

Loss Resilient Strategy in Body Sensor Networks

Nick Read

California State University Fresno
Department of Computer Science
Fresno, CA 93740, USA
+1 559-278-4792

nich.read@gmail.com

Ming Li

California State University Fresno
Department of Computer Science
Fresno, CA 93740, USA
+1 559-278-4792

mingli@csufresno.edu

Yu Cao

University of Tennessee
Department of Computer Science
Chattanooga, TN 37403, USA
+1 423-425-4351

yu-cao@utc.edu

Shih-Hsi Liu

California State University Fresno
Department of Computer Science
Fresno, CA 93740, USA
+1 559-278-4789

shliu@csufresno.edu

Todd Wilson

California State University Fresno
Department of Computer Science
Fresno, CA 93740, USA
+1 559-278-9138

twilson@csufresno.edu

B. Prabhakaran

The University of Texas at Dallas
Department of Computer Science
Richardson, TX 75080, USA
+1 972-883-4680

praba@utdallas.edu

ABSTRACT

Body sensor networks have gained continuous popularity in applications such as monitoring, sports training, and interactive gaming. However, data communication reliability is critical for the success of various applications such as ECG/EKG data analysis. In this paper, we propose a loss resilient strategy where prioritized scheduling and packet concatenation schemes are integrated for high network capacity and acceptable data reliability. The scheme is implemented at the application layer and therefore is compatible with existing standards such as IEEE 802.15.4. Performance evaluation on a real sensor test bed shows that the concatenation scheme significantly improve the performance and the prioritized scheduling ensures good reliability of high importance sensors.

Categories and Subject Descriptors

C.2.2: Network Protocols – protocol architecture.

General Terms

Performance, Design.

Keywords

IEEE 802.15.4, body sensor networks, quality of service.

1. INTRODUCTION

A body sensor network (BSN [6]) is a combination of wired, wireless, and embedded sensors on and around the body of a person. It has gained continuous popularity in applications such as medical monitoring, sports coaching, and interactive gaming with the decrease of the cost and size of sensors and the ubiquitous

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

BODYNETS 2011, November 07-08, Beijing, People's Republic of China

Copyright © 2012 ICST 978-1-936968-29-9

DOI 10.4108/icst.bodynets.2011.247076

presence of portable computing devices like smart phones.

As outlined by IEEE 802.15.6 task force [2], wireless sensor motes usually require high energy efficiency, high reliability, privacy, and desirable throughput. However, existing prevalent technologies such as ZigBee [5] and Bluetooth low energy [1] does not meet this demand. For example, Zigbee only provides up to 250kbps physical bandwidth. In this case, data loss may increase significantly when several sensors sample human related data and stream to the same data aggregator simultaneously, making it less scalable for wide deployment of BSN applications.

On the other hand, sensor heterogeneity imposes additional challenge for efficient bandwidth usage. First, body sensor data are usually ranging from a few bytes to less than 100 bytes, incurring high overhead for the existing IEEE 802.15.4 protocol. Second, some sensors (such as ECG/EKG) usually contain more important data than other sensors (such as blood pressure sensors). Therefore, how to maximize network capacity while providing reliable communications for high priority sensors is critical.

We observed that important sensors such as ECG/EKG usually sample data more frequently than less important sensors. This characteristic provides a unique way to improve network capacity by concatenating multiple small size frames [9] to a big one without affecting delay significantly. By this approach, the corresponding protocol overhead can be reduced, thereby improving capacity remarkably. Furthermore, to provide better reliability for streaming high priority sensor data, an adaptive data transmission scheme is designed. In this scheme, lower priority sensors reduce their data rate when high priority sensors experience considerable packet loss.

By integrating these two techniques, we propose an integrated strategy to improve the loss resilience of high importance sensors in BSNs. If high priority sensors experience considerable loss due to heavy traffic load, lower priority sensors enter a backoff phase and reduce their sending rate gradually. At the mean while, low priority sensors concatenate packets so that they still send the same amount of data. Experimental results with a test bed of six TelosB sensors have shown that the proposed approach improve the network performance significantly.

