Available online at www.sciencedirect.com
ScienceDirect
Ad Hoc Networks 6 (2008) 986–1012
Ad Hoc Networks
www.elsevier.com/locate/adhoc

Robust medical ad hoc sensor networks (MASN) with wavelet-based ECG data mining

Fei Hu [a,*], Meng Jiang [a], Laura Celentano [a], Yang Xiao [b]

[a] *Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY, United States*
[b] *Department of Computer Science, The University of Alabama, Tuscaloosa, AL 35487, United States*

Received 23 January 2007; received in revised form 1 July 2007; accepted 18 September 2007
Available online 1 October 2007

Abstract

Heart disease is the top elder killer in the world. To reduce the healthcare cost, it is a necessary tendency to deploy self-organized, wireless heart disease monitoring hardware/software systems. Telemedicine platform based on ad hoc interconnection of tiny ECG sensors, called medical ad hoc sensor networks (MASN), can provide a promising approach for performing low-cost, real-time, remote cardiac patient monitoring at any time. The contribution of this research is the design of a practical MASN hardware/software platform to perform real-time healthcare data collections. It has reliable, cluster-based communication scheme. Due to the radio broadcasting nature of wireless networks, a MASN has the risk of being attacked. This research also designs a low overhead medical security scheme to achieve confidential ECG data transmission in the wireless medium. Finally, our MASN system has the capability of keeping track of cardiac patients and extracting ECG features based on wavelet theories. Our MASN platform is very useful to practical medical monitoring applications.

Keywords: Ad hoc networks; Wireless sensor networks; Cardiac monitoring; Medical privacy; Medical security

1. Introduction

CARDIOVASCULAR diseases are one of the most wide-spread health problems and the single largest cause of morbidity and mortality in US and the Western world [1]. Based on the World Health Report 2000, each year the Coronary Artery Disease (CAD) kills an estimated 7 million people representing 13% of all male deaths and 12% of all female deaths. No country spends more per capita on healthcare delivery than the US. The entire nation has doubled its healthcare expenditure over the last two decades. Thus, low-cost, high-quality cardiac delivery is a critical challenge.

The progressive adoption of new paradigms in cardiovascular disease care (such as primary/secondary prevention and patient empowerment) promotes the development of novel care approaches [2,3] in which out-of-hospital monitoring and follow-up are basic aspects [4–8]. Therefore, the development and utilization of tele-cardiology systems that provide new modes of cardiac patient contact

* Corresponding author. Tel.: +1 585 475-5139.
E-mail addresses: fei.hu@rit.edu (F. Hu), yangxiao@ieee.org (Y. Xiao).

1570-8705/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2007.09.002

with the health care system is of increasing interest [35,36]. Most tele-cardiology systems use wearable devices (such as portable ECG recorder, Sphygmomanometer, Pulse Oximetry, and so on) to collect remote cardiac patients' physiological data (including 2 or 3 lead ECG, Blood pressure, Pulse rate). For reducing the number of network flows per patient, a wireless or wired Body Area Network (BAN) [37,3] is used to coordinate the operations of these body-worn medical devices and send the data to the patient's Personal Server (such as a PDA, i.e. Personal Digital Assistant). After post-processing (such as filtering ECG noise), the Personal Server then aggregates all the data into a network flow (including Text and Numbers) to be sent out to the health provider's remote medical server. In the following we classify the tele-cardiology systems into four generations.

The First-Generation (1G) [60] cardiac patient monitoring was based on traditional home telephone systems that uses low speed (<30 kbps) modems to modulate the collected ECG data into telephone voice signals and then send them the physician's office.

The Second-Generation (2G) [59] tele-cardiology systems have included the most popular wireless infrastructure–cellular networks. This allows physical mobility, i.e. the cardiac patients can be monitored anytime both in and outside their homes.

The 3G tele-cardiology systems [59] have attracted people's attentions since last decade. Many buildings or public places (such as airports, hotels, etc.) have installed high-speed (>10 Mbps) Wireless Local Area Networks (WLANs) or Bluetooth networks, which could be used to implement tele-cardiology systems as long as the cardiac patient's PDA has IEEE 802.11/15 wireless interfaces. In the WLAN/Bluetooth, each patient's PDA communicates with a Base-station or Access Point that then uses a high-speed Internet backbone to connect to a remote medical server.

The latest 4G [60] tele-cardiology systems incorporate a new type of wireless networks, called Mobile Ad hoc Networks (MANETs), which have the following advantages compared to 3G systems: (1) More flexible deployment: a 4G PDA can automatically search a nearby PDA within a certain wireless communication range (typically 100 feet–1 mile depending on the antenna power strength). Through such a "hop-to-hop" wireless relay, a PDA can send signals over a long distance with satisfactory transmission speed (100 k $\sim$2 Mbps) and finally reach a medical base-station that is connected to the Internet; However, in 3G WLAN/Bluetooth, all PDAs are required to directly (instead of using multi-hop) talk with a central point. It limits the moving range of a patient and also brings higher deployment cost due to the installation of base-stations. (2) Lower power consumption: "hop-to-hop" reduces wireless communication energy consumption than direct end-to-end communication [38,39]. This would increase the PDA battery lifetime that is a major concern in wireless systems.

It has been shown that Telemedicine through the ad hoc interconnection of ECG sensors is a promising approach to perform "automatic" heart beat anomaly detection [3]. Today, many ECG machines, both standard and continuous, are marketed as "portable" – but this does not always indicate that they are small and unobtrusive. By contrast, most such appliances receive power from an electrical outlet and are sufficiently heavy that they must be mounted on a cart and wheeled from one location to the next.

Low-power medical ad hoc sensor networks (MASN), consisting of mobile, low-cost ECG sensors that are attached to the patients' bodies, if deployed in nursing homes, will have the potential to significantly improve the ECG portability and timeliness. The tiny ECG sensors (weight < 250 g; size is comparable to a few coins) are particularly advantageous because of their low cost, radio communication capability, rapid deployment, and ease of integration with existing hospital computer systems. In next decade, we could even use MEMS (Micro-Electro-Mechanical System) technology to make an ECG sensor smaller than one coin [4].

MASN can also be regarded as a special type of wireless sensor networks (WSNs). WSN research is originally motivated by military applications such as battlefield surveillance. As the field slowly matured and technology rapidly advanced, it has found itself merging into many of the civilian applications as well, such as environment and habitat monitoring, home automations, traffic control, and more recently healthcare applications. Often equipped with wireless communication devices (i.e. RF boards) and microcontrollers, a WSN is a computer network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutions, at different locations [25]. Tiny sensors are most often referred as "motes".

