

# Reliability Measure Model for Assistive Care Loop Framework Using Wireless Sensor Networks

Venki Balasubramanian and Doan. B. Hoang

*iNEXT - Centre for Innovation in IT Services and Applications*

*Faculty of Engineering and Information Technology*

*University of Technology, Sydney,*

*PO Box 123, Broadway 2007, NSW, Australia*

*{vsubru, dhoang}@it.uts.edu.au*

## ABSTRACT

Body area wireless sensor networks (BAWSNs) are time-critical systems that rely on the collective data of a group of sensor nodes. Reliable data received at the sink is based on the collective data provided by all the source sensor nodes and not on individual data. Unlike conventional reliability, the definition of retransmission is inapplicable in a BAWSN and would only lead to an elapsed data arrival that is not acceptable for time-critical application. Time-driven applications require high data reliability to maintain detection and responses. Hence, the transmission reliability for the BAWSN should be based on the critical time. In this paper, we develop a theoretical model to measure a BAWSN's transmission reliability, based on the critical time. The proposed model is evaluated through simulation and then compared with the experimental results conducted in our existing Active Care Loop Framework (ACLF). We further show the effect of the sink buffer in transmission reliability after a detailed study of various other co-existing parameters.

**Keywords:** body area wireless sensor networks; reliability; assistive care; remote health monitoring systems

## 1. INTRODUCTION

Body area wireless sensor networks (BAWSNs) consist of a number of wireless sensors located on or in close proximity to the human body, such as on the clothing. The low-power sensors are equipped with a wireless interface and are capable of sensing the required intrinsic health data of the person over an extended period of time. In addition, these sensors can send data to a personal device, such as a personal digital assistant (PDA) or a smartphone, via the sink or base station, so that the data can be used for any health care application. In this process, the "sink" is a data aggregation point that collects the measured data from the surrounding sensors and sends them to an end-point of the application. In recent years, these capabilities have allowed a BAWSN to be deployed in many time-critical health care monitoring applications [1–6]. In time-critical health care monitoring applications, it is imperative that the end-point receives

the monitored data within the expected time period. The success of these time-critical applications not only depends upon the reliable delivery of error-free data to the end-point, but also upon the episodic delivery of the data by the underlying wireless sensor networks. Therefore, these applications should be evaluated based on the reliable delivery of the data at regular time intervals.

### 1.1. Conventional Reliability

In conventional reliability<sup>1</sup>, the transport service has little knowledge about the semantics of the data; therefore the reliability solutions are based on “per transport message segment” [7, 8]. In such transport solutions, end-to-end reliability ensures that data are received by the intended end-point successfully using retransmission and/or other error control techniques. However, a conventional reliability would only involve the reliable delivery of the data using retransmission that may constitute a redundant delivery of the data. Much of the prior work on reliability for wireless sensor networks addresses similar message-level reliability for both upstream (sensor-to-sink) and downstream (sink-to-sensor) reliable data delivery. For instance, the *Pump Slowly, Fetch Quickly* (PSFQ) [9] approach is proposed for downstream reliable data delivery from the source to the sensor nodes. It is based on the hop-by-hop error recovery with in-network caching and sending the repair request via a negative acknowledgement (Fetch) that is faster than the source transmission rate (Pump). In *GARUDA* [10], the downstream message reliability is achieved by a virtual infrastructure composed of local and designated loss recovery servers. *Reliable Multi-Segment Transport* (RMST) [11] is a transport layer protocol designed to run in conjunction with directed diffusion. It is a selective NACK-based message-level reliability scheme used for the transference of large amount of data from sensors to the sink.

### 1.2. BAWSN Reliability

An underlying wireless sensor network with a sensor-to-sink (aggregation point) and a sink-to-sensor interaction constitutes only part of a real-time application. To build a real-time application, the underlying wireless sensor network should be extended with an end-point for further processing of the data. However, an extended wireless sensor network often consists of many heterogeneous components [1–6], a BAWSN is one such application extensively used for in-house health monitoring with PDAs or smartphones. The existing approaches [9–11] work well in delivering error-free data segments from the sensor to the aggregate point or vice-versa by using retransmission or other error-control techniques over a data-exchange session. However, the performance of the message-level reliability for wireless sensor networks may not be appropriate for a BAWSN application, because data are delivered from the sensor to the end-point via an intermediate aggregation point. Moreover, in time-critical applications such as BAWSNs, the arrival of an error-free event data to the end-point later than the critical expected time is considered as futile or unreliable. Therefore, in

---

<sup>1</sup>The term “conventional reliability” is used to refer to the TCP reliability in which continuous streams of data split into multiple data packets, with each data packet carrying a fragment of the continuous stream.

such applications, the measure of reliability using the message-level reliability methods may not be suitable.

