et al [7] built a detailed model to obtain the delivery delay distribution of events sent by concurrently contending sensors toward a central controller and carried out a transient analysis for delay performance. Huang et al [8] proposed an effective mathematical analysis methodology to be applied to real-time WSN under the IEEE

802.15.4 slotted Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) mechanism. The results suggested the slotted CSMA/CA mechanism cannot be applied effectively to event-driven WSNs and will waste a great deal of bandwidth as the event collision rate will increase up largely. Wang et al [9] developed a spatio-temporal fluid model for the detection event delay in large-scale WSNs to meet the requirement of real-time operation. By using the model, mean event delay and soft event delay bounds under different network parameters can be obtained. These research works shared an assumption that all events come from several source nodes and their researches are based on different network scenarios or network protocols.

We have to notice that, however, the complete research on event delay in event-driven techniques was not considered. Event-driven WSNs often maintain one or more data transmission path to the sink node, which usually is end system or base station. Before the base station receives the events, we should pay attention to many unpredicted limitations such as the intermittent network connectivity and transmission path will impact the network performance. Therefore, we construct a generic and universal end-to-end router path to explain the principle that data transmission in this paper (see Figure 1). As a result, the above research works are not suitable for such network.

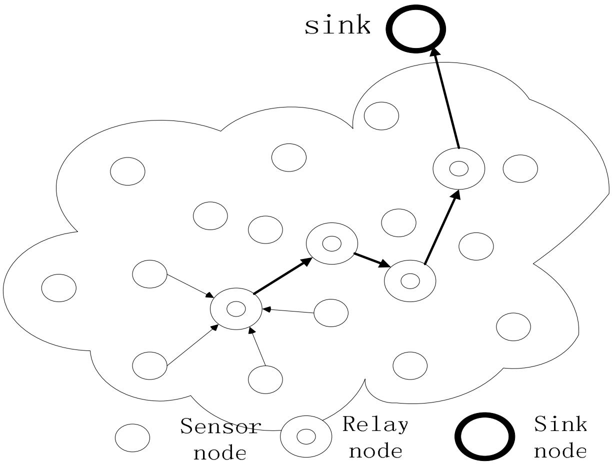

Figure 1. Typical data transmission path in WSNs.

III. PROBLEM STATEMENT

In this paper, we consider an end-to-end event transmitting route in WSN scenario where the source node delivers event s to the destination node through several intermediate nodes. The links among nodes may be in intermittent connection due to interference, obstacles and channel contention etc… Events may be dropped because of the node's finite storage capacity and bandwidth. Hence, data communication is implemented by "store-and-forward" [10] network architecture. A node will generate event (data picking) or receives events from its neighbors and then forwards them to the next hop node. We assume that the event packets arrived at a sensor node in the Poisson distribution.

Supposing that there are several kinds of packets with different priorities arriving at a node, we can utilize several queues to manage different kind of packets according to their priority and the buffer manage strategy is First Input First Output (FIFO). For simplicity, the buffer contains two queues, the high priority queue (Q_h) and the low priority queue (Q_l). All the arriving packets are injected into either of the two queues in terms of their priority. Packets with high priority are forwarded preferentially. Figure 2 shows the queueing system of a node in wireless sensor network.

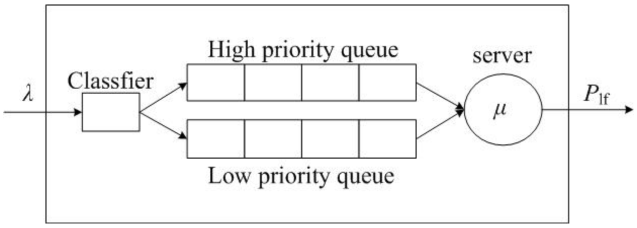

Figure 2. Queueing system of a node in a wireless sensor network.

The event may arrive at a node in a time interval with a probability p. The time is divided into time slots. The nodal service is regulated by the probability of event arrival (Figure 3). Each node decides whether into working state with probability p or in sleeping state with probability $1-p$ at the end of a time slot, according to the probability of event arrival.