2. RELATED WORKS

Many of the related works in BSN research focus on designing algorithms that enables specific applications such as motion detection and medical information analysis. Zhou and Lu [11] proposed BodyQoS, a virtual MAC for quality of service scheduling in BSNs. The approach basically measures the effective bandwidth and adaptively allocates remaining resources to meet the QoS requirements of applications. A desirable feature of BodyQoS is that it does not require the modification of the underlying MAC layer implementation. Fulford-Jones and Malan [7] proposed CodeBlue, an ad hoc infrastructure for emergent medical care. In this project, several types of body sensors (e.g., pulse oximeter, ECG/EKG sensor) are individually connected to Zigbee enabled radio transmitters, which communicate with access points directly. Due to the ad hoc architecture and the capability of self-organizing, CodeBlue yields scalability for network expanding and flexibility to connect various wireless devices. Jiang and Cao [8] proposed CareNet, an integrated wireless environment used for remote health care systems. CareNet offers features such as high reliability and performance, scalability, security and integration with web based portal systems. High reliability is achieved by using two-tier architecture. Younis and Akkaya [10] proposed Distributed Queuing Body Area Network, a MAC protocol aiming at providing better QoS. It uses a cross-layer fuzzy rule based scheduling algorithm to optimize MAC layer performance in terms of QoS and energy efficiency.

There have been considerable standardization efforts during recent years. Among many potential technologies, Zigbee and Bluetooth are most widely deployed. ZigBee is a very low power, collision avoidance protocol optimized for lower power sensors. It has developed a health care specific protocol and are compliant with all IEEE 11073 devices as well as most other IEEE 802.15.4 wireless devices. Bluetooth supports high-bandwidth and many several existing devices and has a health care compliant version defined, but has very high power requirements and uptime for the radios. Bluetooth Low Energy [1] is a new proposed system from Bluetooth which will have lower energy requirements and still be interoperable with Bluetooth Classic, but details are still forthcoming at this time.

This paper is organized as follows. Section 3 introduces the test bed system architecture and sensor data characteristics. Section 4 overviews IEEE 802.15.4 MAC and described the proposed concatenation and prioritized scheduling strategies in details. Section 5 describes the test bed setup and discusses the experimental results. Finally, section 6 concludes the paper with future works.

3. SYSTEM MODEL

3.1 Test bed System Architecture

We established a wireless body sensor network test bed for our performance evaluation and protocol design. As illustrated in Figure 1, six TelosB sensor motes [3] and a computer with Java and a USB port are used. In this configuration, four TelosB sensor motes were used to simulate the sensors on the body, one was used as the data aggregator, and the last one was attached to the computer so the computer could act as the data storage and data processor. This simulates a fairly generic configuration where the data storage is directly attached to the data processor. It could simulate someone being monitored at home with a stationary data aggregator (like a wireless access point) and the server somewhere else, it could simulate someone being monitored with

a smart phone while they walk around, or it could simulate someone being monitored in a large building with a system of strategically placed data aggregators.

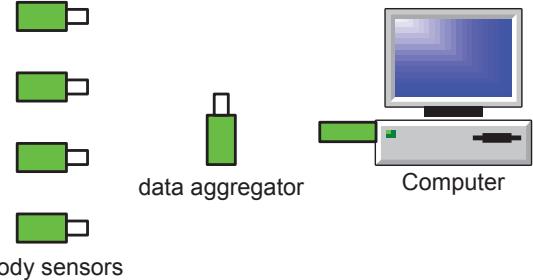


Figure 1. Testbed architecture.

Each real Crowsbow TelosB sensor mote (TPR2400) contains a IEEE 802.15.4/Zigbee compliant RF transceiver, integrated light, temperature, and humidity sensor, programmable memory and chip, and the USB connector. This sensor runs on TinyOS [4]. When connected to a computer through a USB port, it is able to communicate with other motes through the wireless radio and at the same time communicate with the computer through the serial port. The computer needs to have the appropriate code and the JRE so it can receive the sensor data.

TelosB motes are operated by two AA batteries or host computers but still operate at low energy consumption level with a maximum data rate of 250kbps and an indoor communication range of 20-30 meters. However, the low rate and long range of TelosB mote make it not the ideal option for body sensor networks. We anticipate better sensors that are tailored for BSNs in the future.

