A Storage-free Data Parasitizing Scheme for Wireless Body Area Networks

Yuan-Yao Shih[1], Ai-Chun Pang[1,2,3], Pi-Cheng Hsiu[3]
[1]Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.
[2]Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan, R.O.C.
[3]Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan, R.O.C.
E-mail: d00922018@csie.ntu.edu.tw, acpang@csie.ntu.edu.tw, pchsiu@citi.sinica.edu.tw

***Abstract*—With the increasing sophistication and maturity of biomedical sensors and the significant advances on low-power circuits and wireless communications technologies, wireless body area networks (WBANs) have emerged recently to provide pervasive health monitoring for humans. In WBANs, smart phones can serve as data sinks to forward the sensing data to back-end servers. Due to the battery concern of smart phones and the postural changes of humans, temporary disconnection between sensors and their associated smart phones may frequently happen in WBANs. In this case, the sensing data would be lost when the limited memory space of sensors overflows. To prevent excessive data loss, this paper proposes a scheme to parasitize the data on existing public Wi-Fi networks, once the links from sensors to the smart phones become unavailable. Specifically, an optimization problem to maximize the time during which data loss can be avoided by exploiting the data parasitizing scheme is formulated, where a decision set of the packets' size and sending timing to public Wi-Fi networks needs to be determined. We develop an offline algorithm to obtain an optimal decision set and present an efficient online algorithm for practical implementations. The feasibility of the proposed scheme and the efficacy of the algorithms are demonstrated through prototype implementations on a WBAN testbed with biomedical sensor devices for real-world experiments.**

***Index Terms*—Wireless body area networks, data loss prevention, data parasitizing**

I. Introduction

Traditional biomedical sensors, such as clinical thermometers and sphygmomanometers, are used popularly but solely for personal health care. Recently, chronic diseases have become one of the leading causes of death and disability in many countries. One of the best ways to save the lives of these patients relies on continuous health monitoring [1]. To effectively collect measurements and send the measured data to a monitoring system, a communication network amongst biomedical sensors is needed. As a result, *wireless body area networks* (WBANs), a new generation of wireless sensor networks (WSNs), have emerged [2]. In a WBAN, biomedical sensors, deployed on the patients' body, report their measurements to a data sink by wireless transmission in a single-hop or multi-hop fashion. The sink often links to a healthcare institution through wide area networks. Thanks to the WBAN, medical specialists in the institution can collect real-time health-related information, so appropriate and timely medical advice or treatment can be delivered to the patients.

The application scenarios of WBANs are initially given in ambulances and hospitals. The patients are often in high risk of death and emergency. The mobility of the patients is low since they cannot move freely by themselves. For this kind of scenario, the WBAN needs to deliver data in time and provide excellent user experience to medical specialists, who may not be familiar with electronic devices [3], [4]. As people are paying more attention to their health, WBANs start to extend their applications to home or public space. The patients in such scenario are usually not in emergency but in some complex circumstances WBANs have to deal with. For example, unlike traditional WSNs, the postural movement of patients has strong impact on the performance of WBANs [5], [6]. Furthermore, people are used to carrying their smart phones anywhere and anytime. Smart phones are equipped with high processing power and multiple wireless radio interfaces. The flexible software development kits and rich user interfaces supported by smart phones make it easy for patients to configure and control their WBANs. With so many great properties, the smart phone is an excellent choice to serve as the data sink of WBANs [7], [8]. In this paper, we focus on the scenario in home or public space where patients deployed with biomedical sensors and carrying smart phones can freely move and change their postures.

For the WBANs used in home or public space, temporary disconnection between any two sensors or between a sensor and the data sink may happen mainly in two kinds of situations. The first one is the postural changes caused by human movement [9], which leads to varying link quality. The other cause is on the data sink, i.e., the smart phone. A smart phone needs to perform many different tasks, such as collecting data from the sensors, making phone calls, or surfing the Internet, which would lead to undesirably high energy consumption [10]. Thus, a smart phone may run out of battery and the battery needs to be recharged or changed. As a result, temporary disconnection between sensors and the data sink occurs. For the focused scenario, since the patients are not in emergency, the measurement data collected by sensors can be delivered later after the connection is recovered, which means that delivering complete and intact data is more important than real-time data delivery [2], [11].

A straightforward solution to prevent data loss is to temporarily store the data on the sensor storage when disconnection happens. However, the sensors in WBANs are small [11], and the external storage is an extravagance on this kind of devices. Instead, the measurement data is usually stored in the embedded memory in the micro-controller of sensors. The size of the embedded memory is relatively small and

ISBN 978-3-901882-58-6 © 2014 IFIP

can just keep the data from the typical measurements, e.g., electrocardiography (ECG), for less than 10 seconds [12], [13]. Since the focused scenario is in home or public space, it is expected that there are some existing wireless accesses such as Wi-Fi hotspots in the surroundings. This observation motivates us to prevent data loss due to temporary disconnection in WBANs by "parasitizing" the measurement data on those Wi-Fi networks without the need for sensors to have external storage.

In this paper, we want to ensure the intactness of biomedical sensing data for WBANs once temporary disconnection happens. It is no doubt that Wi-Fi networks are ubiquitous nowadays. In addition, PING, a popular network administration utility to test the reachability of a host, operates by sending Internet Control Message Protocol (ICMP) echo request packets to a target host. Upon receiving an echo request packet, the target host will send an echo reply packet back, and the reply packet will contain the data received in the request packet [14]. The above observations serve as the foundation of our proposed scheme that tries to parasitize the measurement data on surrounding Wi-Fi networks. The proposed data parasitizing scheme makes use of ICMP echo request/reply packets to carry the measurement data in Wi-Fi networks by embedding the data into multiple echo request packets and sending those packets to the Internet hosts, such as gateways and routers. With the time difference (i.e., delay) between a pair of echo request and reply packets due to network propagation and host processing, the Wi-Fi netowrk can be considered as temporary storage for sensors. By carefully making a decision on when to send the request packets and how much data to be embedded in the echo request packets, the measurement data generated during temporary disconnection can be preserved.