A simple MASN scenario is shown in Fig. 1 (Top). Each patient's ECG signal could be automatically collected and processed (such as Analog-to-Digital conversion) by a small ECG sensor, and then be wirelessly sent to a remote ECG server for analysis purpose (such as using data classification to find out arrhythmia). If an ECG sensor reports any abnormal heart beat signals, an emergency communication channel established between the physician's office and the patient's wireless device (such as a beeper or cellular phone), will be used to send out alerts to provide the patient some medical suggestions (such as taking drugs or performing other further processing). In a more advanced MASN, (see Fig. 1 (Bottom)), a patient's ECG sensor can even use a neighbor sensor to relay its data if his/her distance is too far away from the ECG server. This communication mode is called "Multi-hop" wireless transmission. Multi-hop MASN not only extends communication distance but also saves the energy consumption of an ECG sensor since direct sensor-server long distance wireless communication is avoided through hop-to-hop relay.

Our MASN *hardware* mainly includes tiny ECG sensors and RF communication boards. The manufacturing cost for all the components of a sensor (such as resistors, amplifiers, etc.) is less than $80 each. If produced in large amount (>1000), the cost will be less than $50. Because our low-power design (through voltage scaling, low duty cycle, less RF collisions, and sleep control), the two AA batteries could provide the entire ECG sensor board 13 months of lifetime. Compared to the current commercial ECG measurement devices, our design is much lighter (<250 g), much cheaper (<$80), more portable without nurse's help, and more power-efficient (no AC power outlet is needed). Moreover, our MASN includes a new RF board design (Section 2), which saves more manufacturing cost than current medical sensor networks (such as CodeBlue [26]).

Our MASN has more advanced ECG transmission/processing *software* than current sensor networks such as CodeBlue [26]. For instance, our MASN software can use Support Vector Machine (SVM) [16] to classify different types of heart beats at fast speed and high accuracy. We have also enhanced CodeBlue MoteTrack [24] algorithm based on our RF chip characteristics in order to keep track of cardiac patients' locations more accurately (see Section 4). We have also built remote ECG sensor control software.

On the other hand, many hospitals hesitate to use advanced telemedicine systems because they are not sure the privacy-preserving capability of such systems. We have thus designed a cluster-based, end-to-end MASN security scheme in our MASN software modules in order to keep confidentiality

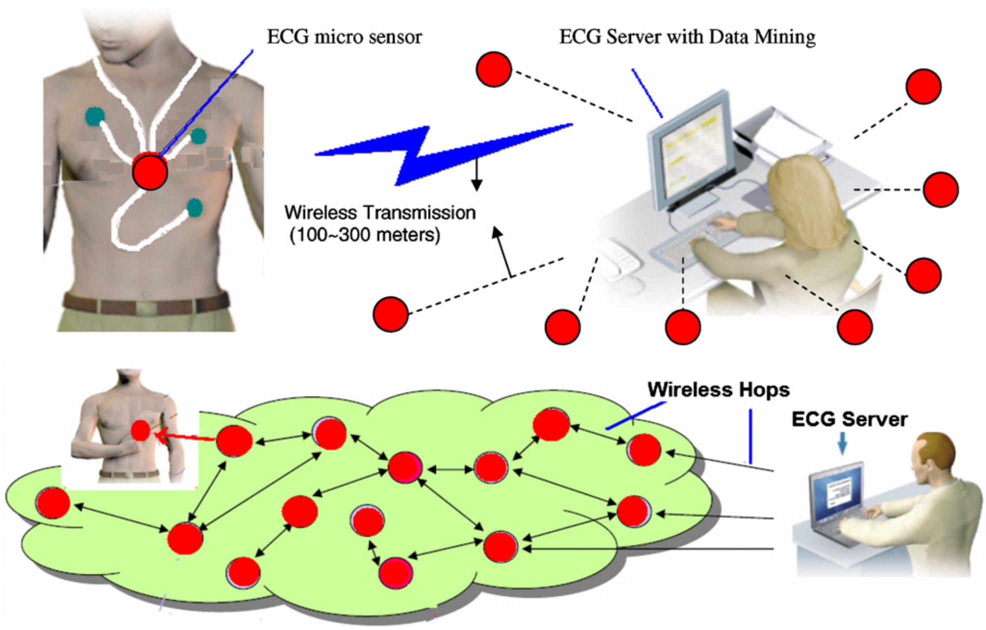


Fig. 1. Tele-cardiology sensor networks (MASNs): (Top): one-hop case; (Bottom) Multi-hop case.

during the patient-doctor ECG transmission. Our security algorithm considers the low-cost, low-memory characteristics of tiny ECG sensor boards. We thus designed a low-communication-overhead, low-complex encryption and decryption scheme.

This research serves the mission of cardiac healthcare very well in terms of improving the cardiac monitoring "quality" (a MASN can provide remote, automatic medical data collection and can thus capture emergency disease events from patients at anytime and anywhere), "safety" (we will build a secure wireless communication scheme), "efficiency" (our MASN is able to perform labor-free, low-delay patient monitoring), and "effectiveness" (our design targets "realistic" nursing home scenario and all networking schemes are designed for convenient healthcare).

The rest of this paper will be organized as follows: First, Section 2 explains our MASN hardware design principle. In Section 3, we will describe our reliable, cluster-based communication scheme. Section 4 then provides our software architecture. It also explains the positioning scheme in details. Next, Section 5 discusses medical data mining scheme. Section 6 has a low overhead medical data security strategy. Section 7 briefly mentions MASN hardware/software integration. Finally, Section 8 concludes this paper.

2. MASN hardware design

2.1. ECG sensors and RF communication hardware

Our MASN consists of large amount of wireless ECG communication units. Each unit is called a "mobile platform" in this work. These mobile platforms are essentially the wearable ECG devices that would be distributed among cardiac patients in order to offer continuous monitoring of the patients' vital signs.

As shown in Fig. 2, each platform is composed of a customized ECG sensor board providing connections to a 3-Lead ECG monitoring system, which is housed on a wireless communication board (also called RF motes). While the ECG sensor board gathers useful patient ECG data, the RF mote provides limited local signal processing capabilities (such as ECG noise filtering), and more importantly wireless communication for transmitting the ECG signals back to the server for feature extraction.

Fig. 3 shows the logic architecture components of the MASN mobile platform in Fig. 2.