TinyOS [4] is an open source operating system designed for low-power wireless devices. This includes body sensors, smart buildings, smart meters, and ubiquitous computing. It is an embedded operating system written in nesC. TinyOS programs are built using a series of components that are software abstractions of the underlying hardware on the chip. It already has support for several wireless sensor mote hardware platforms.

3.2 Sensor Data Characteristics

Each TelosB sensor mote generates three blocks of data at every data collection, temperature, humidity, and a time stamp. Each block corresponds to two bytes. The time stamp is actually just a counter that increments every time a new data point is gathered. This information can be used to order the data when it is retrieved by the data processor. This means that at an absolute minimum we need three blocks of data sent every time. This increases by a factor of two (as we only need one time stamp) if we include more than one data point in the message.

The sending speed needs to approximate a reasonably generic use of a body sensor network. A position sensing program for human movements using a body sensor network is a common application and a good one to model. Human movement is, at a maximum, 60 Hz, but can be modeled very well with 30 Hz. This means a sending speed of once every 33 milliseconds. So three data blocks every 33 milliseconds should be the minimum goal. In addition, sensors have heterogeneous data characteristics. For example, an ECG/EKG sensor may sample data at much higher frequency than

blood pressure sensors. In this case, the overall data rate may be significantly higher. Finally, the built-in message formats in TinyOS is used for sensors, which includes a broadcast and a pseudo TCP mode where acknowledgement of message receipt is required. More fully implemented modes such as UDP and TCP can be adopted as well. But using this message formats does not affect the correctness and effectiveness of the proposed protocol.

4. LOSS RESILIENT PROTOCOL DESIGN

4.1 Overview of IEEE 802.15.4 MAC

IEEE 802.15.4 is a standard for physical and MAC layer of low-rate personal area networks. It focuses on very low energy communications among devices and offers 10 meters range and maximum bandwidth of 250kbps. This standard supports two nodes: *full-function device* (FFD) and *reduced-function devices* (RFD). FFD is usually designated as coordinators and RFDs can only communicate with FFDs. IEEE 802.15.4 is the basis of the widely support Zigbee specification.

Basically, with the underlying IEEE 802.15.4 MAC protocol used by TelosB motes, the physical bandwidth is limited to 250kbps. Furthermore, the aggregate throughput achieved by all the motes is also affected by traffic characteristics especially packet size. Due to the framing overhead in IEEE 802.15.4 MAC, the received throughput decreases when payload size is small. As shown in [9], concatenating multiple small payload size helps improve the performance significantly in IEEE 802.11.

4.2 Packet Concatenation

We propose to concatenate sampling data from sensors until it reaches a specific threshold before transmitting to the data aggregator. Another support of the concatenation scheme is attributed to the observation that some sensors, especially important sensors such as ECG/EKG sensors, sample data at a relatively higher frequency. Therefore, instead of immediately transmitting each sample, concatenating multiple samples and then sending out does not yield significant loss. For others such as blood pressure sensors that samples data infrequently, it might not be desirable since a higher delay may be created.

It may be desirable to perform some fine-tuning with the MAC message structure and design the concatenation at the MAC level for maximum throughput improvement. However, one advantage of our approach is the compatibility with IEEE 802.15.4 standard. We only perform packet concatenation at the application layer, not the MAC layer. Therefore, there is no modification required at the underlying layers. By reducing the overall framing overhead, we can achieve better aggregate throughput while maintaining a very high degree of interoperability with other devices as well as a standard format which makes it ideal for a test bed.

Finally, the key issue in the prescribed concatenation strategy is to determine the concatenation threshold, i.e., the maximum number of data blocks that we can practically put together. We start this investigation by performing various experiments to identify the relationship between the maximum throughput and message size. The result shows that a hard cap of 13 data blocks achieves the highest throughput. With more than 13 blocks, the hardware takes too long to compose the message and TinyOS's non-blocking property would prevent the message from being sent.