This paper makes three contributions: firstly, we propose the idea of "data parasitizing" and derive the key requirements to make use of data parasitizing to ensure the intactness of measurement data when temporary disconnection happens in a WBSN. Secondly, an optimal offline solution to fulfill the requirements has been proposed to serve as a performance baseline, and a practical online algorithm inspired by the properties of the offline solution is developed. Finally, we implement our proposed algorithm on WBAN devices and validate the performance of our data parasitizing scheme through extensive experiments in real-world environments.

The rest of this paper is organized as follows. Section II reviews some related works. In Section III, we describe the system model and formulate the problem. The optimal offline algorithm and the efficient online algorithm are introduced in Section IV. Section V reports some experimental results, and Section VI concludes this paper.

II. Related Works

WBANs, the key role in future e-health, have attracted significant interest for a wide range of research topics recently [2], [11]. Unlike traditional WSNs, WBANs are deployed around human bodies; thus, some researchers focused on understanding the impact of human body tissues on the propagation of radio waves [15], [16]. Based on those WBAN radio wave studies, researchers proposed specific *medium access control* (MAC) protocols to provide reliable communications in WBANs [17], [18]. In addition, since WBANs are adopted to convey important and personal medical information, some researches have been carried out to offer more stringent security and privacy for WBANs [7], [19].

Apart from the obstruction of human bodies, the performance of WBANs is also significantly influenced by the postural movement of patients. The postural movement may cause temporary disconnection between nodes in the network [20]. Some researches have been done to tackle this problem from the routing perspective. Latre et al. [21] proposed a protocol that constructs routing trees in a distributed manner. Then, by carefully scheduling communication time-slots among the nodes in WBANs, the proposed protocol aims to offer good resilience to mobility. Another group of researchers borrowed the concept from delay tolerant networks, where the information about the postural movement has been used to improve the efficiency of routing when external storage is available [22], [23]. Furthermore, Liang et al. [9] proposed a routing framework for WBANs. They developed a model to predict link quality, and the prediction result is used to improve routing reliability and resist data injection attacks. This work can ensure reliable routing, but the task of ensuring that the measurement data is intact remains untouched.

Our proposed scheme exploits the properties of ICMP echo request/reply packets. In networking area, ICMP request/reply packets are mostly utilized to measure the round trip time so as to determine the bandwidth and latency of network links [24]. Instead of using such packets as probes, ICMP can be used for another purpose. The ptunnel [25] and icmptx [26] applications tunnel TCP/IP connections using ICMP echo request and reply packets. Those implementations serve as proofs-of-concept that ICMP packets can be used to encapsulate data. When temporary disconnection happens in WBSNs, we cleverly utilize ICMP echo request/reply packets to "parasitize"the measurement data in surrounding Wi-Fi networks to prevent excessive data loss. To the best of our knowledge, this is a very early attempt and has not been considered before.

III. System Model and Problem Formulation

In this section, we introduce our system model, network architecture, and assumptions underlying the system model. Then, we formally define the design objective and the problem under investigation.

A typical WBAN is comprised of a data sink and multiple biomedical sensors. The data sink performs the initialization, maintenance, and control functions in the network. The data sink also serves as a gateway connected to healthcare institutions through a wide area network (WAN). The communication of WBAN nodes can be achieved in a single-hop or multi-hop fashion. Without loss of generality, in this paper, we consider the system model based on a WBAN with multi-hop communications [27], where sensors are equipped with data forwarding capability. There exists a communication link between any two nodes if they are within each other's communication range. A routing tree rooted at the data sink is constructed. Here, any existing WBAN tree construction algorithm can be used, and any existing WBAN routing protocol can be utilized for identifying next-hop routing candidates. In other words, our proposed scheme is compatible with any kind of routing trees and protocols.

Our focused scenario for WBANs consists of a smart phone, carried by a patient serves as the data sink, and multiple biomedical sensors deployed on the patient's body. Each of those biomedical sensors can sense one or more kinds of health conditions, such as blood pressure, glucose, intraocular pressure, and electrocardiography. For ease of presentation, we assume that every biomedical sensor senses only one kind of health conditions, although our scheme remains applicable when this assumption is relaxed. Each sensor can send the measurement data to the data sink via multi-hop wireless communications. The connection among sensors and the smart phone generally use the low-power radio (IEEE 802.15.4, 802.15.6 or Bluetooth LE). The smart phone is connected to a healthcare institution through the WAN. The medical specialists in the institution can monitor, collect, and analyze the health conditions from the measurement data forwarded by the smart phone.

As mentioned previously, temporary disconnection of some communication links may happen due to the postural movement of patients and the running-out of batteries on smart phones. This may lead to communication disruption, and the measurement data from those sensors cannot be sent to the data sink during the disruption period. If a sensor's embedded memory is fully occupied by its sensing data during that period, then some of the measurement data would get dropped and be lost. In this paper, we propose a data loss prevention scheme by parasitizing the measurement data on surrounding Wi-Fi networks. We can observe that Wi-Fi networks are ubiquitous. If the Wi-Fi networks are completely-open, the device can upload the sensing data directly to the server on the Internet. However, most of the Wi-Fi networks are using web-based authentication. Even though a person does not have a valid account and password, some hosts, like web servers, domain name servers, or authentication servers owned by the service provider, in the provider's network can still be reached by ICMP echo request packets supported by the Internet protocol suite. The standard protocol of ICMP echo request/reply packets has an innate property that the reply packet must contain the same data carried in the corresponding request packet.

By exploiting this property, we can parasitize the measurement data in those Wi-Fi networks. When a sensor device notices that it is disconnected from its WBAN, the device starts to find available Wi-Fi networks in the surroundings. Since the disconnection is caused by the postural changes of patients or the battery issue of the data sink, the sensor on the body can still get a chance of having good link quality with those nearby Wi-Fi access points. Then, the device embeds the still-generated measurement data into multiple ICMP echo request packets and sends those packets to some hosts on one of the available Wi-Fi networks. After a period of time, the echo reply packets carrying the exactly same measurement data will be sent back. The device can repeat constructing and sending such ICMP echo request packets with the data carried until the connection in the WBAN is recovered. In this way, we can keep the measurement data intact during the disconnection period.