2.1.1. "RF Mote" (see Fig. 2) design

Our original RF mote (see Fig. 2) was based on TelosB motes from Crossbow Inc. [27]. The TelosB mote is also referred to as the Tmote Sky. It is an ultra low-power wireless module intended for sensor networks applications. Regarded as the next-generation mote platform, it offers the on-chip RAM of 10 kB and also provides IEEE 802.15.4 Chipcon radio [28] with an integrated on-board antenna providing up to 125 m of range. Constructed around a TI MSP430 microcontroller [29], the TelosB worked for this project for its on-board ADC peripherals with expansion bays, from which the customized sensor board is connected to.

However, we found out a few problems when using TelosB: First, its unit price of TelosB is high in terms of large-scale MASN deployment. For instance, currently the TelosB RF mote is around $150 each [27] and there is no discount for

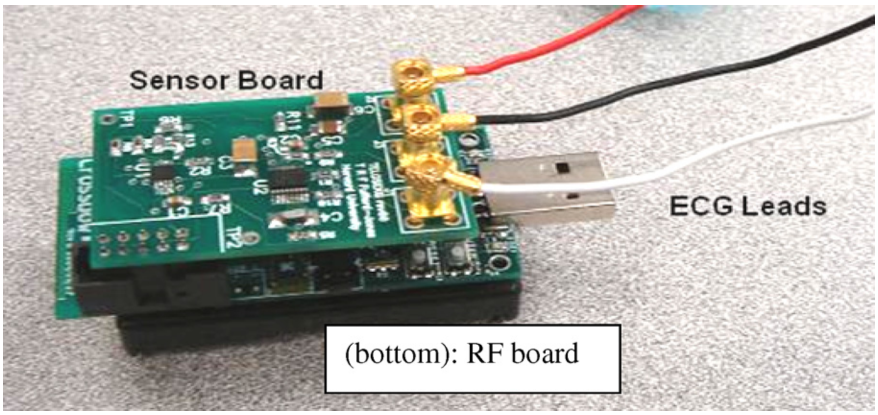


Fig. 2. Mobile platform appearance (includes ECG sensor + RF Mote).

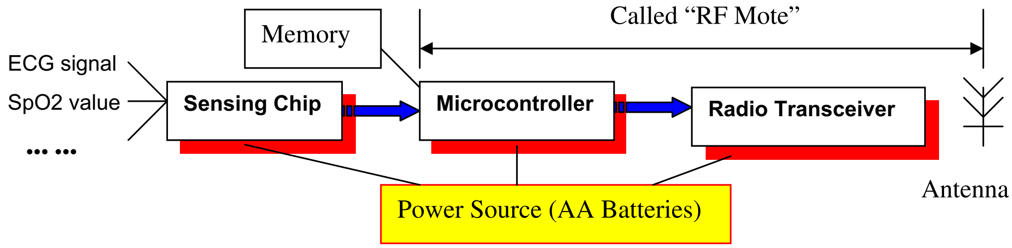


Fig. 3. MASN Mobile platform: logic architecture.

educational purpose. Because we need to use the MASN platform (with at least 30 motes in each MASN network) to train large-amount of computer engineering/science students, we decided to build our own RF boards. Secondly, its power lifetime is around 3–6 months depending on how often the ECG signal is transmitted back to the server, which is somewhat short for medical applications (ideally, we wish the cardiac patient can carry such a low-cost ECG sensor for at least one year without worrying about power exhaustion). Thirdly, its radio components cannot be enhanced (we cannot use a better radio transceiver/antenna to reach a longer distance).

Due to the above reasons, we have used Ember CPU-RF chips [30] to build our own RF motes. As shown in Fig. 4, it is also AA battery driven. The RF mote size is a little larger than 2 AA batteries. The cost for electronic parts is $11.06 per board. The estimated quote of PCB fabrication (mass production) is $1.93 per board. The estimated cost for board assembly is $5.00 per board. This gives a total cost of only $17.99 per mote (mass production). The heart of the RF board is the Micro Central Unit (MCU)/ZigBee [31] Transceiver unit. Multiple options and configurations were considered before selecting the final option. The two options that resulted from this were using a separate MCU and Transceiver or using a SoC (System-on-Chip) that incorporates the two devices together. The SoC option was chosen as it would be cheaper to implement, decrease programming complexity, and create an easier Printed Circuit Board (PCB) layout, as there will be fewer parts to layout.

The Ember EM250 SoC [30] was selected for use in the ZigBee Data Forwarding Unit (DFU). The EM250 also includes 128 kB of onboard Flash Read Only Memory (ROM). It also allows for three different modes of operation. The Active operation will allow for execution of the program code, typically using 8.5 mA of current. The Idle operation allows for the MCU to shut down until an interrupt occurs while allowing peripherals and the transceiver to operate normally. The EM250 also allows for a Deep Sleep operation which powers down the MCU and Transceiver until either an external interrupt or a timer wakes the device. In the Deep Sleep operation, the EM250 typically uses 1.5 mA of current. The EM250 has four ADCs, of which two are used for use of capturing analog data. The digital interrupts can be used for receiving 1-bit digital data while either 'awake' or 'sleeping'. The EM250 also has the capability of communicating over serial peripheral interface (SPI), allowing for future revi-

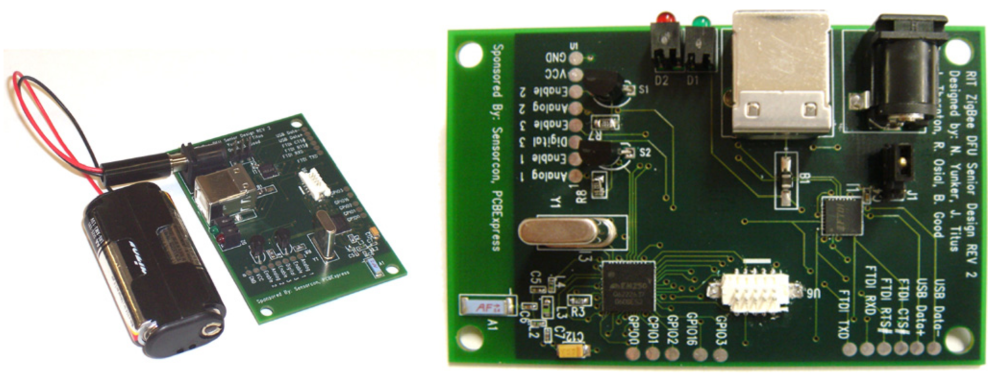


Fig. 4. Our built RF Board with ECG wireless communication capabilities.

sions to include serial data from sensors. Fig. 5 shows the schematic diagram of the core part of the RF board – Ember integrated CPU-RF chip.