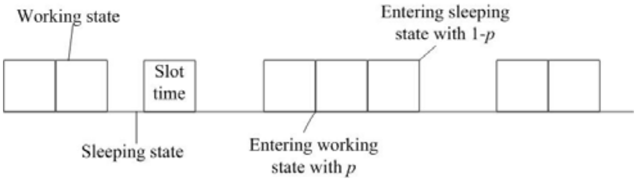
Working state
Slot time
Entering sleeping state with 1-p
Sleeping state
Entering working state with p

Figure 3. State change of a node

Our further research works are under the following assumption:

(1) Each node receives packets from its neighbors in Poisson distribution.

(2) In sleeping state, the sensor node does nothing, which means the node neither receives nor forwards packets. While in working state, the node receives and forwards packets.

(3) When a node is serving a packet, it cannot be disrupted by a new packet or switch to sleeping state.

IV. ANALYSIS OF EVENT DELAY

A. Analysis of the Delay in a Single Node

First of all, we initiate our analysis by modeling individual node behavior in WSNs. Packets arrival is in Poisson distribution, and the packet service process follows General distribution. When a packet arrives at a sensor node, if the node is in sleeping state (so called "service in vacation") or there are any other packets in the buffer, it must be stored in the buffer and therefore produce a delay. We denote the packet arrival rate as λ_i. As the packets have different priorities and should be dispatched into one of the two queues Q_{h} and Q_{l}, we assume the packets are sent into Q_{h} with P_{d}. Service time in different queue is different. p is a probability that a node is triggered to serve packets into the next slot time (if one of the queues is not empty) when the former slot time ends.

The service time of high priority queue $\mu_{i,\mathrm{h}}$ follows a general distribution with the mean $\overline{x_{i,\mathrm{h}}}$ and the second moment $\overline{x_{i,\mathrm{h}}^2}$, and low priority queue has service time $\mu_{i,\mathrm{l}}$ that follows a general distribution with the mean $\overline{x_{i,\mathrm{l}}}$ and the second moment $\overline{x_{i,\mathrm{l}}^2}$. Define traffic intensity (or utilization factor) for each queue as $\rho_{i,\mathrm{h}} = \dfrac{\lambda_{i,\mathrm{h}} P_{\mathrm{d}}}{\mu_{i,\mathrm{h}}}$, and $\rho_{i,\mathrm{l}} = \dfrac{\lambda_{i,\mathrm{l}}(1-P_{\mathrm{d}})}{\mu_{i,\mathrm{l}}}$. Then, the total traffic intensity of individual node is $\rho_i=\rho_{i,\mathrm{h}}+\rho_{i,\mathrm{l}}$. Since the residuals of a nodal successive vacation time are mutually independent and identically distributed. We denote their mean to be $\overline{V_i}$ and their second moment to be $\overline{V_i^2}$.

In order to obtain the total waiting time in node W, a residual service time R is introduced. According to Ref. [11], $W=R/(1-\rho)$. In vacation queuing theory, R can be decomposed into two parts as follows.

(1) The mean residual service time in working state R_s, it is given by follows:

$$R_s = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{2}(\lambda_{i,j}\sigma_{i,j}{}^2 + \frac{\rho_{i,j}}{\mu_{i,j}}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{2}\lambda_{i,j}\overline{x_{i,j}{}^2}$$

(2) The mean residual service time in sleeping state for all packets R_v.

$$R_v = \frac{1}{t}\sum_{i=1}^{N}\sum_{j=1}^{2}\frac{1}{2}\overline{V_i^2}L(t)$$

where $L(t)$ is the number of arriving packets in sleeping state.

When $\lim\limits_{t\to\infty}\dfrac{L(t)}{t} = \dfrac{1-\rho_{i,j}}{\overline{V_i}}$, the $R_v = \sum\limits_{i=1}^{N}\sum\limits_{j=1}^{2}(\dfrac{(1-\rho_{i,j})\overline{V_i^2}}{\overline{V_i}})$.