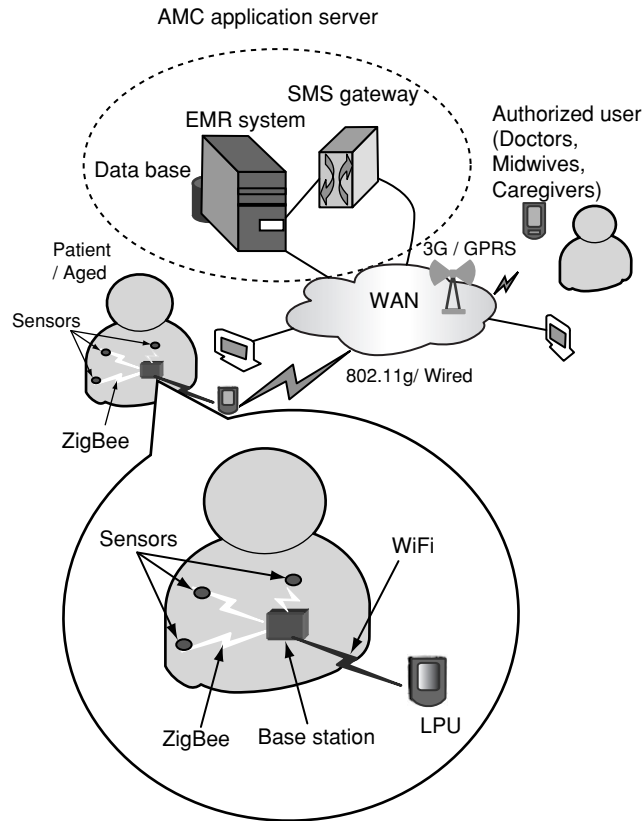
The main objective of this paper is to present an approach for measuring the reliability of a time-critical BAWSN application. Based on this prerequisite for a time-critical BAWSN application, we define reliability *as the capability of the BAWSN to send error free data in a required time interval*. To evaluate such capability, an effective measure of reliability for any time-critical BAWSN application is needed. This paper addresses such a need for measure by proposing a reliability measure model, which is demonstrated using our existing Active Care Loop Framework (ACLF) [1]. The ACLF can be used for any types of BAWSN application to monitor the health conditions of the aged and/or the sick by means of sensors. These sensors include, but are not limited to, medical, wearable, mobile and fixed sensors, depending on the disease or needs [1]. In developing our reliability model, various parameters were investigated and a preferred set of three parameters crucial for determining the reliability of the ACLF were selected. The proposed model was simulated and compared with the experimental results obtained with our ACLF. The comparison results show that the selected set of parameters is indeed crucial in determining the reliability of the ACLF, and varying these parameters would significantly influence the reliability.

The remainder of this paper is organized as follows. A brief introduction to the ACLF is presented in Section 2, followed by a rationalization of the selected parameters in Section 3. The experimental setup and results are detailed in Section 4. The derivations of the reliability model and simulation analysis are presented in Section 5. Finally, Section 6 concludes the paper.

## **2. ACTIVE CARE LOOP FRAMEWORK (ACLF)**

The ACLF extends our earlier work on the Assistive Maternity Care (AMC) application [2], shown in Figure 1, which also demonstrates the details of the sensory part. Currently, paper-based health records are maintained for pregnant women at most of the health services in New South Wales, Australia. Paper-based records have a number of disadvantages, including the poor use of space, fragility, being limited to a single user at all times, vulnerability of records being misplaced or lost and the effort required to search for information from either a large single record or a collections of records [2]. Hence, we have developed an AMC application (pilot) by establishing an Electronic Maternity Record (EMR).

The AMC application enables practitioners (such as doctors and midwives) and pregnant women to access health records from beyond the physical location of a hospital zone not only through desktop computers, but also through PDAs. In the AMC application, a pregnant woman enters her blood pressure (BP) value using a PDA or desktop regularly at home or at work, for instance, to allow her health care providers to monitor her condition. To monitor the BP without any manual data entry, we extended this application by adding a sensory part. The sensory part consists of self-organized sensors, sink and a Local Processing Unit (LPU). The sensors are capable of collecting the non-intrusive and intrusive health data of patients while the sink (aggregation point) receives all the sensed data from the sensors in a High-level Data Link Control



**Figure 1.** ACLF with emphasized sensory part.

(HDLC)-like format using the serial port. The sink can read the data from the serial port and then forward it to the IP-based network that, in turn, can be received by the LPU. In the ACLF, the PDA also assumes the role of the LPU. The sensible and meaningful data (BP in our case) are collected and filtered at the PDA. The association between the PDA and the sink is established using the registration protocol described in [12]. The PDA pulls all the data from the sink by sending request and then stores data in the server database, thereby creating an electronic record via the combinations of 802.15.4 (ZigBee), 802.11 (WiFi), and 3G/GPRS connections for sensor-to-sink, sink-to-PDA and PDA-to-Web application server, respectively. Furthermore, depending on the gravity of updated data, the database would notify the AMC application that would then automatically send an alert message with specific information on the patient to the care staff, either as an SMS or an email.

The ACLF has been tested and implemented successfully in [1]. Our AMC application houses an EMR system and a database that contains active data. The term active data refers to the data being monitored regularly for clinical analysis and changes in such data may trigger alert actions. The EMR system is an exact reproduction of the

existing paper-based records maintained in the NSW health services. It deploys Java Servlets at the application logic, while Java Server Pages (JSP) is used for the application inputs together with Java Beans. The developed application is intended for use with PDAs or smartphones. Accessing web pages using these devices has limitations, because of the device's small screen size. The web pages are designed based on our research evaluations and results in [13]. Apache Tomcat acts as a web application container, which houses the pilot application for execution. This pilot application can be used in a real-time-like environment by connecting the web application container to a powerful Internet Information Services (IIS) 6.0 web server using a public IP-address; the application itself can then be reached by using the internet. The data received from either the sensor or user entry will be stored in the MySQL database.