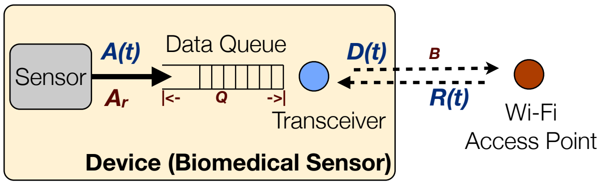


Fig. 1. The system model on a sensor device

The objective of our scheme is to keep the measurement data intact during the disconnection period. In the previous paragraph, we elaborate on the main idea of our "data parasitizing" scheme. Here, we further formulate the problem that our scheme tries to tackle. Since the scheme is applied to each device (i.e., biomedical sensor) individually, the interaction among devices is not discussed in this paper.

Fig. 1 depicts the system model of the scheme on a senor device. A sensor device consists of a sensor, a data queue, and a transceiver. The sensor is capable of sensing one specific health condition and generates raw measurement data periodically. The data arrival rate of the raw measurement data is denoted as A_r (bps). The data queue is the temporary storage of the measurement data before they are sent out. Since we assume that our sensing device does not have external storage, the length of data queue Q (bits) is limited by the memory size of the micro-controller on the device. The transceiver is capable of sending and receiving data via a wireless link. Here we assume the transceiver on each WBAN device is compatible with Wi-Fi/802.11 standards. Also, we assume that when the device is disconnected from the WBAN, the device itself will refuse to forward other devices'data. That is, the transceiver on the device will not receive data from other WBAN devices during the disconnection. The bandwidth of the Wi-Fi link to the surrounding access point (AP) is denoted as B (bps). t denotes the system time, counting from zero after the disconnection happens. The raw measurement data generated by the sensor will flow into the data queue. We denote $A(t)$ as the total amount of accumulated data generated from the sensor to the data queue at time t. $A(t)$ can be derived from $A_r \times t$. The transceiver sends the packets to a Wi-Fi access point with the data taken out from the data queue. To protect the privacy of the data, the data in those ping packets should be encrypted. We denote $D(t)$ as the total amount of accumulated data departed from the transceiver at time t. In addition, the data from echo reply packets received by the transceiver will be put into the data queue. We denote $R(t)$ as the total amount of accumulated data received by the transceiver at time t. A major constraint in the system is that the amount of the data inside the data queue cannot be larger than the length of the queue at any time. If the constraint is violated, data loss will happen due to data queue overflow. Then, the system is deemed dead. The system lifetime, denoted by T, means that the system dies at time T. In other words, the system lifetime is the interval between the time of the disconnection happening and that of the first data loss.

Recall that our objective is to keep the measurement data intact during the disconnection. If the scheme can achieve a longer system lifetime, we can more resist against temporary disconnection for WBANs. As a result, for this problem of data parasitizing, our objective can be transferred to maximize the system lifetime. To solve this problem, we need to decide when to send an ICMP echo request packet and how much data the packet should carry. For each ICMP echo request packet sent out by the transceiver, a corresponding ICMP echo reply packet will be sent back to the transceiver. For each pair of ICMP echo request/reply packets, t_n, d_n, and r_n for the nth packet pair are denoted as the request packet sending time, the amount of data the packet carries, and the time of the reply packet received by the transceiver, respectively. Then, $D(t) = \Sigma_1^n d_n$ such that $t_n \leq t$, and $R(t) = \Sigma_1^n r_n$ such that $r_n \leq t$. We want to send the data out of the data queue as quickly as possible to prevent the queue from overflow. However, in addition to the data generated by the sensor, when the reply packet is back, the data carried by that packet needs to be stuffed into the data queue as well, so sending out packets too early may not be a good choice. Thus, how to find an optimal packet sending decision is the main challenge for the problem.

To keep the system alive, the key is to prevent the data queue from overflow. During the disconnection period, the data will be fed into the data queue of a device from two sources: the measurement data generated from the sensor and the data in the echo reply packets received by the transceiver. The sensor periodically generates the measurement data, and these data will then be fed into the data queue continuously. Eventually, the queue will be full if we do not send the data out via echo request packets. Intuitively, we want to send the data out as soon as possible to keep the queue empty. However, those data sent out will be returned after some time. When those data are back, we must make sure that the queue has some space to accommodate them. This observation yields the following constraint for the data queue:

$$A(t) - D(t) + R(t) \leq Q - d_n, n = 1, ..., N-1 \mid t = r_n$$

The equation ensures that the data queue has some space to accommodate the data carried by the reply packets when the packets arrive. Specifically, the amount of data in the data queue at time t should always be less than the length of the queue minus the amount of data carried by the reply packet arrives at time t.

For optimizing the performance of our data parasitizing scheme, we formulate the Maximum system Lifetime Data Parasitizing (MLDP) Problem as follows.

Maximum system Lifetime Data Parasitizing (MLDP) Problem

Input: Sensor data arrival rate A_r (bps), length of data queue Q (bits), and Wi-Fi link bandwidth B (bps) to an AP, the maximum amount S of data an ICMP echo request packet can carry.

Output: System lifetime T, packet sending decision set $\rho = (t_1, d_1), ..., (t_N, d_N)$, $0 < t_1 < ... < t_N < T$, where N is the number of packets sent by the transceiver before time T.

Objective: Maximize system lifetime T under the constraint that $A(t) - D(t) + R(t) \leq Q - d_n, n = 1, ..., N-1 \mid t = r_n$ always holds when $t \in (0, T)$ and every $d_1, ..., d_N \leq S$.

IV. Our Data Parasitizing Scheme

This section presents our data parasitizing scheme. In Section IV-A, we discuss some important properties of packet sending decision sets. Then, based on the discussion, we propose an offline algorithm in Section IV-B to solve the MLDP problem and obtain an optimal decision set. Finally, an efficient online algorithm inspired by the optimal offline algorithm is developed in Section IV-C to provide a practical implementation in real-world scenarios.

A. Discussion on Packet Sending Decision Sets

To further analyze the impact of the packet sending decision set on the system lifetime, we define the system capacity C and depict the relationship between the system capacity and the system lifetime. The system capacity indicates how much data the system can store. Intuitively, a larger system capacity means that the system can store more data generated from sensors; thus, the system lifetime is longer, and vice versa. Consequently, we have the first property that if a decision set can achieve a larger system capacity, then a longer system lifetime can be obtained, and vice versa.