Hence, the total service residual time R is:

$$R = R_s + R_v = \sum_{i=1}^{N}\sum_{j=1}^{2}[\frac{1}{2}\lambda_{i,j}\overline{x_{i,j}{}^2} + \frac{(1-\rho_{i,j})\overline{V_i^2}}{2\overline{V_i}}]$$

where N is the hop count through an end-to-end transmitting path, j represents the Q_{h} or Q_{l}.

Since a node enters the sleeping state with a probability $1-p$ when a slot time ends, we utilize the probability scheduling system to model and analyze it. After a slot time is finished, the system decides whether to serve or not in the next slot time. As a result of this nature, we can take the node as a multiple vacation time system. Each queue has service time $\mu_{i,j}^{'} = \mu_{i,j} + \xi$, the mean is $E(u_{i,j}^{'}) = E(\mu_{i,j} + \xi) = \overline{x_{i,j}} + (1-p)\overline{V_i}$, and the second moment is

$$E(u_{i,j}^{'2}) = E(\mu_{i,j} + \xi)^2 = \overline{x_{i,j}^2} + 2\overline{x_{i,j}}(1-p)\overline{V_i} + (1-p)\overline{V_i^2}.$$

Therefore, the new total residual service time is

$$R^{'} = \sum_{i=1}^{N}\sum_{j=1}^{2}\frac{1}{2}\{\lambda_{i,j}[\overline{x_{i,j}{}^2} + 2\overline{x_{i,j}}(1-p)\overline{V_i}] + (1-p)\overline{V_i} + \frac{(1-\rho_{i,j}\overline{V_i^2})}{2\overline{V_i}}\}.$$

By enforcing non-preemptive priority queueing, we set multiple queues to store different packets, different packets have different priority. According to the total wait time formula $W = R^{'}/(1-\rho)$, we can calculate average wait time in each queue separately, such as average wait time in high priority queue $W_{i,\mathrm{h}} = R^{'}/(1-\rho_{i,\mathrm{h}})$, average wait time in low priority queue $W_{i,\mathrm{l}} = R^{'}/[(1-\rho_{i,\mathrm{h}})(1-\rho_{i,\mathrm{l}})]$.

Summarizing above analysis, we derive the average event delay of a node in each queue as follows:

$$T_{i,\mathrm{h}} = W_{i,\mathrm{h}} + \overline{x_{i,\mathrm{h}}}\ ,\ T_{i,\mathrm{l}} = W_{i,\mathrm{l}} + \overline{x_{i,\mathrm{l}}}$$

Then, the event delay (T_i) of packet spent in a sensor node on average is given by:

$$T_i = \frac{T_{i,\mathrm{h}} + T_{i,\mathrm{l}}}{2} \tag{1}$$

B. Analysis of the Delay through a Route Path

Having known the average event delay (T_i) in each sensor node, now we begin to study the total average event delay through a route path. The block probability, which can be used to predict the packet loss, is a critical parameter in performance evaluation. Assuming the buffer in each node is a predetermined value K_i, we can obtain the block probability of each node $P_{\mathrm{b}i}$ [12].

$$P_{\mathrm{b}i} = \frac{\rho_i^{a_i} b_i}{\rho_i^{2a_i} - 1} \tag{2}$$

Where $a_i = (2 + \sqrt{\rho_i}s^2 + 2K_i)/(2 + \sqrt{\rho_i}s^2 - \sqrt{\rho_i})$, $b_i = \rho_i - 1, s^2$ is service process coefficient. In this paper, we set $s^2 = 0.04$.