Fig. 5 shows the connection between ECG sensor board (to be discussed later) and our built RF board. The RF board takes the analog ECG data (sensed from a patient's body), converts it to digital format, then uses network protocols to form packets, and finally sends out through RF antenna (see Fig. 6 on entire packet forming procedure). Its RF transceiver can also receive ECG data from a neighbor RF board (to achieve patient-to-patient relay communications).

2.1.2. Mobile platform – ECG sensor design

Our ECG sensor board design is assisted by Harvard University CodeBlue team [26]. The different styles of packaging information are the surface mounting types that correspond to the PCB layout choices.

The ECG lead extensions from the sensor board are pin-compatible and color coded to standard 3-Lead ECG monitoring systems. While there are different flavors of physiological chest leads, this system was designed to match any 3-Lead ECG Snap Set Leadwires. The Snap Set may be used to collect data by attaching to it the appropriate jellied ECG conductive adhesive electrodes if real people were to be used for testing purposes. An alternative would be ECG signal simulators, as explained next.

2.2. ECG signal generator

The ECG generator used in this project is the Model 430B, 12-lead ECG generator as shown in

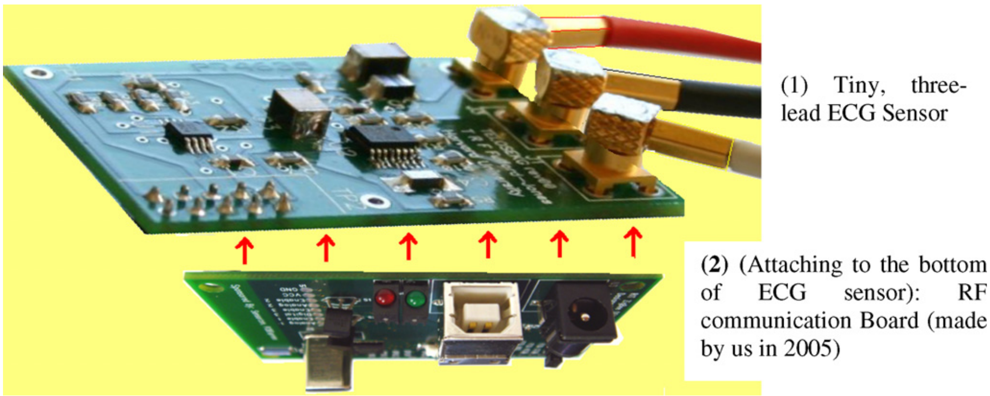


Fig. 5. The connection between ECG sensor and our built RF mote.

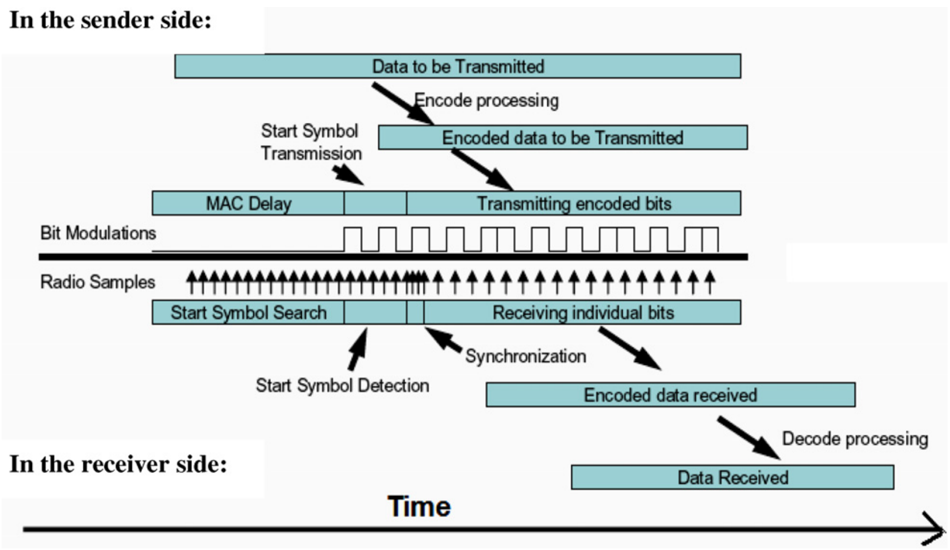


Fig. 6. The working procedure of a RF Mote [32].

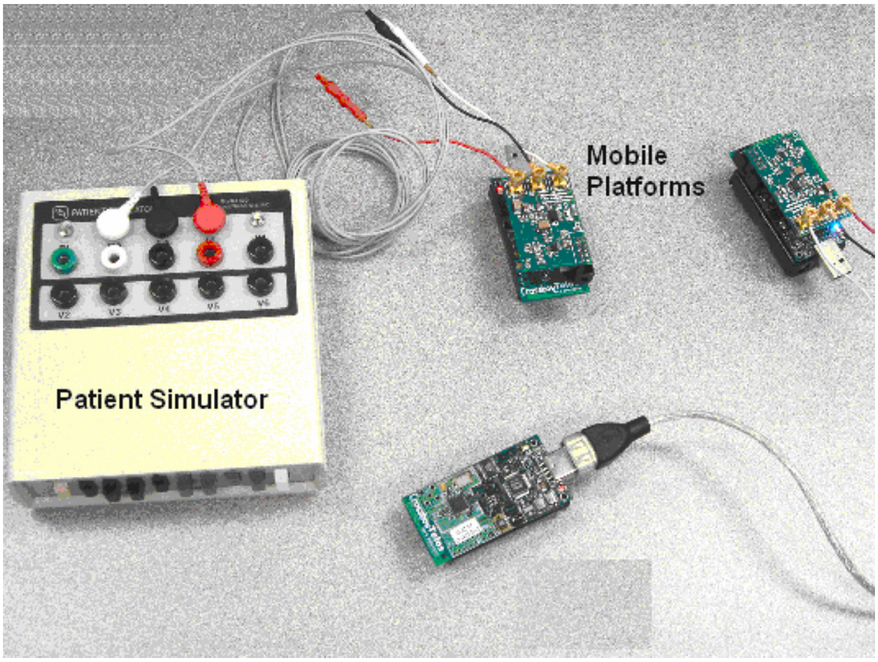


Fig. 7. Model 430B Patient Simulator (see the white box).

Fig. 7 (see the white box). This generator provides a complete PQRST waveform at six preset rates (60, 75, 100, 120, 150, and 200 BPM) as well as six preset amplitudes (0.1, 0.2, 0.5, 1.0, 2.0, and 5.0 mV). It is also capable of generating square waves using its 5 ECG snaps plus 10 banana jacks. This provides a good testing interface even if this project will be adapted into a 12-lead monitoring system in the future. Fig. 9 also shows the connection between 430B ECG simulator and our designed RF communication boards. The ECG signal collected from 430B can be transmitted to a computer (not shown in Fig. 7) through the RF board antenna.