Property 1. *Two packet sending decision sets: ρ_1 and ρ_2.*

$$T_{\rho_1} > T_{\rho_2} \iff C_{\rho_1} > C_{\rho_2}$$

In the system, there are two places where the data can be stored. One is the data queue on the sensor device. The other is the Wi-Fi network via our data parasitizing scheme. Obviously, the capacity of the data queue is the length of the queue. On the other hand, we denotes C_B as the capacity of the Wi-Fi network. Then, $C = Q + C_B$. Since the length of the data queue is limited by the micro-controller's memory space, the only parameter we can adjust is the capacity of the Wi-Fi network. The size of C_B depends on how much data echo request packets can carry before they are sent back. That is where the decision set comes in. The decision on each packet's sending timing and data size will affect C_B. During the time interval from a request packet sent out to its corresponding reply packet back, the data can be: (1) on the way to the destination host (i.e., carried by the echo request packet), (2) in the host (i.e., processing), or (3) on the way back to the device (i.e., carried by the echo reply packet). A maximum C_B can be obtained if the communication links and the processing queue of the host are fully occupied by the measurement data. Ideally, the upper limit of C_B is decided by the wireless link bandwidth B. However, in reality, the communication links cannot be completed filled with the measurement data since some other information, such as control messages and packet headers, needs to be delivered in the Wi-Fi network. To increase C_B, the total size of each packet's header should be reduced. Since each packet's header size is fixed, the only thing we can do is to reduce the number of packets required to carry a unit of data. Thus, we have the second property that the fewer the packets required to carry a unit of data, the larger the system capacity.

Property 2. *Two packet sending decision sets: ρ_1 and ρ_2.*

$$\frac{N_{\rho_1}}{E_{\rho_1}} < \frac{N_{\rho_2}}{E_{\rho_2}} \implies C_{\rho_1} > C_{\rho_2}$$

where $E = \Sigma_{n=1}^{N} d_n$ is denoted as the total amount of data carried by the echo request packets sent out.

Based on Properties 1 and 2, if we can reduce the number of packets required to carry a unit of data, the system capacity can be increased and the system lifetime will be prolonged. Finally, we have a corollary that the fewer the packets required to carry a unit of data, the longer the the system lifetime.

Corollary 1. *Two packet sending decision sets: ρ_1 and ρ_2.*

$$\frac{N_{\rho_1}}{E_{\rho_1}} < \frac{N_{\rho_2}}{E_{\rho_2}} \implies T_{\rho_1} > T_{\rho_2}$$

B. An Optimal Offline Algorithm

The above corollary suggests the form of the optimal offline determination, i.e., to maximize the interval between each pair of successive echo request packets. Suppose that the arrival times of the echo reply packets are known in advance once the sending times of the corresponding echo request packets are determined. That is, r_n can be obtained once the algorithm determines (t_n, d_n). In this way, we can utilize a greedy approach to derive the optimal decision set. We then proceed to define the optimal determination.

Given the sensor data arrival rate A_r, the length of data queue Q (bits), the maximum amount of data a packet can carry S, and a function $\text{PKT\_RETURN}(t, d)$ that will return the corresponding r_n based on (t_n, d_n) for the nth packet. For the first packet, we define

$$\begin{aligned} t_1^* &= t \mid A(t) = \min\{Q, S\}. \\ d_1^* &= \min\{Q, S\}. \end{aligned} \tag{1}$$

The first echo request packet will be sent out once the data queue is about to overflow or the amount of data in the queue reaches the upper bound a packet can carry. In other words, this packet will be sent out only when necessary, and will carry as much data as possible.

Then, we derive the reply packet return time: $r_1^* = \text{PKT\_RETURN}(t_1^*, d_1^*)$. For the nth packet ($n \geq 2$), we define (assuming that the nearest coming reply packet is the mth packet)

$$\begin{aligned} t_n^* &= t \mid t > t_{n-1}^*, A(t) - D(t) + R(t) = \min\{Q - d_m^*, S\}, \\ &\quad r_m^* = \text{PKT\_RETURN}(t_m^*, d_m^*) > t \\ d_n^* &= \min\{A(t) - D(t) + R(t), S\}. \end{aligned} \tag{2}$$

That is, the packets will be sent when (1) the amount of data in the data queue is about to become larger than the length of the queue minus the amount of data carried by the coming reply packet, or (2) the amount of data in the queue reaches the upper bound a packet can carry. Moreover, this packet must carry as much data as possible. The above calculation repeats until we cannot find a feasible t_n^*. That is, $\forall\ t > t_{n-1}^*, A(t) - D(t) + R(t) > Q$. Suppose that the calculation stops at the Nth packet. Then, the decision set determined by the offline algorithm is defined as follows.

Definition 1 (OOD)**.** *The packet sending decision set ρ^* given by*

$$\rho^* = (t_1^*, d_1^*), ..., (t_N^*, d_N^*)$$

is called the Optimal Offline Determination (OOD).

Now, we proceed to prove the feasibility and the optimality of OOD.

Lemma 1. *OOD is a feasible decision set.*

Proof: In OOD, based on Eq. 1 and 2, the next echo request packet will be sent out immediately before the amount of data in the data queue becomes larger than the length of the queue minus the amount of data carried by the coming reply packet. Thus, $A(t) - D(t) + R(t) \leq Q - d_n$ holds for $n = 1, ..., N-1$. In addition, the amount of data carried by a request packet is always less than or equal to S. Thus, $d_n \leq S$ holds for $n = 1, ..., N-1$. As a result, ODD is a feasible decision set for the MLDP problem. ■

Theorem 1. *OOD is an optimal decision set.*

Proof: We prove this theorem by contradiction. Suppose there exists a feasible decision set ρ' with $T' > T^*$.