The potential packet loss probability (P_{l}) in network can be estimated by the block probability. Packet loss occurring relies on two possibilities, buffer fullness and channel contention. If an arrival packet (both of the high priority packet and the low priority packet) does not encounter the full buffer, it will be lost with a contention probability P_c; on the contrary, when it meets with a full buffer, it will be lost with block probability P_{bi}. Therefore, according to Ref. [13], we can obtain the potential packet loss probability at i-th link:

$$P_l^i = P_c \times (1 - P_{bi}) + P_{bi} \tag{3}$$

Clearly, as we have known the number of hops between source node and destination node N, the potential packet loss probability P_{path} on the routing path can be calculated as follows:

$$P_{\text{path}} = 1 - \prod_{i=1}^{N-1} (1 - P_l^i) \tag{4}$$

Generally, after delivering a packet to another node the send node has to wait ACKnowledgement (ACK) or Negative ACKnowledgement (NACK) message to determine whether transmitting is successful or not, otherwise timeout. Therefore, we need to assign an interval [1, k] [14] to wait ACK or NACK message after sending a packet. Taking consideration of the packet loss probability, the average time spent by retransmitting on a path is:

$$T_{\text{re}} = P_{\text{path}} \left(\sum_{i=1}^{N-1} \frac{2L_i}{C} + \frac{k+1}{2} \right) \tag{5}$$

where C is the radio speed.

Similarly, when the link of two nodes disconnected with a probability P_{lf}, which caused by physical factors such as the moving obstacle (in this scenario, we assume the link will be connected sooner or later and P_{lf} is a constant), propagation delay can be estimated by:

$$T_{\text{p}} = \sum_{i=1}^{N-1} \frac{L_i}{C} (1 - P_{\text{lf}})^i \tag{6}$$

Integrating Eqs. (1), (5), and (6), the total average event delay through a route path in WSNs is given as follows:

$$T_{\text{delay}} = \sum_{i=1}^{N} T_i + T_{\text{re}} + T_{\text{p}} \tag{7}$$

V. SIMULATION AND NUMERAL RESULTS

We build our simulation with OMNeT++ [15]. The service scheduling of a node depends on the probability of event arrival. In the view of event delay in a node, it is obvious that the best case is p = 1 because the node works without vacation and packets are delivered continuously, but it consumes much of energy (in this moment, the best technique should be time-driven); while the worst case is p = 0, it means the node is switched between sleeping state and working state frequently. Nevertheless, there is still an event delay as the arrival rate and service rate of packets are different. It is why the works of QoS control emphasize on reducing event delay as small as possible rather than eliminating it.

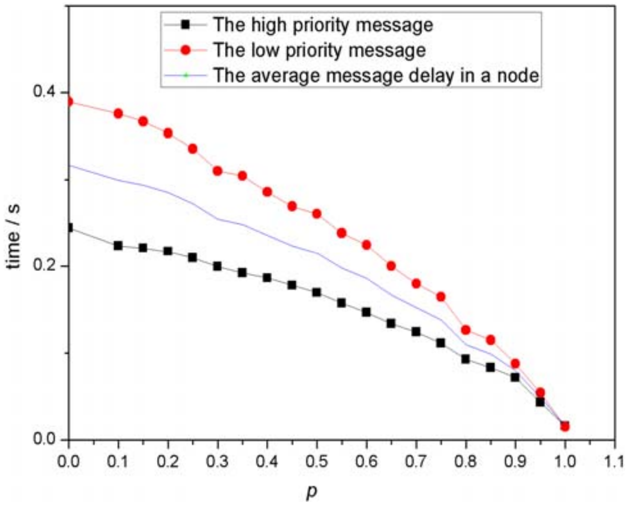

Figure 4. Event delay in each queue and in a node versus the probability p

We initiate our simulation with λ_i = 4, $\mu_{i,\text{h}}$ = 10, $\mu_{i,\text{l}}$ = 5, P_{d} = 0.6, the slot time equals to 0.1 s, and the sleeping time follows exponential distribution with parameter 1.0 s. As shown in Figure 4, the intermediate line is the average event delay in a node and the rest of the two lines represent the average delay of event with different priorities in buffer respectively. They decrease as the probability of event arrival p increases. It indicates that the event delay in queues and in a node is sensitive to the probability of event arrival.