The ECG data collection is a two-step process (next section has more details). The first step involves the sensor network communication that takes place between the mobile platforms and the receiving sensor mote connected to the workstation. After this step, all of the useful patient data have been collected and now reside onboard the workstation. The next level of communication occurs within the workstation environment, where a MATLAB server (a software module) is created to transfer data from a Java runtime environment into the MATLAB workspace via localhost connection. This is the final procedure before sending the patient data for signal processing, which leads to feature extraction. Our RF mote devices have IEEE 802.15.4-compliant radio capabilities and a communication range of 125 m. These sensor motes operate under the TinyOS environment and may be programmed using the NesC language (to be discussed next).

3. Reliable MASN communication protocols

3.1. Enhanced cluster-based MASN data transmission

It is important to achieve the fast and reliable detection of the ECG signals from the patients. Grouping the wireless sensor nodes in clusters to detect signals is carried out for prolonged MASN lifetime, load balancing and scalability. We propose a cluster based, energy-aware ECG collection scheme where the ECG data are reliably relayed to the sink[1] in the form of aggregated data packets. The clustering scheme provides faster and better event detection and reliability control capabilities to the areas of the network where event is occurring. It also reduces the overhead, latency and loss of even information due to the cluster rotation.

[1] In our MASN, we refer the ECG server as a “sink”.

The clustering schemes have been previously investigated as either stand-alone protocols for ad hoc networks, e.g., [40–43], or in the context of routing protocols, e.g., [44–48]. Several clustering algorithms have been proposed for the purpose of reducing energy consumption and extending lifetime of a sensor network. Low Energy Adaptive Clustering Hierarchy (LEACH) [47] is a distributed algorithm for sensor networks in which the sensors elect themselves as Cluster-Heads (CHs)[2] with some probability and broadcast their decisions. It assumes that all nodes can hear each other which is not a good assumption for randomly distributed sensor nodes. LEACH also assumes a large difference between the CH and normal node power requirement. A small difference would cause LEACH to become less effective.

Power Information Gathering in Sensor Information Systems (PEGASIS) scheme [49] is an improvement of LEACH, in which the key idea is to form a chain among the sensor nodes so that each node will receive from and transmit to its close neighbors. The gathered data moves from node to node, gets aggregated and eventually a leader node transmits it to the sink. The leader node will rotate in each round to have energy load evenly distributed among the sensor nodes. Its disadvantage is large time delay and faster energy depletion. The nodes far away from the sink will require more energy to transmit data to the sink.

A Hybrid, Energy-Efficient, Distributed Clustering approach for ad hoc sensor networks (HEED) [49] periodically selects CHs according to a hybrid of their node residual energy and a secondary parameter, average minimum reachability power (AMRP).

The above-mentioned clustering scheme cannot send the event packets to the sink as fast as possible without vital information loss. They also may have considerable latency involved in the set up of the clusters and overhead messages in the schemes due to random rotation of the cluster heads. Most of the schemes do not take into account the message losses due to collisions and congestion at the sensor nodes. The cluster rotation involves a complete change over of the entire topology, which requires synchronized control and consumes lot of sensor energy. There is no reliability control in these schemes for proper event detection at the sink, which could also help in efficiently utilizing the scarce energy resource of the sensor nodes.

Our proposed MASN routing scheme is different from LEACH [47] and the abovementioned other clustering schemes due to our consideration of energy level determination of sensor nodes, event-triggered and energy-aware cluster formation, dynamic adaptation of reliability based on the cluster member density and event proximity. The details are as follows:

We assume that the sensor nodes know their maximum energy ($E_{\max}$), residual energy (E_{R}) and threshold energy (E_{th}). Here E_{th} is the minimum energy required by the sensor nodes to identify themselves in one of the 'n' energy levels. A sensor node with $E_{\mathrm{R}} \leqslant E_{\mathrm{th}}$ belongs to the energy level '0'. Initially the energy of a sensor node is divided into n levels as shown below:

$$n = \left\lceil \log_x \frac{E_{\max}}{E_{\mathrm{th}}} \right\rceil, \tag{1}$$

where the energy range of a level L is defined as the difference between the upper and lower energy values and 'x' is the ratio between the maximum and minimum values of a level. The value of 'x' depends on the requirement of the application. The energy level (L) of a sensor node is determined as:

$$\text{If}(E_{\mathrm{R}} < E_{\mathrm{th}})L = 0; \quad \text{Else } L = n - \left\lfloor \frac{E_{\max}}{E_{\mathrm{R}}} \right\rfloor. \tag{2}$$

A sensor node decides to participate in the cluster formation process if amplitude of the event parameter that it detects crosses a predetermined threshold 'Δ'. Here the value of 'Δ' depends on the measured event parameter.

While forming clusters the sensors with the highest energy level (L) are given opportunity to become the CHs, to ensure longer cluster lifetime. In areas lacking high energy sensor nodes the lower energy sensor nodes take initiative to form CHs. This is mainly to ensure that the primary purpose of reliable event detection at the sink is achieved. The sensor nodes then elect their cluster heads based on the energy level and the AMRP value. Here the AMRP is defined as the average minimum power level required by the 'r' neighboring nodes to reach the sensor node claiming to become the CH as shown below [50]:

[2] Cluster-Head (CH): a small amount of sensors with higher energy storage in LEACH are chosen as CHs. All sensors only send data to a local CH. The CH then search a neighboring CH to relay the data until finally reaching the sink.

$$\mathrm{AMRP} = \frac{\sum_{i=1}^{r} \min \mathrm{PWR}_i}{r}, \tag{3}$$

where MinPWR_i denotes the minimum power level required by a node v_i, $1 \leqslant i \leqslant r$, to communicate with the CH and 'r' is the number of neighbor nodes. The sensor nodes advertise themselves as CH based on their energy level. The sensor node claiming to be a CH broadcasts the advertisement message to its neighbors using maximum power (MaxPWR). The normalized AMRP is defined as the ratio of AMRP to that of the MaxPWR.

The other sensor nodes on receiving the advertisements decide to join a CH based on a function of CH energy level and communication power. Every sensor node waits for a random time before advertising itself to other sensor nodes to become a CH. This delay time for sending the advertisement message is based on a function of the energy level (L) of the sensor node and normalized average minimum reachability power (nAMRP).