To indicate the accountability of any framework, it is also imperative to show the framework's reliability. The reliability of the ACLF ( $R_{ACLF}$ ) can be defined as a function of reliability of two organized parts, (i) the reliability of sensory part ( $R_S$ ) and, (ii) the reliability of the AMC application ( $R_{AMC}$ ). Because the ACLF connects these two parts, either by using 3G/GPRS or TCP/IP, we formulated the reliability of the ACLF as follows:

$$R_{ACLF} = \min (R_S, R_{AMC}) \quad (1)$$

The AMC application is a web-based application connected to the sensory part using a wireless/wired IP that deploys the HTTP/TCP over the 3G/GPRS wireless mobile network, as shown in Figure 1. The reliability measure of the wireless IP of the application is outside the scope of this paper. This paper focuses on the reliability measure for the sensory part, especially for a BAWSN application that has not been covered in the literature. The aim is to formulate a realistic reliability measure enabling the quantification of the reliability of any ACLF-like application. The following section details the prime parameters that are considered implicit in measuring the reliability.

### 3. SELECTED PARAMETERS

The reliability of an application is generally defined as a function of a number of parameters that strongly influence the dependability measure of the application. Literature [14–17] shows that the reliability of a wireless sensor network (WSN) is closely related to its routing algorithm. Consequently, the parameters are also related to the routing algorithm. For example, in free space, it is shown that the power a receiver receives is proportional to the wavelength of the signal ( $\lambda$ ) and the height of antennas ( $H$ ) [17]. The distance ( $D$ ) between the transmitter and receiver also influences the reliability of the WSNs [15]. The environment coefficient ( $C_E$ ) parameter was chosen to resolve the complex influence of environmental factors in the communications quality for the WSNs [16, 17]. The energy coefficient ( $E_F$ ) parameter was considered paramount in sensor networks because wasted energy shortens life and reliability [14]. Although these existing parameters influence the reliability measure in the WSN, they are device-centric and are at the levels where application developers have very little control over them. In this paper, we identify and selectively define several reliability parameters at a level visible to the application developers and measurable.

In fact, these parameters summarize and manifest the reliability concerns mentioned previously. We have identified parameters based on the following three considerations: (i) the user should have some control over these parameters, (ii) these parameters should cover most of the pre-existing parameters and, (iii) the measure of these parameters should be time dependent (an essential criterion for time-critical applications). Based on these considerations, we selected the parameters, critical lack time, error time and buffer time. The critical lack time,  $T_{LACK}$ , is defined as the elapsed time for which the application is inoperative owing to the delayed packet arrival. Error time,  $T_E$ , is defined as the inoperative time of the application caused by the loss of the packets in the sink buffer. Buffer time,  $T_B$ , is defined as the inoperative time of the application due to loss of packet in sink buffer. The derivation of these parameters with the rationalization for choosing these parameters together with the formulation of the reliability measure is presented in the following paragraphs.

In continuous monitoring health care applications, the arrival data rate at the end-point should be within the required critical time rate. This arrival rate depends on the performance of the individual sensor nodes, with the variance in any one sensor node influencing the total arrival data rate. To achieve this critical time rate, all sensor nodes are assumed to have the same data flow rate.

### 3.1. Derivation of Parameters

Let  $t_A$  be the regular time interval for sending the event data of a sensor node ( $A$ ). A sensor node in ACLF, for example, sends an event data every 10 seconds. If this sensor node generates  $n$  packets within the operative time<sup>2</sup> of application from 0 to  $t$ , then the average time interval  $T_{AVG}^A$  for  $n$  packets sent from sensor  $A$  is given by:

$$T_{AVG}^A = \sum_{i=1}^n \frac{T_i^A - T_{i-1}^A}{n-1} \quad \text{for } n > 1 \quad (2)$$

where  $T_1^A, T_2^A, T_3^A, \dots, T_i^A$  are the timestamps when the packets were received within the operative time  $t$  by the end-point from sensor node  $A$ .

If  $T_{AVG}^A$  is greater than the anticipated critical time interval, the critical lack time  $T_{LACK}^A$  is defined as the elapsed time for which the application is inoperative due to delayed packet arrival of sensor node  $A$ , and is given by:

$$T_{LACK}^A = T_{AVG}^A - t_A \quad \left. \begin{array}{l} 0 \\ \\ 0 \end{array} \right\} \begin{array}{l} t_A > T_{AVG}^A \\ t_A < T_{AVG}^A \\ t_A = T_{AVG}^A \end{array} \quad (3)$$

Consider a sequence of transmission instants for sensor  $A$ ,  $T_1^A, T_2^A, T_3^A, T_4^A, T_5^A, T_6^A$ , and assume that the following packets are received in the order  $\rho_1^A, \rho_2^A, \rho_{ERR}^A, \rho_4^A, \rho_6^A$ , with

<sup>2</sup>The operative time is defined as the time in which the continuous monitoring healthcare application is turned "on".

$\rho_1^A, \rho_2^A, \rho_4^A, \rho_6^A$ , being error-free packets received at  $T_1^A, T_2^A, T_4^A, T_6^A$ , respectively, and the packet  $P_{ERR}^A$  denoting the failure packet at the time instant  $T_3^A$ . It is further assumed that during the expected time instant  $T_5^A$ , no packets are received, because they are lost in the sink buffer. For the time difference between  $T_4^A$  and  $T_2^A$ , a BAWSN application is deemed unreliable, because the application is not capable of generating reliable data. The summation of such time differences between the error free packets within the operative time  $t$ , in case of error packets, is defined as the error time ( $T_E^A$ ) for sensor  $A$ . Likewise, the summation of the time differences such as  $T_6^A - T_4^A$  within the operative time  $t$  is defined as the buffer time ( $T_B^A$ ) for sensor  $A$ . It can be observed that for both  $T_E^A$  and  $T_B^A$  the application is inoperative. By amalgamating these two parameters, one can compute the total time for which the application was inoperative owing to errors or lost packets, and we represent it by  $T_{EB}^A$ . The afore-mentioned parameters are used for deriving the total unreliable time for a given sensor in a BAWSN. Nevertheless,  $T_{AVG}^A$  is computed irrespective of the  $T_E^A$  and  $T_B^A$  timestamps. The possibilities of overlaps in the timestamp between these parameters are considered and represented using the intersection in the total unreliable time  $T_{UR}^A$  for sensor  $A$  as shown below:

$$T_{UR}^A = T_{LACK}^A + T_{EB}^A - ((n-1)T_{AVG}^A \cap T_{EB}^A) \quad (4)$$

In the sensory part of our ACLF application, the intrinsic data received at the end-point is based on the aggregation of data provided by all source sensor nodes. The elapsed and error time of any one sensor will subsequently count towards the unreliable time of the entire application, because the application is considered inoperative until the end-point receives all data from every sensor node. The total unreliable time of the sensors is given by:

$$T_{UR}^S = \max(T_{UR}^A, T_{UR}^B, T_{UR}^C, \dots) \quad (5)$$

From the parameters derived above, the reliability of the sensory part in terms of *time* is given by:

$$R_s = 1 - \frac{T_{UR}^S}{t} \quad (6)$$

The reliability of the real-time applications is mainly affected by factors, such as the delay in the packet arrival, the error in the arriving packet and the loss of packets. Considering that the reliability is affected by many parameters at different levels, we choose to resolve reliability by three parameters at the user-visible and controllable levels and encompassing the existing parameters as in [14–17]. In addition, these existing parameters are very much device-centric and are not feasible for measuring by an application developer. Taking these difficulties into account, we defined  $T_{LACK}$ ,  $T_E$  and  $T_B$ , and derived their formulations. The reliability of a BAWSN is defined based on these parameters. Because the parameters are empirically calculated, the application

developer can easily compute the reliability of the ACLF-like BAWSN applications. The following section describes a series of experiments conducted and the reliability evaluated using the proposed parameters.

#### 4. EXPERIMENT

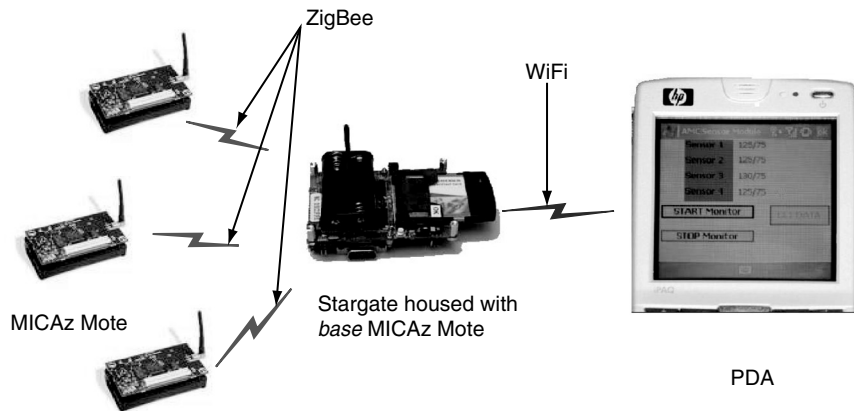
In this section, the computed reliability is based on the empirical formulation given in Section 3. The equivalent mathematical model in terms of probabilities for the BAWSN reliability is reported.

##### 4.1. Experimental Facilities

The experiment was conducted in the segregated sensory part, as shown in Figure 1. Our sensory part includes heterogeneous components differing not only in processing power and storage capacity, but also having a diverged operating platform. The sensor components consist of four crossbow-technology-based mote sensors [1,18] that operate using TinyOS [19]. The end-point in our scenario is a local processing unit, usually a PDA connected with a Stargate gateway [1, 18], which is used as the sink/base station and interconnects the 802.15.4/ZigBee-based crossbow family motes and 802.11b<sup>3</sup>-based PDAs. The following sub-sections present in-depth details of the software, hardware, and communication infrastructure used in the sensory part of our ACLF framework.

##### 4.1.1. Sensor

A crossbow-technology-based mote consists of a MICAz processor and a radio platform together with a sensor board (see Figure 2). This sensor board is a single



**Figure 2.** Hardware components in the sensory part.

<sup>3</sup>For simplicity, 802.11-based a/b/g technologies are generally referred to as 802.11



hardware and crossbow produces different types of sensor boards that are capable of sensing light, temperature, heart rate, arterial pressure, etc. The MICAz has an 8 MHz Atmel (ATMega128L) 8-bit microprocessor connected to a 2.4 GHz Chipcon CC2420 (IEEE 802.15.4/ZigBee compliant) radio transceiver which has a transfer rate of up to 250 Kbps. ZigBee or 802.15.4 compliant devices consume the least amount of power relative to other available wireless technologies, such as Bluetooth, 802.11 and 3G/GPRS. Sensor motes operate using a TinyOS [19], which is a simple but highly concurrent open source operating system that has been implemented using nesC [21], and a C-based programming language for networked embedded systems, such as sensor networks. The packet size for all the sensor motes was configured to 24 bytes including the header. The sensor motes have a hardware on/off switch, and have been programmed to transfer data every 10 seconds when the switch is turned on in ACLF.