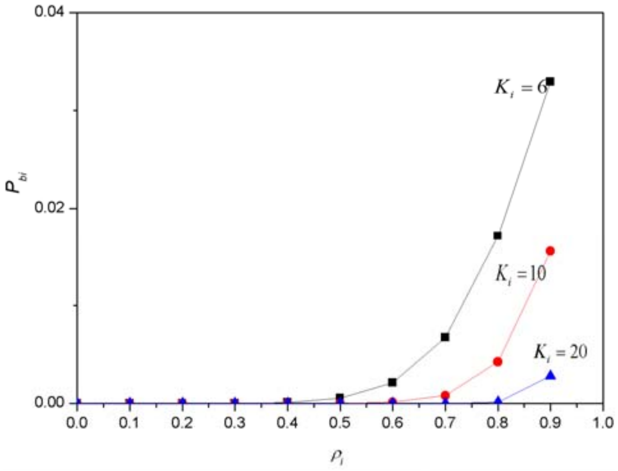

Figure 5. Block probability vs. traffic intensity

The main reason that packet loss occurs is the buffer fullness. We investigate the relationship between traffic intensity and the block probability due to the retransmitting time. According to Eq. (2), we can obtain the result as shown in Figure 5. When ρ_i is greater than 0.45, the P_{bi} will increase obviously if K_i has been known. Meanwhile, if the value of K_i is greater, the block probability also becomes greater. As K_i is usually limited, the approach to avoid packet loss should emphasize on regulating traffic intensity ρ_i. It suggests that the packet loss is associated with the traffic intensity and buffer.

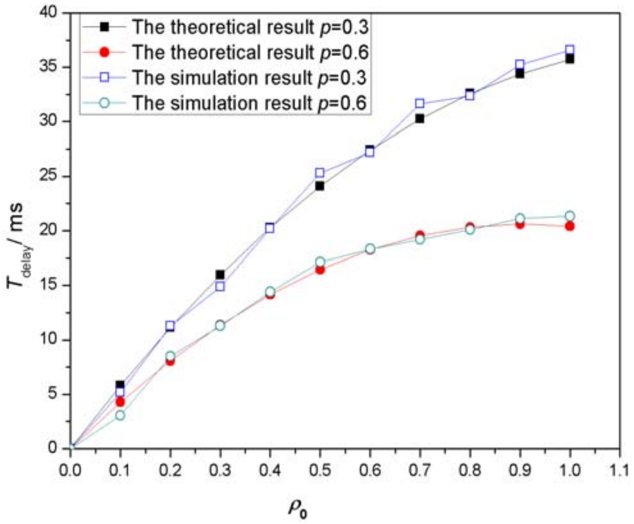

Figure 6. Average event delay obtained by the model and simulation under different probabilities.

At last, we utilize a simulation tool on OMNeT++ to verify our theoretical analysis, which is called MiXiM [16]. We employ Simple Path-Loss Model [15] as wireless channel attenuation model, the carrier frequency is 2.4 GHz, and the media access control (MAC) layer is the carrier sense multiple access (CSMA) scheme. We use a base scenario, where 30 hosts are deployed in 500 m×500 m region in uniform distribution, and the hop count between source node and destination node is 8 hops. In order to evaluate the average event delay through a route path, we increase the traffic intensity of source node (denoted by ρ_0) step by step from 0.1 to 1 over several simulations. We set p=0.3 and p=0.6, respectively, to compare the results under different probabilities. From Figure 6, we conclude if the p is smaller the delay is also smaller; moreover, the event delay increases largely as ρ_0 increases. The simulation results show that our analysis result is suitable for such kind of event-driven wireless sensor networks.

VI. CONCLUSION AND FUTURE WORKS

This paper focuses on the event delay in event-driven WSN data transmission, and builds an M/G/1 vacation queueing model with multiple priorities. We get the event delay in a single node that changes with varying probability of event arrival, find the block probability that related to the packet loss is determined by the traffic intensity and buffer size, and also derive the total average event delay through a route path under several limitations such as channel contention. The results showed that the probability of event arrival significantly affects the event delay in a node.