The sink assigns a reliability value 'REL' for an event in terms of the total number of packets of the event required to be reported in a time 'T' at the sink. This reliability factor is distributed among the clusters formed in the event area based on (i) the number of sensor nodes in the cluster, and (ii) the cluster-event proximity. Every CH of the event area transmits the number of its cluster members in the aggregated data packet header to the sink through multi-hop. Analyzing the values of the measured event parameters in the aggregated data packets the sink knows which of the CHs are closest to the event. The sink assigns a reliability value to each cluster shown below:

$$\mathrm{CR}_i = \frac{\mathrm{REL} * (J_i)(m_i)}{\sum_{i=1}^{z} J_i m_i}, \tag{4}$$

where CR_i is reliability assigned to ith cluster, z is the number of clusters, J_i is the event proximity for its cluster, and m_i is the number of sensor nodes in the cluster. If $J_i = 1$ then the reliability is distributed among all the clusters based on their member density. By assigning higher value of J_i, the sink can acquire more number of packets from the clusters closer to the event. The event proximity parameter J_i varies from cluster to cluster from a minimum value of 0 to a maximum value of 1.

The sink will vary the reliability values for the clusters if the event propagates to other areas. If the event propagates to other areas of the network, their sensors will also form clusters based on the values of the measured event parameters. This idea of 'Dynamic Reliability Adaptation' at the sink is helpful obtaining maximum information of the event.

3.2. MASN routing performance

3.2.1. Energy consumption

A major concern in MASN networking design is energy consumption. Our experiments have shown that most of sensor battery is consumed in radio communications instead of in local data processing (such as ECG compression) or sensing (see Fig. 8). Therefore, any MASN networking protocols (such as finding optimal route) should be of low-complexity to save energy consumption.

3.2.2. Throughput

For the better observation of a patient's health condition, a sensor can send out data at high reporting frequency and then use a high data rate to send out the large amount of sensed data wirelessly. Fig. 9 shows the packet reception ratio (the number of "received" packets divided by the number of "transmitted" packets) for different sending rates (number of network packets per second). We can

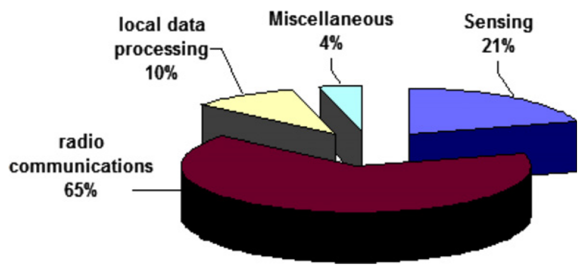


Fig. 8. Energy consumption of MASN.

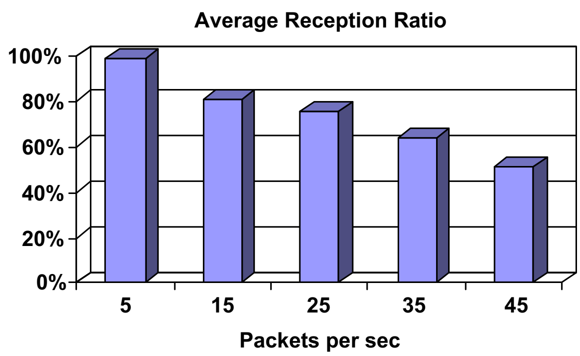


Fig. 9. Reception ratio–sending rate.

see that the MASN performance drops sharply if the sending rate is higher than 25 packets/s. Thus it is important to use a reasonable reporting frequency in each sensor.

3.2.3. Scalability

We have investigated the MASN performance with the increasing of number of sensors (it also means more patients since each patient carries one sensor). Our MASN system can still maintain good performance (reception ratio > 80%) even there are large amount of MSS (see Fig. 10). It indicates that our MASN will be suitable to a large Nursing Home.

3.2.4. Mobility

We have tested the MASN delay performance under users' mobility behaviors. Currently, our system cannot achieve real-time data collection (delay > 10 s) if the users move quickly (such as at 30 mph) (see Fig. 11).

3.2.5. Delay

We define "aggregated packet delay" as the time taken for the first aggregated event packet to reach the sink from the time an event is detected by the sensor nodes. This parameter represents the speed of reaction of the network to the event occurrence. In the proposed as well as HEED [50] schemes, we consider that clusters are formed 'on the fly' when the event occurs. In our experimental results (see Fig. 12), HEED scheme consumes more time for the first aggregate data packet to reach the sink due to the set up phase. In this phase no packets are reported to the sink and clusters are formed with the help of overhead messages. In the proposed scheme the event packets are transmitted to the sink even as the clusters are being formed.

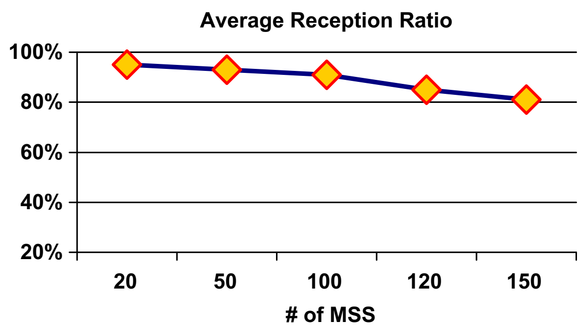


Fig. 10. Reception ratio–no. of MSS.

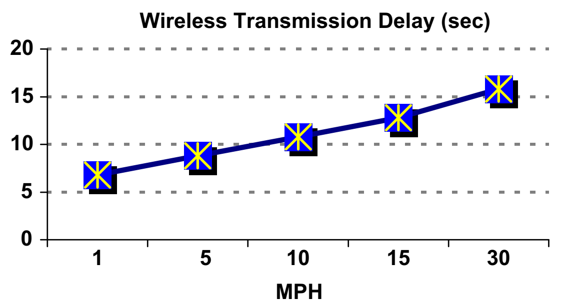


Fig. 11. End-to-end delay–mobility speed.

4. MASN software design

4.1. ECG sensor mote wireless communication software

All of our MASN RF mote control software runs in a special operating system called TinyOS [33]. Developed primarily by the University of California, Berkeley in cooperation with Intel Research, TinyOS is an open-source embedded operating system designed for wireless sensor networks. NesC is a programming language designed for applications targeting the TinyOS platform. Again by University of California, Berkeley and Intel Research, it is an extension to the C programming language that is component based as the TinyOS operating system. The most important feature of this programming language is that it produces fairly small sized code to be able to load on to sensor network nodes.

In our Medical Server that receives all patients' ECG data, we can monitor the entire MASN network topology. As shown in Fig. 13, each patient's ECG data can be collected remotely. The relative location of each patient can also be monitored through our patient tracking software (see Section 4.2). If two patients are close enough, a radio link will be shown between them to indicate the possibility of transmitting ECG data between them (in Fig. 13, ECG RF motes in Patient ID = 1 and ID = 3 can talk with each other).