The first packet in ρ' cannot be sent later then that in ρ^*. Otherwise, the data queue will overflow. Thus, the first packet's sending time in ρ' is not later than that in ρ^*. Accordingly, the amount of data carried by the first packet in ρ' is not more than that in ρ^*. Thus, we have

$$t_1' \leq t_1^*, d_1' \leq d_1^*. \tag{3}$$

Assume that the kth packet's sending time in ρ' is not later than that in ρ^*, and the amount of data carried by the kth packet in ρ' is not more than that in ρ^*, for some unspecified number k. If the $k+1$th packet's sending time in ρ' is later than that in ρ^*, the data queue must overflow since the time interval between the kth and $k+1$th packets in ρ' is larger than that in ρ^*. Thus, the $k+1$th packet's sending time in ρ' is not later that in ρ^*. With the same argument, the amount of data carried by the $k+1$th packet in ρ' is not more that in ρ^*. Thus, we have

$$t_n' \leq t_n^*, d_n' \leq d_n^*\ \forall n > 1. \tag{4}$$

Then, we discuss three possible cases, depending on the total number of packets N in ρ' and ρ^*:

Case 1 ($N' < N^*$): If the total number of packets in ρ' is smaller than that in ρ^*, based on Eq. 4, the N'th packet's sending time in ρ' is earlier than the N^*th packet's sending time in ρ^*. Thus, the system lifetime achieved by ρ' will be shorter than that achieved by ρ^*. It can be derived that $T^* > T'$.

Case 2 ($N' = N^*$): Since the total number of packets in ρ' is equal to that in ρ^*, according to Eq. 3 and 4, the system lifetime achieved by ρ' will not be longer than that achieved by ρ^*. It can be derived that $T^* \geq T'$.

Case 3 ($N' > N^*$): If the total number of packets in ρ' is larger than that in ρ^*, the system lifetime achieved by ρ' will not be longer than that achieved by ρ^* since the number of packets required to carry a unit of data in ρ' is larger than that in ρ^*. Thus, it can be derived that $T^* \geq T'$.

Finally, we conclude that $T^* \geq T'$, which contradicts the assumption and complete the proof. ■

C. An Online Algorithm

In this section, we present an online algorithm. Unlike the optimal offline algorithm which is given the exact time at which each echo reply packet will be received, an online algorithm has no idea about the information of future reply packets. Thus, for online decision set determination, we utilize two techniques to overcome that challenge: one is a simple estimation on each echo reply packet time's receiving time. The other one is an additional swap space for the data queue.

Before proceeding to define the online determination, we elaborate on the two techniques first. The first technique is to estimate the receiving time of each reply packet. Given the sensor data arrival rate A_r, the length of the data queue Q (bits), the maximum amount of data a packet can carry S, and the wireless communication bandwidth B (bps), unlike the offline algorithm, the online algorithm will send out an echo request packet carrying an amount S of fake data before sending out the first echo request packet. This packet serves as a probe to do the echo reply packet time estimation. We denote B_{est} as the estimated bandwidth of the wireless link. In the beginning, we have $B_{est} = B$ before the corresponding echo reply packet of the probe packet is received. After the corresponding echo reply packet is received, B_{est} will be updated:

$$B_{est} = \frac{S + S_h}{r_0 - t_0}. \tag{5}$$

where r_0 is the reply packet's receiving time, t_0 is the request packet's sending time, and S_h is the size of the packet header.

Note that, B_{est}, different from traditional bandwidth estimation, cannot be considered as a real bandwidth estimated since the host's processing time is also included in the estimation. However, it just fits our need here because what we need is the time when the echo reply packet will be back, not the real-time bandwidth of the wireless communication link. The online algorithm only sends one probe packet. After that, B_{est} will keep being updated by the echo reply packets carrying the measurement data according to Eq. 5. When each echo request packet is sent out, the corresponding reply packet's receiving time is estimated by

$$r_n = \frac{d_n + S_h}{B_{est}} + t_n$$

for the nth packet.

The second technique is to increase the swap space of the data queue. The offline determination has only to ensure that the queue has enough space for the data carried by each echo reply packet immediately before receiving that packet. However, for the online determination, we are not sure when the packet will be received exactly. The simple estimation described above may give rise to some errors. The errors can lead to data queue overflow and the death of the system as a consequence, if there is no enough space for the data carried by the coming reply packet. To reduce the possibility for the occurrence of this case, an extra space is reserved from the data queue. The size of the extra space, denoted as $Q_s(t)$ at time t, equals the size of the data carried by the *two* coming reply packets estimated, instead of only one as did in the offline determination. Then we have

$$Q_s(t) = d_n + d_m \mid r_m > r_n > t, \\ \nexists r_o, r_n > r_o > t \text{ or } r_m > r_o > r_n.$$

where $Q_s(t)$ equals to the total amount of data carried by the two coming reply packets estimated at time t.

Now, we proceed to define the online determination. Following the same design concept of the offline algorithm presented in the previous section, the online algorithm is fundamentally based on Corollary 1. That is, the online algorithm attempts to have each request packet carry as much data as possible. For the first packet, we define

$$t_1 = t \mid A(t) = \min\{Q, S\}. \\ d_1 = \min\{Q, S\}.$$

Like OOD, the first echo request packet is sent out once the data queue is about to overflow or the amount of data in the queue reaches the upper bound a packet can carry. Next, for the nth packet, $n > 1$, we define

$$t_n = t \mid t > t_{n-1}, A(t) - D(t) + R(t) = \min\{Q - Q_s(t), S\}, \\ d_n = \min\{A(t) - D(t) + R(t), S\}.$$

That is, the packets will be sent when (1) the amount of data in the data queue is about to become larger than the length of the queue minus the size of the extra space $Q_s(t)$ or (2) the amount of data in the queue reaches the upper bound a packet can carry. In addition, the packet will carry as much data as possible. As can be seen, the only difference between the decision set determined by the online algorithm and OOD is the extra space reserved from the data queue. The above calculation repeats until we cannot find a feasible t_n. That is, $\forall t > t_{n-1}, A(t) - D(t) + R(t) > Q$. Suppose that the calculation stops at the Nth packet. Then, the decision set determined by the online algorithm is defined as follows.

Definition 2 (Online Determination)**.** *The packet sending decision set ρ given by*

$$\rho = (t_1, d_1), ..., (t_N, d_N)$$

is the online determination.

V. Performance Evaluation

In this section, we conduct extensive experiments to validate the feasibility of the proposed scheme and report some useful insights in practice.