4.3 Priority based Adaptive Transmission

An additional issue in BSNs is to provide sufficient data transmission reliability. We focus on important sensors where reliable streaming is critical. For example, ECG/EKG sensors require reliable data set in order to identify frequency, rhythm, and signal strength/timing of heartbeats, therefore matching specific symptom patterns for further diagnosis. If data loss rate is significant, then the accuracy of the pattern recognition will decrease, leading to either false detection or failure of detection. For other sensors such as blood pressure sensors, higher loss rate is usually acceptable due to its relatively high data sample independence and less importance.

Therefore, we may define a priority for each sensor mote and include it in messages to the data aggregator. In our experiments, we use 1 for high priority and 0 for low priority. If necessary, we may further differentiate sensors by defining more priority levels. The data aggregator would then compare the last timestamp it received from that sensor to see if it had missed a packet. If this happened it would send a broadcast message telling sensors with a lower priority to send at a slower rate. By this scheme, high priority sensors receive more shares of bandwidth and the corresponding data loss reduces until reaching a satisfactory level.

To further optimize the performance, once a low priority sensor receives a broadcast message from a high priority sensor, it concatenates the data and reduces their sending rate proportionally. This allowed them to reduce their sending rate without reducing the amount of data being sent.

4.4 Discussion

One limitation of our current implementation is that the sensor priority is hardcoded into the program of the sensor mote and there's no way to guarantee that the sensor motes will receive the message. This problem can be alleviated by allowing sensors to receive messages telling them to change priority or coding them to change their priorities if necessary. The issue of no guarantee of message delivery can be alleviated by using the pseudo or true TCP message structures to contain each sensor mote samples. However, this drastically increases sending time and ties up the radio on the data aggregator for a much longer time than a single broadcast message.

5. EXPERIMENTAL RESULTS

In our experiments, each of these tests was set up with four motes simulating sensors, one mote simulating the data aggregator, and one mote attached to a computer. For the raw sending and concatenation tests, each of the sensors sends the same amount of data with an included timestamp. For the priority tests two different setups were used. One was set up to only use the backoff part of the priority scheme and the other used data concatenation on the sensors for lower priority sensors. In both cases one sensor was the highest priority and the other three sensors were the lowest. Additionally, the radio attached to the computer was used as a second data aggregator to reduce the difficulties of multiple message sizes.

We attempted to start the sensors as close to the same time as possible to create as homogenous a time frame as possible. We look at the time to receive a message averaged over 50 received messages so that any dropped messages or network delays will be factored into the result. The percentage of packets lost is investigated as well. These tests were designed to induce packet loss. Therefore, the packet loss should be higher than normal.

Figure 2 shows the time to send raw data messages. It can be observed that with concatenation, the system achieves a lower time to send the same number of message, indicating high network capacity. Thus, packet concatenation is effectiveness in improving the aggregate throughput.

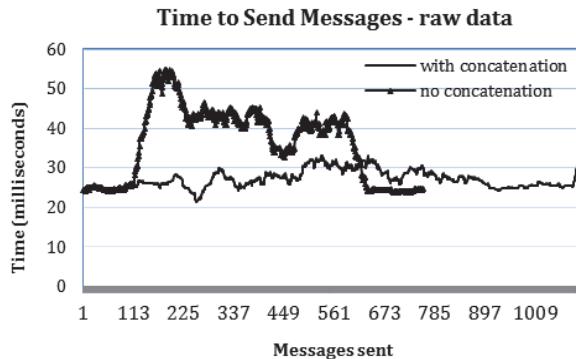


Figure 2. Time to send messages of raw data.

As shown in Table 1, for the raw data sending without concatenation, the packet loss ratio reaches 60% and the average time to send a message from the aggregator is 36.16 milliseconds. However, with concatenation, the packet loss is reduced to 42% and the average time to send a message from the aggregator is reduced to 27.13 milliseconds. Overall, a 30% reduction in packet loss and a 25% reduction in sending time have been achieved.