An important feature of our MASN software is that we are able to control the ECG sensors' performance parameters (such as ECG detection threshold) through the remote command transmission from the server to any ECG sensor. Fig. 14 shows our sensor control GUI (Graphical User Interface). We can set up the ECG server (i.e. the MASN workstation) control parameters to change the sensors'

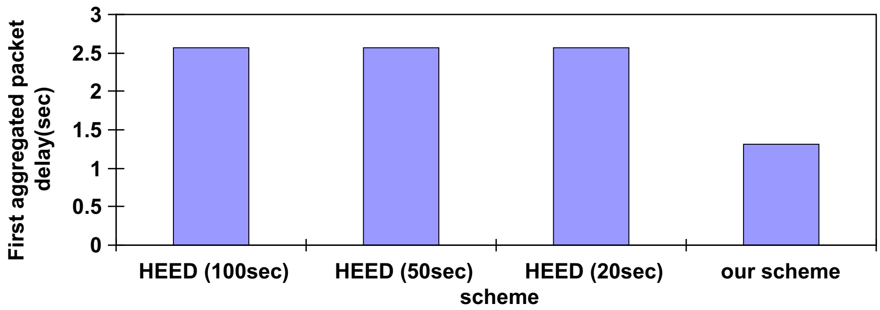


Fig. 12. The first aggregated packet delay from the cluster head to the sink.

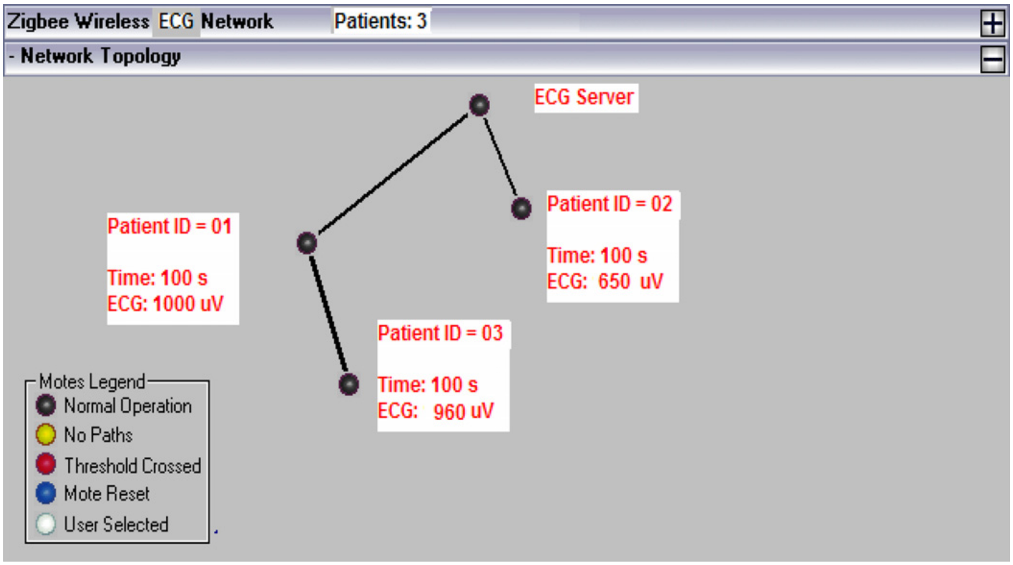


Fig. 13. Cardiac monitoring software for a simple Nursing Home with three cardiac patients.

detection frequency (i.e. how many ECG values we should collect in each second). As we know, a higher detection frequency can bring higher ECG signal quality, however, it also causes the higher power consumption in each sensor, and more memory storage overhead in each RF board. A good balance is needed. Here we collect ECG values every 0.01 seconds, which is good enough to capture each change of heart beats.

The software used to govern the sensor network communication and displaying the received patient data on the workstation is based on a program called VitalDust Plus [26]. This software is essentially a stripped down version of the CodeBlue [26] software that provides a simple demonstration of its wireless pulse oximeter and wireless ECK devices. The software has two parts, the TinyOS software for the mobile platforms to sample and transmit vital sign data over the radio, and a Java GUI application to display the vital signs a graphical form.

We have enhanced VitalDust Plus in many aspects such as network topology monitoring, real-time ECG display, etc. We call our MASN monitoring software as Flavor RIT, which made several functional additions to the Java applications. The most notable enhancements are the inclusions of MATLAB support and the ability to select data, at run time, from only the desired patient for feature extraction (see Section 3.3). Some of the unused features are also removed from the original GUI. Fig. 15 shows a screen shot of Flavor RIT while it is receiving patient data from two separate mobile platforms: mote30 and mote40. The patient

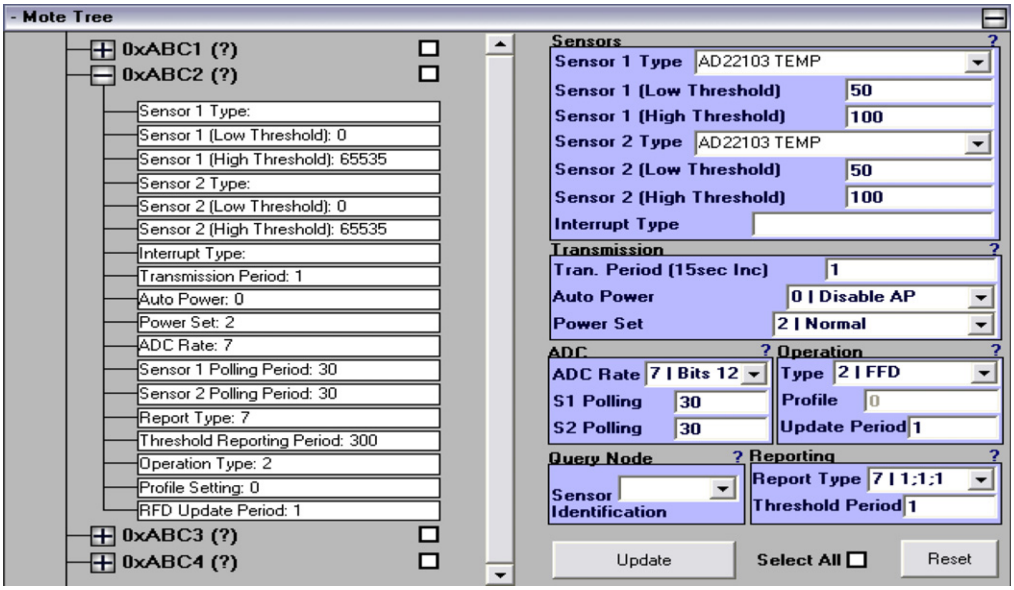


Fig. 14. ECG sensor parameters remote control software.