A. Experiment Settings

The proposed data parasitizing scheme has been implemented on a WBAN testbed based on the Arduino open-source electronic prototyping platform. Each of the WBAN sensor devices, shown in Fig. 2, is comprised of two parts: e-Health sensors shielded with different biomedical sensing capabilities from Cooking Hacks [28] and an arduino duo micro-controller board [13] with Wi-Fi module enabling 2.4GHz IEEE 802.11b/g compliant radio support for wireless communication. The sensor device can be powered by either a computer via 5V USB connection or an external power supply like a battery. The SRAM size of the micro-controller

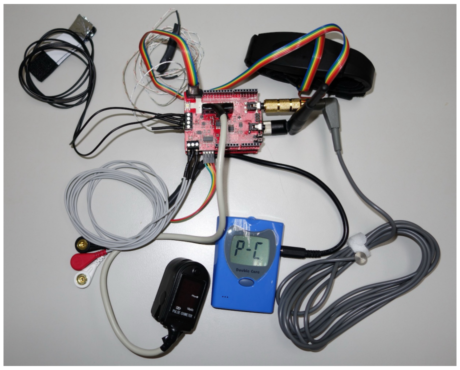


Fig. 2. The WBAN testbed and sensor devices

TABLE I
SENSING RATES OF BIOMEDICAL SENSORS

Sensor type	Data rate
Electrocardiogram (ECG)	12KBps
Accelerometer (ACC)	4 KBps
Pulse and oxygen in blood (SPO2)	2 KBps
Airflow (AF)	1.2 KBps
Glucometer (GLU)	200 Bps
Body temperature (BT)	15 Bps

(ATmega328) used in an arduino duo board is 2KB. Since some space of SRAM needs to be reserved for program execution, the size of the data queue Q is set at 1KB in our experiments. The sensor device tries to connect to a Wi-Fi network and then starts to process the measurement data based on the online algorithm of the proposed data parasitizing scheme.

We consider two different kinds of scenarios for investigation: one is in a controlled environment and the other is in a real-world environment. In the controlled environment, we set up a WBAN sensor device with a Wi-Fi AP and a web server in a private network, as shown in Fig. 3. The WBAN sensor device is connected to the AP via 802.11g wireless connection, and the server is behind the AP with wired Gigabit Ethernet connection. On the other hand, in the real-world environment, we place the WBAN sensor device at different public spaces, such as coffee shops, department stores, airports, and train stations with Wi-Fi hotspot services provided by WIFLY [29] and CHT Wi-Fi [30] in Taipei City. Based on the ICMP standard [14], the default value of the maximum amount of data a packet can carry S is set at 8 KB. The transmission rate of the wireless link can be decided by the sensor device itself. The rate can vary from 1 to 54 Mbps and is set at 1 Mbps by default. The sensing rates of the measurement data for different kinds of biomedical sensors available in our testbed are detailed in Table I. More detailed settings will be specified in the following subsections. For each setting, we perform 20 runs and demonstrate the average performance. The performance metric, system lifetime, is the interval between the time of the disconnection happening and that of the first data loss.

B. A Controlled Environment

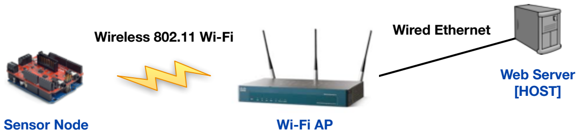
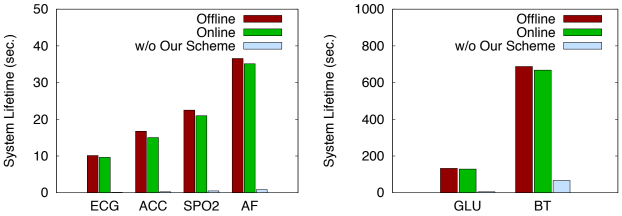


Fig. 3. The scenario investigated in the controlled environment



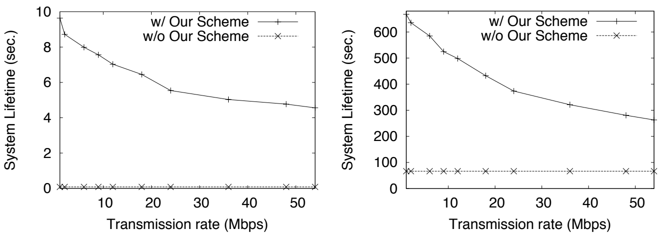
(a) High data rate (b) Low data rate

Fig. 4. Performance evaluation in the controlled environment

Fig. 4 and Fig. 5 show the evaluation results for the sensor devices with and without our proposed scheme under the controlled environment. We measure the system lifetime against different kinds of measurement data and transmission rates. Some observations and insights from the experimental results are summarized as follows.

Here we compare the performance of the online algorithm and that of the optimal offline algorithm. To make a fair comparison, the related information recorded during the experiment conducted for the online algorithm is used as the inputs of the optimal offline algorithm. In Fig. 4, the results demonstrate that the system lifetime achieved by the online algorithm is very close (about 90%) to that achieved by the optimal offline algorithm. In addition, the lifetime of a sensor device that keeps the measurement data intact with our scheme is significantly longer than that without our scheme. The sensor device without the data parasitizing scheme achieves quite low system lifetime (only a few microseconds in some cases), since it only relies on its very limited space of the data queue to store the measurement data. From Fig. 4, we can also observe the impact of different measurement data rates on the system lifetime. An intuitive phenomenon indicates that the system lifetime is shorter with the high data-rate biomedical sensors (Fig. 4(a)) than with low data-rate ones (Fig. 4(b)) since the length of the data queue and the capacity of the wireless communication link are limited.

Fig. 5 depicts the system lifetime against different transmission rates of sensor devices with and without the proposed scheme. The rate varies from 1 to 54 Mbps. The biomedical sensors tested here are ECG sensors and body temperature sensors. The reason for the sensor selection in this experiment is that they represent two WBAN devices with diverse measurement data rates (i.e., sensing rates): ECG for a high data rate (12 KBps) and body temperature for a low data rate (15 Bps). We observe that the system lifetime is shorter with a higher transmission rate. That is because the prorogation delay is shorter with a higher transmission rate, and thus the capacity of the wireless link is smaller. As a result, the system lifetime is shorter as we explained in Property 1. Nevertheless, the sensor device with our scheme still achieves much longer lifetime (dozens of seconds) than that without our scheme (less



(a) ECG

(b) Body temperature

Fig. 5. The impact of different transmission rates in the controlled environment

TABLE II
STATISTICS OF THE NETWORK CONDITION AT THE FIVE SITES

Site	Max. Link Speed (Mbps)	Round-trip (ms)	
		Avg.	Stddev
Train Station	12	240	12.17
Airport	28	197	6.45
MRT Station	12	227	23.56
Department Store	36	126	3.18
Coffee Shop	54	105	1.06

than one second) even with a high transmission rate. This verifies the effectiveness of our proposed scheme.