#### 4.1.2. The Stargate gateway and PDA

The ZigBee or 802.15.4 communication standard used by the mote sensor is not widely used for a personal area network (PAN). Most of the available PDAs or smartphones come only with Bluetooth as the communication standard for PAN, and 802.11 for IP-based communication. To overcome the incompatibility of communication standards between PDAs and mote sensors, a Stargate is used as the gateway to interconnect 802.15.4/ZigBee-based Crossbow family motes and 802.11-based PDAs. Note that a Stargate can be used as a base station/sink only after housing a *base* mote sensor in the Stargate board, as shown in Figure 2. Stargate is a powerful single board computer with enhanced communications and sensor signal processing. It uses an Intel 400 MHz XScale processor (PXA255) that directly supports applications around the TinyOS based on the WSNs. The Stargate development kit is shipped with a preinstalled embedded Linux operating system kernel. This device has a larger memory, better processor capabilities and connectivity (WiFi, USB, Ethernet, serial connector) than the sensors and other gateways [18].

Stargate is programmed using a C-based application, *SerialForwarder*, provided by TinyOS [19]. The *SerialForwarder* is used to read/send packets from/to the serial port used by Stargate to contact the sensor mote. Stargate is reachable from a PDA using 802.11 in the *ad hoc* mode. However, the PDA must start a protocol to register the *SerialForwarder* application installed in Stargate. To register a PDA with Stargate, we have developed a protocol [12] and implemented it in a PDA using a .NET Compact Framework 2.0 web service application.

## 4.2. Experimental Results

The experiment was conducted in the sensory part components as shown in Figure 2. When the user initiates a continuous monitoring application [1] using a PDA, the sink receives sensed data from all the mote sensor nodes. Once the PDA has been registered, the gathered data at the sink can be fetched through the Stargate gateway. Because the *base* sensor mote attached to the Stargate has a limited buffer memory to store the sensed data received from all the sensor nodes, it is essential that data are fetched from the sink before the buffer overflows, thus causing loss of data. Hence, we have modeled

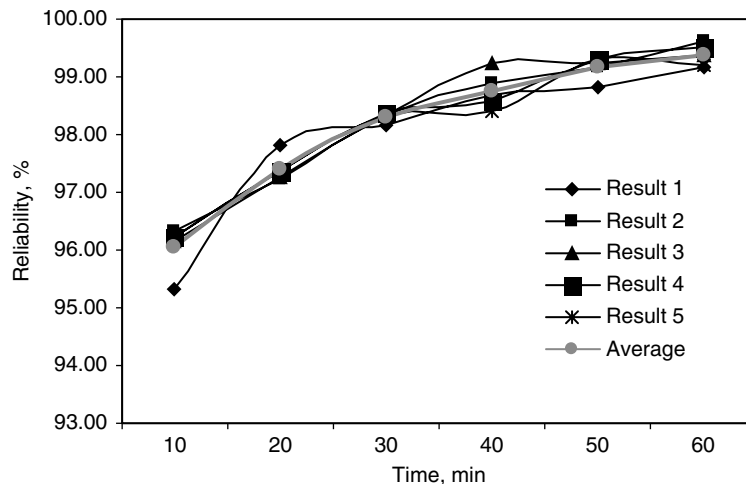
the PDA to request Stargate at a specific frequency,  $f_R$ , the number of requests sent from the PDA over a selected time interval which should satisfy the following inequality:

$$f_R < \frac{N \cdot d}{S - C} \quad (7)$$

where  $S$  is the size of the maximum sink buffer,  $C$  is the current level of sink buffer,  $N$  is the number of sensors used in a BAWSN, and  $d$  is the data rate of the sensors.

The above inequality means that data must be requested by the end-point before the buffer of the aggregation point is full; otherwise, it is lost. This inequality is programmed into the PDA to mitigate the loss of packets in the sink buffer. After the initial request, the PDA computes the request time interval  $t_R$  ( $1/f_R$ ) on the fly at which the next request is sent, based on the current buffer level  $C$ , whereas other factors  $S$ ,  $N$  and  $d$  are kept constant for a given BAWSN. Note that the PDA's request time interval can be achieved merely by using lesser value of  $t_R$  compared to the programmed time in the sensor — less than 10 seconds in our scenario. This proved to be highly inefficient in terms of sending a request to the Stargate/sink even if the buffer is empty. Consequently, these inefficient requests considerably reduce the PDA's battery power. The inequality in eqn. (7) facilitates the end-point sending a request to the sink just before the buffer is full and empties the buffer; the timely request from the PDA gradually reduces the sink buffer overflow. The impact of the inequality was analyzed by conducting a series of experiments under similar conditions with a duration of 60 minutes in our ACLF.

Our experiments demonstrated that the loss of packets in the sink buffer is reduced gradually, thereby decreasing  $T_B$ . A significant reduction in  $T_{LACK}$  is also noted owing to the timely retrieval of the packets from the sink. Reliability is calculated using eqn. (6) in Section 3. Figure 3 shows the operational reliability of a BAWSN vs. time. It is noted



**Figure 3.** Operational reliability of a BAWSN.

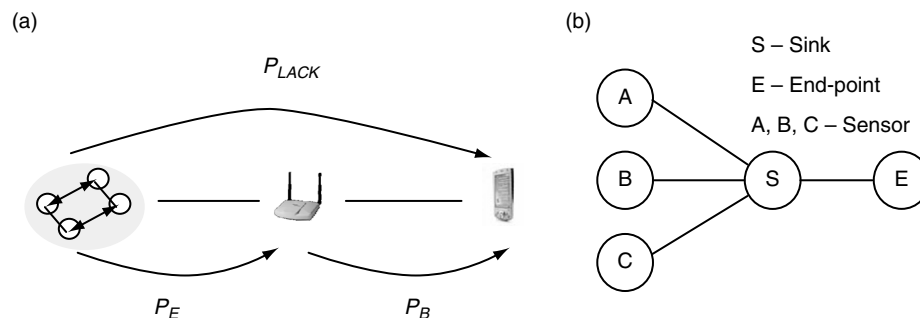
that owing to decreased  $T_B$  and  $T_{LACK}$ , the reliability of a BAWSN improves during the initial time period and stabilizes with time because of negligible  $T_B$  and  $T_{LACK}$ . Even though negligible  $T_B$  and  $T_{LACK}$  are achieved, the reliability remains basically unchanged because a consequential  $T_E$  is noted from time to time throughout the experiment. In Figure 3, the operational reliability shows very little increase in the later half of the experiment.