To simplify our research, we only consider the arrival probability of an event is a constant. It is widely acknowledged the arrival probability of an event should be followed by certain distribution as the randomness of an event occurring. So the next work we would perform is to analyze the event delay when the arrival probability of an event is Gauss distribution or Normalized distribution. These research works should provide a guidance for understanding the event delay and future designing more flexible event-driven WSN routing protocols.

ACKNOWLEDGMENT

This work has been partially supported by the grants from Shanghai Education Bureau Grant No.11YZ93, Grant No.A-3101-10-035 and National Natural Science Foundation of China Grant No. 61003215.

REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: A survey,” Computer Networks, vol. 38, no. 4, pp. 393–442, December 2002.

[2] M. Kuorilehto, M. Hannikainen, and T.D. Hamalainen, “A survey of application distribution in wireless sensor networks,” EURASIP Journal on Wireless Communications and Networking, vol. 2005, no. 5, pp. 774–788, March 2005.

[3] K. Akkaya and M. Younis, “A survey on routing protocols for wireless sensor networks,” Ad Hoc Network, vol. 3, no. 3, pp. 325–349, November 2005.

[4] H. Le, H. Guyennet, V. Felea, and N. Zerhouni, “A low latency MAC scheme for event-driven wireless sensor networks,” Mobile Ad-Hoc Sensor Network, vol. 4864, pp. 291–301, 2007.

[5] Y.C. Wang, K.Y. Cheng and Y.C. Tseng, “Using event detection latency to evaluate the coverage of a wireless sensor network,” Computer Communications, vol. 30, no.14–15, pp. 2699–2707, October 2007.

[6] F. Bouabdallah, N. Bouabdallah, and R. Boutaba, “Analysis of the latency-lifetime tradeoff in wireless sensor networks,” in Proceedings of the 2008 IEEE/ACS International Conference on Computer Systems and Applications, pp.1076-1081, March 31-April 04, 2008.

[7] M. Gribaudo, D. Manini, A. Nordio, and C. Chiasserini, “Analysis of IEEE 802.15.4 sensor networks for event detection,” in Proc. IEEE Globecom ’09, pp. 152–157, Honolulu, Hawaii, Nov 2009.

[8] W.T. Huang, J.T. Lin, C.H. Chen, Y.J. Chang, and Y.Y. Chen, ”Accurate Analysis of IEEE 802.15.4 Slotted CSMA/CA over a Real-Time Wireless Sensor Network”, Lecture Notes of the ICST, Vol 24, pp: 98-110, 2010.

[9] C.F. Chiasserini and M. Garetto, “Modeling the performance of wireless sensor networks,” in Proceedings of the 23th IEEE International Conference on Computer Communications (INFOCOM ’04), Hong Kong, China, pp. 220–231, Mar 2004.

[10] S. S. Lam, "Store-and-forward buffer requirements in a packet-switching network," IEEE Transactions on Communication, vol. 24, No. 4, pp. 394–403, Apr 1976.

[11] D.P. Bertsekas and R.G. Gallager, Data Networks, 2nd ed. Englewood Cliffs, New Jersey: Prentice Hall, 1992, ch. 3.

[12] J. MacGregor Smith, “M/G/c/K blocking probability models and system performance,” Performance Evaluation, vol. 52, no. 4, pp. 237–267, May 2003.

[13] M. Baharloo, R. Hajisheykhi, M. Arjomand, and A.H. Jahangir, “An Analytical Performance Evaluation for WSNs Using Loop-Free Bellman Ford Protocol,” in Proceedings of AINA-2009, Bradford, UK, pp. 568–571, May 2009.

[14] K. Mizanian, and A. H. Jahangir, “A Quantitative Real-time Model for Multi-hop Wireless Sensor Networks,” Third International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP’07), Melbourne, Australia, pp. 79–84, December 2007.

[15] OMNET++, http://www.omnetpp.org/.

[16] A. Kopke, et al., “Simulating wireless and mobile networks in omnet++: The MIXIM vision,” SIMUTools, Brussels, March 2008.