C. A Real-world Environment

For the experiments under a real-world environment, we take our sensor devices to public spaces with Wi-Fi hotspots. We let those devices connect to Wi-Fi networks with web-based authentication. Five sites are chosen for the experiments. They are the train station, airport, mass rapid transit (MRT) station, department store, and coffee shop in Taipei City. Before actually conducting the experiments, we measure the network conditions (the round-trip time of ICMP echo request/reply packets and their standard deviation) at each site by sending 100 ICMP echo packets with 8 KB data. As shown in Table II, the average round-trip times at the train station, airport, and MRT station are longer than that at the department store and coffee shop, since those public transit places are crowded with passengers. The more the people are using the Wi-Fi access, the longer the round-trip time. As for the standard deviation of the round-trip time, the MRT station is with a higher standard deviation than the airport and the train station, since the turnover rate of passengers at the MRT station is generally higher than that at the train station in Taipei. The coffee shop has the lowest standard deviation of the round-trip time, since people usually stay there longer than at other sites.

Then, we conduct some experiments to evaluate the system lifetime under the aforementioned five sites. The biomedical sensors tested here are ECG sensors and body temperature (BT) sensors, since they are with the highest and lowest data rates among the biomedical sensors available in our testbed. As shown in Table III, the system lifetime of sensor devices with our scheme in the MRT station is lowest among the five sites. Although the low transmission rate is beneficial to our scheme, as shown in Fig. 5, the high round-trip time deviation of the networks in the MRT station has an negative impact on the performance of our scheme, since the accuracy of the packet time estimation of the online algorithm is lower. This can be verified by comparing the lifetime in the train station with that in the MRT station since they both have a similar link speed, but a higher round-trip time deviation leads to a shorter system lifetime. Nevertheless, the system lifetime with our scheme is all significantly improved (a number of seconds to hundreds of seconds), compared with that without our scheme (as low as to dozens of microseconds), regardless of different sites and data rates. This verifies the feasibility of the proposed scheme in the real-world environment.

TABLE III
THE EXPERIMENTAL RESULTS AT THE FIVE SITES

Site \ System lifetime (sec.)	w/ the scheme		w/o the scheme	
	ECG	BT	ECG	BT
Train Station	5.66	418.77	0.083	66.64
Airport	4.67	372.94		
MRT Station	3.78	332.56		
Department Store	4.88	392.25		
Coffee Shop	3.98	356.43		

D. Discussion on the Feasibility of the Proposed Scheme

To implement the proposed scheme in the real world, there will be some feasibility issues to be considered. The issues include the energy consumption of sensors, the pre-processing (e.g. disconnection detection, radio switching from ZigBee to Wi-Fi, AP selection), and the possibility of being considered as a kind of denial of service attack on Wi-Fi APs.

Regarding the energy consumption of sensors, a larger scale and more sophisticated experiment is on-going. A preliminary result of the experiment indicates that the energy consumption for running our proposed scheme (i.e., wireless data transmission through public Wi-Fi access points) is close to that in the normal operation including data sensing and low-power wireless communications to smart phones. In other words, the energy consumption of the proposed scheme is not as much as we expect. We will continue to make a more detailed analysis on energy consumption and finding any possibility to lower the energy consumption further. Besides, the energy consumption can be decreased by optimizing hardware design. For example, some companies have started to develop the low-power ZigBee and Wi-Fi dual radio chip [31].

The pre-processing should be done before the data queue of the sensor device overflows. We are now developing high-efficient mechanisms for radio switching and Wi-Fi AP selection. According to the preliminary result of the aforementioned on-going experiment, detecting disconnection needs about 5-10 ms, the delay of radio switching is below 10 ms, and the delay of AP selection and association is about 70-100 ms. The numbers are expected to be lower as we are tuning the mechanisms to perfectly perform in terms of delay.

Even though the implementation of ICMP protocol should be considered mandatory, we recognize that the operators of Wi-Fi networks may block ICMP packets since those operators may consider our scheme as a kind of denial of service attack. We are examining the impact of the proposed scheme on the parasitized Wi-Fi networks. The preliminary result indicates that a single device only cause about 0.1% throughput degradation.

In this paper, we focus on dealing with the key requirements of "data parasitizing", finding perfect decisions on when to send the request packets and how much data to be embedded in the packets, to give a first glance over the feasibility of the proposed scheme. Instead of trying to solve all the issues at once, tackling the core problem of the proposed scheme is a most important first step. As mentioned above, a more realistic and comprehensive experiment on the testbed for the whole procedure of the proposed scheme is on-going. A thorough investigation report with effective solutions for those issues will be presented in the near future.

VI. Conclusions

In this paper, we propose a data parasitizing scheme for *wireless body area networks* (WBANs) to avoid excessive data loss due to temporary disconnection between sensors and their associated smart phone. The scheme parasitizes the measurement data of sensors on existing public Wi-Fi networks once the links to the smart phone become unavailable. We derive the key requirements for the data parasitizing scheme, and formulate an optimization problem to maximize the "system lifetime" (i.e., the time of data loss prevention). We present an optimal offline algorithm that inspires our online algorithm design. Experiments are conducted by implementing the proposed scheme on a WBAN testbed to evaluate its feasibility and effectiveness in practice. The experimental results verify that the concept of data parasitizing is a promising direction to prevent data loss due to temporary disconnection, without needing any external storage to be equipped on sensors.

Full-scale and more comprehensive experiments on the testbed are underway. A complete depiction of the proposed scheme including the solutions dealing with the issues mentioned in Section V and the details on the pre-processing (e.g. switching radio from ZigBee to Wi-Fi, selecting the most suitable AP) and post-processing (how to redirect the data back to the smart phone) will be presented in the near future.

Acknowledgement

Prof. Pang's work was supported in part by Excellent Research Projects of National Taiwan University under Grant 103R890821, by National Science Council under Grant NSC102-2221-E-002-075-MY2 and Grant NSC101-2221-E-002-018-MY2, Chunghwa Telecom, Information and Communications Research Laboratories, Industrial Technology Research Institute (ICL/ITRI), and Institute for Information Industry (III). Dr. Hsiu's work was supported by National Science Council of Taiwan, under Grant 102-2221-E-001-007-MY2.