## 5. RELIABILITY MODEL – SIMULATION RESULTS AND DISCUSSION

A mathematical model for a BAWSN was developed to show the relationship among the critical lack time, buffer time and error time in calculating the reliability of a BAWSN. This model also demonstrates how reliability is affected by the sink buffer, a parameter the application developer has total control over. To illustrate the coherent effect of the proposed parameters, a BAWSN is modeled by a probabilistic graph, as in Figure 4(b) with the equivalent network diagram in Figure 4(a). The sensors, the sink and the end-point in a BAWSN are represented by nodes in the graph while the edges in the graph represent the links between them.

In a BAWSN, it is common that all sensors have the same coverage area and can communicate directly to the sink node, as shown in Figure 4(b). However, without losing its generality, our reliability model provides a general framework in which the reliability of any specific BAWSN architectures can be mapped and formulated. The critical lack time occurs when the LPU receives data at the elapsed time interval from all the sensor nodes. Even when all sensor nodes are assumed to have the same data flow rate, a delay in the total arrival data at the LPU is expected owing to various factors [14–17] as discussed in Section 3. The probability that a BAWSN is considered inoperative is derived below. Considering an ACLF-like scenario, where  $n$  sensors generate data at the same rate (every 10 seconds) because all the sensors can communicate directly to the sink node, the operation probability  $p$  for  $n$  sensors is the same. The inoperative probability  $P_{LACK}$  of a BAWSN can be calculated by:

$$P_{LACK} = 1 - \sum_{k=1}^{k=n} \binom{n}{k} p^k (1-p)^{n-k} \quad (8)$$



**Figure 4.** (a) A Body Area Wireless Sensor Network (b) The corresponding graph model.

where the inoperative probability  $P_{LACK}$  denotes the probability of delayed packet arrival from all sensors nodes to sink.

In a real-time BAWSN, it is not feasible to show the exact cause or network link for error and lost packets. Henceforth, the probability of error packet  $P_E$  in ZigBee (sensors to sink) under the influence of WiFi (sink to end-point) is considered as proposed in [20]. In our traffic model (Figure 4(b)), it is possible that the lost packet encountered in the end-point may occur either in the sensor-to-sink link or in the sink-to-end-point link. Moreover, an error packet can get lost either in the sensor-to-sink link or in the sink-to-end-point link. Hence, when a packet is lost at the end-point, it is assumed to be lost in the sink buffer, which rationalizes all the possibilities and is denoted by the parameter  $P_B$ . To substantiate the claim,  $T_E$  and  $T_B$  in Section 3 are calculated at the LPU rather than at the sink. Although the probabilistic factors  $P_E$  and  $P_B$  are independent of each other, their combined effect influences the BAWSN application's reliability. The inoperative probability  $P_{ERR}$ , owing to error or packet loss in terms of  $P_E$  and  $P_B$ , is given by:

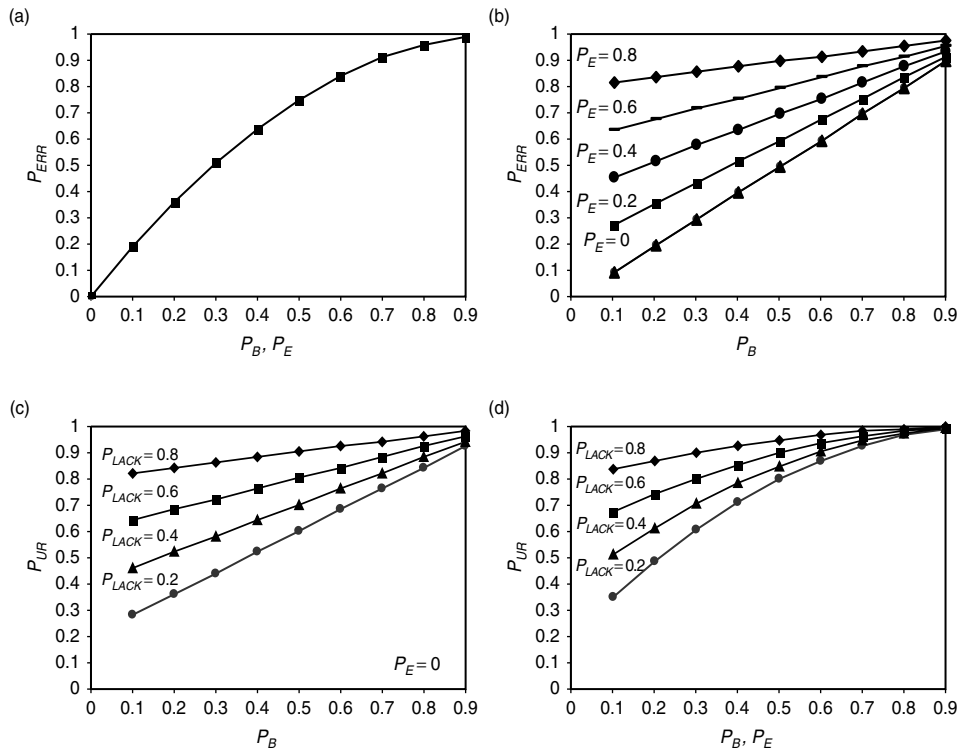
$$P_{ERR} = P_E + P_B \cdot (1 - P_E) \quad (9)$$