Table 1. Effectiveness of Packet Concatenation for Raw Data

	Without Concatenation	With Concatenation
Average time to send messages (ms)	36.16	27.13
Packet loss ratio (%)	60%	42%

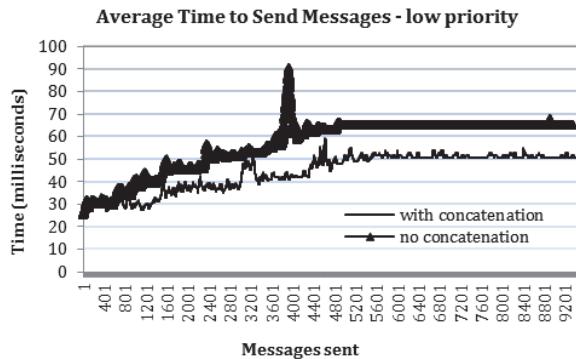


Figure 3. Average time to send low priority sensor data.
Note: concatenation is only for low priority data.

For the priority sending without concatenation at low priority sensors, the high priority sensor had a packet loss of 4% and an average time to send of 25.96 milliseconds. The lower priority sensors had an average packet loss of 1.91% and an average sending time of 56.41 milliseconds. This test was sending half as much data as other tests. For the priority sending with concatenation at low priority sensors, the high priority sensor

experienced a packet loss rate of 5% and an average time of 25.74 milliseconds to send a message. For other priority sensors, the packet loss averaged 8% with an average time of 46.97 milliseconds to send a message. Figure 3 shows the comparison of average time to send messages for low priority sensors in the case of with and without concatenation. We can see that the prioritized scheduling does yield acceptable loss rate (4%-5%) for high priority sensor. On the other hand, the improvement of average time to send messages of lower priority sensor data with concatenation is attributed to its lower overhead and higher maximum throughput.

6. CONCLUSION

In this paper, we investigated the issue of reducing loss rate and maintaining data communication reliability in IEEE 802.15.4 based body sensor networks. We established a real test bed with multiple sensors for performance evaluation. A sensor importance based prioritization approach is adopted to integrate with a packet concatenation scheme at the application layer. Experimental results validate the effectiveness of the proposed strategy.

7. REFERENCES

- [1] Bluetooth Low Energy Core Specification Version 4.0. http://www.bluetooth.com/English/Technology/Works/Pages/Bluetooth_low_energy_technology.aspx.
- [2] IEEE 802.15 WPAN Document Archive. DOI= <http://www.ieee802.org/15/pub/TG6.html>.
- [3] TelosB sensor data sheet. DOI=http://www.willow.co.uk/TelosB_Datasheet.pdf.
- [4] TinyOS. www.tinyos.net.
- [5] Zigbee Alliance. DOI=<http://www.zigbee.org/About/AboutAlliance/TheAlliance.aspx>.
- [6] Cordeiro, C. and Patel, M. Body area networking standardization: present and future directions *Proceedings of the ICST 2nd international conference on Body area networks*, Florence, Italy, 2007.
- [7] Fulford-Jones, T., Malan, D., Welsh, M. and Moulton, S., CodeBlue: An ad hoc sensor network infrastructure for emergency medical care. in *Proceedings of International Workshop on Body Sensor Networks*, (2004).
- [8] Jiang, S., Cao, Y., Iyengar, S., Kuryloski, P., Jafari, R., Xue, Y., Bajcsy, R. and Wicker, S. CareNet: an integrated wireless sensor networking environment for remote healthcare *Proceedings of the ICST 3rd international conference on Body area networks*, ICST, Tempe, Arizona, 2008.
- [9] Li, M., Zhu, H., Xiao, Y., Chlamtac, I. and Prabhakaran, B., Adaptive Frame Concatenation Mechanisms for QoS in Multi-Rate Wireless Ad Hoc Networks. in *INFOCOM 2008. The 27th Conference on Computer Communications*. IEEE, (2008), 1112-1120.
- [10] Younis, M., Akkaya, K., Eltoweissy, M. and Wadaa A., On handling QoS traffic in wireless sensor networks. in *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, (Big Island, HI, USA, 2004).
- [11] Zhou, G., Lu, J., Wan, C., Yarvis, M. and Stankovic, J., Bodyqos: Adaptive and radio-agnostic qos for body sensor networks. in *IEEE INFOCOM*, (Phoenix, AZ, 2008), 565-573.