References

[1] A. V. Chobanian, G. L. Bakris, H. R. Black, W. C. Cushman, L. A. Green, J. L. Izzo Jr., D. W. Jones, B. J. Materson, S. Oparil, J. T. Wright Jr., and E. J. Roccella, "The seventh report of the joint national committee on prevention, detection evaluation, and treatment of high blood pressure," *National High Blood Pressure Education Program*, Aug. 2009.

[2] B.H. Calhoun, J. Lach, D.D. Wentzloff, K. Whitehouse, A.T. Barth, J.K. Brown, Q. Li, S. Oh, N.E. Roberts, Y. Zhang, "Body Sensor Networks: A Holistic Approach From Silicon to Users," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 91–106, 2012.

[3] Z. Ren, G. Zhou, A. Pyles, M. Keally, W. Mao, H. Wang, "BodyT2: Throughput and Time Delay Performance Assurance for Heterogeneous BSNs," *IEEE INFOCOM*, 2011.

[4] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They Can Hear Your Heartbeats: Non-Invasive Security for Implantable Medical Devices," *ACM SIGCOMM*, 2011.

[5] A. Natarajan, B. de Silva, K.-K. Yap, and M. Motani, "Link layer behavior of body area networks at 2.4 ghz," *ACM MOBICOM*, 2009.

[6] R. Fu, Y. Ye, K. Pahlavan, "Characteristic and Modeling of Human Body Motions for Body Area Network Applications," *International Journal of Wireless Information Networks*, vol. 19, no. 3, pp. 219–228, 2012.

[7] Custodio V, Herrera FJ, Lpez G, Moreno JI, "A Review on Architectures and Communications Technologies for Wearable Health-Monitoring Systems," *Sensors*, vol. 12, no. 10, pp. 13 907–13 946, 2012.

[8] U. Mitra, B.A. Emken, Sangwon L., Ming L., V. Rozgic, G. Thatte, H. Vathsangam, D. Zois, M. Annavaram, S. Narayanan, M. Levorato, D. Spruijt-Metz, G. Sukhatme, "KNOWME: A Case Study in Wireless Body Area Sensor Network Design," *IEEE Communications Magazine*, vol. 50, no. 5, pp. 116–125, 2012.

[9] X. Liang, X. Li, Q. Shen, R. Lu, X. Lin, "Exploiting Prediction to Enable Secure and Reliable Routing in Wireless Body Area Networks," *IEEE INFOCOM*, 2012.

[10] D.-S. Zois, M. Levorato, and U. Mitra, "A POMDP Framework for Heterogeneous Sensor Selection in Wireless Body Area Networks," *IEEE INFOCOM Mini*, 2012.

[11] B. Latr, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," *Wireless Network*, vol. 17, no. 1, pp. 1–18, 2011.

[12] C.-Y. Chen; Y.-T. Chen; Y.-H. Tu; S.-Y. Yang; P.H. Chou, "EcoSpire: An Application Development Kit for an Ultra-Compact Wireless Sensing System," *IEEE Embedded Systems Letters*, vol. 1, no. 3, pp. 65–68, 2009.

[13] "Arduino due." [Online]. Available: http://arduino.cc/en/Main/ArduinoBoardDue

[14] *Requirements for Internet Hosts – Communication Layers*, IETF RFC-1122, 1989.

[15] S. Gupta, S. Lalwani, Y. Prakash, E. Elsharawy, and L. Schwiebert, "Towards a propagation model for wireless biomedical applications," *IEEE ICC*, 2003.

[16] A. Natarajan, B. Silva, K.-K. Yap, and M. Motani, "Towards a propagation model for wireless biomedical applications," *ACM MobiCom*, 2009.

[17] C. Hu, H. Kim, J. C. Hou, D. Chi, and S. Shankar N, "Provisioning Quality Controlled Medium Access in UltraWideBand WPANs," *IEEE INFOCOM*, 2006.

[18] Z. Ren, G. Zhou, A. Pyles, M. Keally, W. Mao, H. Wang, "BodyT2: Throughput and Time Delay Performance Assurance for Heterogeneous BSNs," *IEEE INFOCOM*, 2011.

[19] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, "They Can Hear Your Heartbeats: Non-Invasive Security for Implantable Medical Devices," *ACM SIGCOMM*, 2011.

[20] M.D. Renzo, R. M. Buehrer, J. Torres, "Pulse shape distortion and ranging accuracy in uwbbased body area networks for fullbody motion capture and gait analysis." *IEEE Globecom*, 2007.

[21] B. Latre, B. Braem, I. Moerman, C. Blondia, E. Reusens, W. Joseph, and P. Demeester, "A low-delay protocol for multihop wireless body area networks," *ICST MobiQuitous*, 2007.

[22] M. Quwaider and S. Biswas, "DTN routing in body sensor networks with dynamic postural partitioning," *Ad Hoc Networks*, vol. 8, no. 8, pp. 824–841, 2010.

[23] M. Quwaider, M. Taghizadeh, and S. Biswas, "Modeling on-body DTN packet routing delay in the presence of postural disconnections," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, no. 3, 2011.

[24] Y. Breitbart, C.-Y. Chan, M. Garofalakis, R. Rastogi, A. Silberschatz, "Efficiently monitoring bandwidth and latency in IP networks." *IEEE INFOCOM*, 2001.

[25] "Ping tunnel." [Online]. Available: http://www.cs.uit.no/ daniels/PingTunnel/

[26] "Icmptx (ip-over-icmp)." [Online]. Available: https://github.com/jakkarth/icmptx

[27] A. Natarajan, M. Motani, B. de Silva, K.-K. Yap, and K. C. Chua, "Investigating network architectures for body sensor networks," *HealthNet*, 2007.

[28] "e-Health Sensor Platform Complete Kit." [Online]. Available: http://www.cooking-hacks.com

[29] "WIFLY." [Online]. Available: http://www.wifly.com.tw

[30] "CHT Wi-Fi." [Online]. Available: http://wifi.hinet.net

[31] "Gainspan ultra low-power 802.11b/g/n + 802.15.4 single chip." [Online]. Available: http://www.gainspan.com/products/gs2000