Similarly, although  $P_{ERR}$  and  $P_{LACK}$  are independent of each other, their combined effect will influence the BAWSN application's reliability. The *unreliable probability*  $P_{UR}$  of a BAWSN is formulated using eqns. (8) and (9):

$$P_{UR} = P_{LACK} + P_{ERR} \cdot (1 - P_{LACK}) \quad (10)$$

Simulation was performed, based on the above equations, and the results are presented in Figure 5. These results clearly demonstrate the effects of the three selected parameters on the reliability of a BAWSN. The overall  $P_{ERR}$  varies with probabilities  $P_E$  and  $P_B$  equally, as shown in Figure 5(a). Reducing either  $P_E$  or  $P_B$  will decrease  $P_{ERR}$  considerably. Our experimental analysis in Section 4 shows that error time caused by an error packet is meagre compared to the buffer time of a lost packet in the sink buffer. In addition, as mentioned in Section 4, error packets occur periodically throughout the experiment. In fact, error packet can occur in any given wireless sensor network depending on various factors as given in [20] over which an application developer has little control. Consequently, there is a decrease in the reliability when the error packet is compromised. However, the buffer overflow can be controlled by an application developer and can be purged using the proposed inequality, and then the overall  $P_{ERR}$  depends on  $P_E$  (Figure 5(b)).

The inoperable probability  $P_{LACK}$  is one of the major factors directly influencing the reliability of a BAWSN. It is noted from Figure 5(c) that even for a negligible  $P_E$  and a minimal value for  $P_B$ ,  $P_{UR}$  almost equals  $P_{LACK}$ . Both Figures 5(c) and 5(d) show that for a constant value of  $P_{LACK}$ ,  $P_{UR}$  increases with  $P_E$  and  $P_B$ . Therefore, by controlling the parameter  $P_{LACK}$ , one can achieve a higher reliability for a BAWSN. In the ACLF, an acceptable reliability is achieved with the use of a proposed inequality; it not only



**Figure 5.** Effect of probability of packet loss in sink buffer on reliability of a BAWSN: (a)  $P_{ERR}$  vs.  $P_E$  or  $P_B$ . (b)  $P_{ERR}$  vs.  $P_B$  for different values of  $P_E$ . (c)  $P_{UR}$  vs.  $P_B$  for different values of  $P_{LACK}$  and negligible  $P_E$ . (d)  $P_{UR}$  vs.  $P_B$  or  $P_E$  for different values of  $P_{LACK}$ .

purges buffer overflow, but also minimizes  $P_{LACK}$  considerably through the prompt retrieval of the packets from the sink buffer.

## 6. CONCLUSION

A realistic reliability measure model for a BAWSN has been proposed in this paper. The proposed model is constructed using three vital parameters: the critical lack time, error time, and buffer time. These vital parameters were selected such that they encompass the existing parameters that affect the reliability of a BAWSN; also, they can be controlled by an application developer. We have validated our model through experiments using our ACLF, a real-time-like BAWSN application used as an assistive health care monitoring system. Our realistic reliability measure model is justified using our probabilistic mathematical model, which also shows the cohesive relationship among the vital parameters. The study of the simulated results from the mathematical model

together with the experimental results justified the choice of the selected reliability parameters for a BAWSN. Apart from the measure model, a method enabling control of these vital parameters and enhancement of the BAWSN reliability has been proposed. This work is a part of our on-going research. The focus of our future work is on the dependability of the ACLF-like applications.

## NOMENCLATURE

### Probabilistic parameters

$n$	Number of sensor used in a BAWSN
$p$	Operational probability of sensors
$P_B$	Probability of lost packet owing to buffer overflow in sink
$P_E$	Probability of error packet in ZigBee (sensors to sink) under the influence of WiFi (sink to end-point)
$P_{ERR}$	Inoperative probability of a BAWSN owing to error or lost packet
$P_{LACK}$	Inoperative probability of a BAWSN owing to delayed packet arrival
$P_{UR}$	Unreliable probability of a BAWSN

### Reliability parameters

$R_{ACLF}$	Reliability of Active Care Loop Framework (ACLF)
$R_{AMC}$	Reliability of Assistive Maternity Care (AMC) application
$R_S$	Reliability of Sensory Part

### Request Time parameters

$C$	Current level of sink buffer
$d$	Data rate of the sensors
$f_R$	Request frequency, the number of requests sent from the PDA over a selected time interval
$N$	Number of sensors used in a BAWSN
$S$	Maximum sink buffer size
$t_R$	Request time, the time interval at which the next request is sent from the PDA

### Time-specific parameters

$t$	Operative time of a BAWSN application
$t_A$	Regular time interval for sensor $A$ to send packets
$T_{AVG}^A$	Average time interval for sensor $A$ within the operative time
$T_B$	Buffer time, inoperative time of application owing to loss of packet in sink buffer
$T_E$	Error time, inoperative time of application owing to error packet
$T_{EB}^A$	Total inoperative time for sensor $A$ owing to $T_E^A$ and $T_B^A$
$T_i^A$	Timestamp of the $i^{\text{th}}$ packet received from sensor $A$ within the operative time
$T_{LACK}$	Critical lack time, elapsed time for which the application is inoperative due to delayed packet arrival

$T_{UR}^A$	Total unreliable time for sensor $A$ within the operative time $t$
$T_{UR}^S$	Total inoperative or unreliable time of sensory part

**Greek**

$\rho_i^A$	$i^{\text{th}}$ packet received from sensor $A$
------------	---

**REFERENCES**

- [1] Venki Balasubramanian and Hoang, D.B., SOAP-based Assistive Care Loop using Wireless Sensor Networks, *Proceedings of the 1<sup>st</sup> IEEE International Symposium on IT in Medicine and Education*, Xiamen, China, 2008, 409–414.
- [2] Hoang, D.B., Homer, C., Lawrence, E., Foureur, M., Ahmad, N.F, Venki Balasubramanian and Leap, N., Assistive Care Loop with Electronic Maternity Records, *Proceedings of the 10<sup>th</sup> IEEE International Conference on e-Health Networking, Applications and Services*, Singapore, 2008, 118–123.
- [3] Jovanov E., Raskovic D., Price J., Chapman J., Moore A., and Krishnamurthy A., Patient Monitoring Using Personal Area Networks Of Wireless Intelligent Sensors, *Biomedical Sciences Instrumentation*, 2001, 37, 373–378.
- [4] Wilson L.S., Gill R., Sharp I.F., Joseph J., Heitmann S. A., Chen C.F., Dadd M.J., Kajan A., Collings A.F., and Gunaratnam M., Building The Hospital Without Walls – A CSIRO Home Telecare Initiative, *Telemedicine Journal*, 2000, 6(2), 275–281.
- [5] Alwan, M., Dalal, S., Mack, D., Turner, B., Leachtenauer, J. and Felder, R., Impact of Monitoring Technology in Assisted Living: Outcome Pilot, *IEEE Transactions on Information Technology in Biomedicine*, 2006, 10(1), 192–198.
- [6] Harvard University. CodeBlue Project: Wireless Sensor Networks for Medical Care, <http://fjji.eecs.harvard.edu/CodeBlue> [accessed January 24, 2010].
- [7] Holland, G. and Vaidya, N., Analysis of TCP Performance over Mobile Ad Hoc Networks, *Proceedings of the 5<sup>th</sup> Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Seattle, WA, 1999, 219–230.
- [8] Wang, F. and Zhang, Y., Improving TCP Performance over Mobile Ad-Hoc Networks with Out-of-Order Detection and Response, *Proceedings of the 3<sup>rd</sup> ACM International Symposium on Mobile Ad Hoc networking and computing*, Lausanne, Switzerland, 2002, 217–225.
- [9] Wan, C.Y., Campbell, A.T. and Krishnamurthy, L., PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks, *Proceedings of the 1<sup>st</sup> ACM International Workshop on Wireless Sensor Networks and Applications*, Atlanta, Georgia, 2002, 1–11.
- [10] Park, S., Vedantham, R., Sivakumar, R. and Akyildiz, I., A Scalable Approach for Reliable Downstream Data Delivery in Wireless Sensor Networks, *Proceedings of the 5<sup>th</sup> ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New York, NY, 2004, 78–89.
- [11] Stann, F. and Heideman, J., RMST: Reliable Data Transport in Sensor Networks, *Proceedings of the 1<sup>st</sup> IEEE International Workshop on Sensor Network Protocols and Applications*, Anchorage, AK, 2003, 102–113.
- [12] Barbar'an, J., D'laz, M., Esteve, I. and Rubio, B., RadMote: A Mobile Framework for Radiation Monitoring in Nuclear Power Plants, *International Journal of Electronics, Circuits and Systems*, 2007, 1(2), 160–165.
- [13] Venki Balasubramanian and San Murugesan, Adaptation of Web Pages for Hand-held Devices, *Proceedings of the 5<sup>th</sup> International Conference on Web Engineering*, LNCS Springer, Sydney, Australia, 2005, 435–440.
- [14] Sankarasubramaniam, Y., Akan, O. and Akyildiz, I., ESRT: Event-to-Sink Reliable Transport in Wireless Sensor Networks, *Proceedings of the 4<sup>th</sup> ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Annapolis, Maryland, 2003, 177–188.

- [15] Brown, S. and Sreenan, C.J., A Study on Data Aggregation and Reliability in Managing Wireless Sensor Networks, *Proceedings of the 4<sup>th</sup> IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, Pisa, Italy, 2007, 1–7.
- [16] Pazynyuk, T., Jian Zhong, Li., Oreku, G.S. and Pan, L., Qos as Means of Providing WSNs Security, *Proceedings of the 7<sup>th</sup> International Conference on Networking*, Cancun, Mexico, 2008, 66–71.
- [17] Zhengming, L., Baosen, L. and Tianhong, P., The Research on Reliability of Wireless Sensors Network, *Proceedings of the 3<sup>rd</sup> IEEE International Conference on Networking, Sensing and Control*, Hainan, China, 2008, 1449–1452.
- [18] Crossbow Technology Inc., Motes, Wearable Sensors, <http://www.xbow.com/> [accessed January 24, 2010].
- [19] TinyOS Embedded Operating System, <http://www.tinyos.net> [accessed January 24, 2010].
- [20] Shin, S., Choi, S., Park, H.S. and Kwon, W.H., Packet error rate analysis of IEEE 802.15.4 under IEEE 802.11b Interference, *Proceedings of the 3<sup>rd</sup> International Conference on Wired/Wireless Internet Communications*, LNCS Springer, Xanthi, Greece, 2005, 279–288.
- [21] Gay, D., Levis, P., von Behren, R., Welsh, M., Brewer, E. and Culler, D., The nesC Language: A Holistic Approach to Networked Embedded Systems, *Proceedings of the ACM SIGPLAN conference on Programming Language Design And Implementation*, San Diego, California, 2003, 1–11.





**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