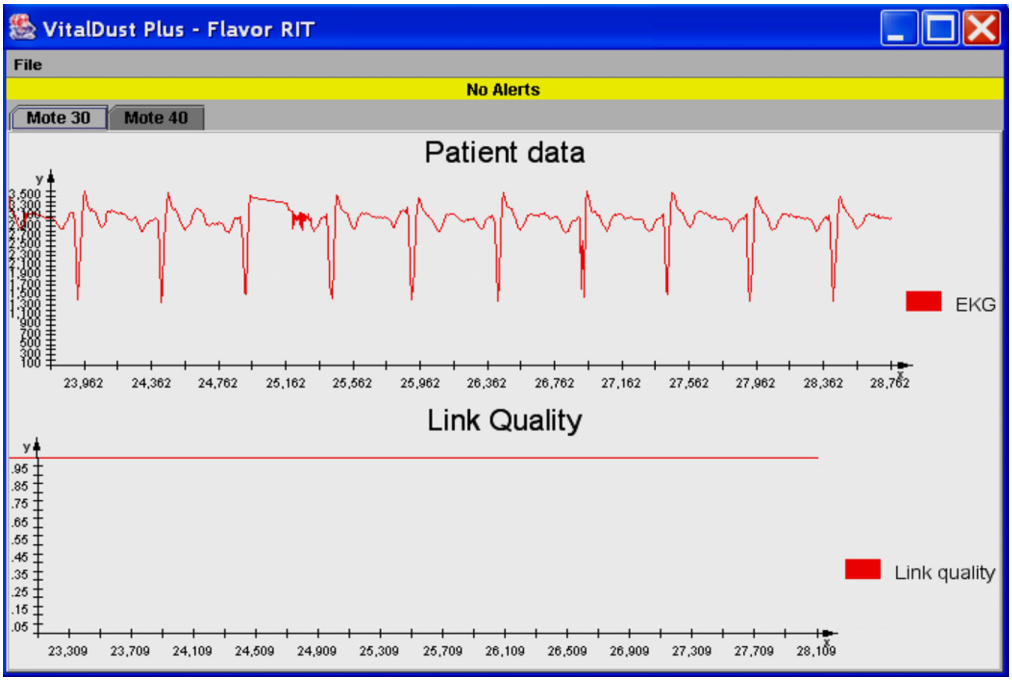


Fig. 15. Enhanced VitalDust Plus – Flavor RIT.

data field is displaying the ECG waveform associated with the selected mobile platform. Only data from the currently selected mobile platform are sent to MATLAB for signal processing. The link quality field shows the quality of the wireless signal also associated with the selected mobile platform.

4.2. Cardiac patients positioning software

Because our MASN software will be used for nursing homes/hospitals patients monitoring, it is important to keep track of patients' locations when they move around. By knowing the exact positions of the patients, the system can quickly lead the medical professionals to the desired locations, thus saving treatment time. Determining the location of a particular sensor in a wireless sensor network is an extremely difficult problem facing the wireless sensor network research community. GPS is far too expensive a solution for wireless sensor networks. The goal of producing wireless sensor nodes for less than one dollar would be severely compromised. Additionally GPS consumes far too much power to be a realistic localization solution for sensor networks that run on limited battery power.

MoteTrack [24,53] is a robust, decentralized localization algorithm to RF-based location tracking. Its purpose is the accurate location tracking of motes, which are small, lower-power, battery operated devices that can be readily embedded into equipment or the environment. Using radio signal information alone, it is possible to determine the location of a roaming node at close to meter-level accuracy. MoteTrack can tolerate the failure of up to 60% of the beacon nodes without severely degrading accuracy, making the system suitable for deployment in highly volatile conditions. In MoteTrack, a building or other area is populated with a number of motes acting as beacon nodes. Beacon nodes broadcast periodic signatures, which consist of the format {sourceID, powerLevel, meanRSSI}. The sourceID is the unique identifier of the beacon node, powerLevel is the transmission power level used to broadcast the message, and meanRSSI is the mean received signal strength indication (RSSI) of a set of beacon messages received over some time interval. Each mobile node that wishes to determine its location listens for some period of time to acquire a signature, consisting of the set of beacon messages received over some time interval. Finally, a reference signature is defined as a signature combined with a known three-dimensional location (x, y, z).

The location estimation problem consists of a two-phase process: an offline collection of reference signatures followed by online location estimation. As in other signature-based systems, the reference signature database is acquired manually by a user with a laptop and a radio receiver. Each reference signature, shown as gray dots in Fig. 16 [53], consists of a set of signature tuples of the form {sourceID, powerLevel, meanRSSI}. sourceID is the beacon node ID, powerLevel is the transmit power level of the beacon message, and meanRSSI is the mean received signal strength indication (RSSI) of a set of beacon messages received over some time interval. Each signature is mapped to a known location by the user acquiring the signature database.

In MoteTrack, beacon nodes broadcast beacon messages at a range of transmission power levels. Using multiple transmission power levels will cause a signal to propagate at various levels in its medium and therefore exhibit different characteristics at the receiver. In the most extreme case, a slight increase in the transmission power may make the difference between whether or not a signal is heard by a receiver. Varying transmission power therefore diversifies the set of measurements obtained by receiving nodes and in fact increases the accuracy of tracking by several meters. The MoteTrack algorithm assumes that the most relevant (closest in signature space) reference signatures are stored on the beacon node with the strongest signal. The mobile node sends a request to the beacon node from which it received the strongest RSSI, and only that beacon node estimates the mobile node's location. As long as this beacon node stores an appropriate slice of the reference signature database, this should produce very accurate results. The communication cost is very low because only one reply is sent to the mobile node containing its location coordinates.

4.2.1. Our enhancements to motetrack algorithm

After initial attempts to install and run the original unaltered MoteTrack codes from Harvard [24], it was discovered that the code would need to be enhanced for our new designed RF boards. In particular, the RSSI scaling operations were not proper and had to be rewritten from scratch. More troubling is the fact that MoteTrack only supports a specific frequency channel (typically at 2.4 GHz). Typically beacon motes report the frequency channel upon which it is transmitting. When installed on our RF board, however, the MoteTrack code was designed to send no channel information at all. On the mobile ECG sensor side, whenever it receives a signal, it first checks to see if the signal was transmitted on a channel it knows about. If no channel is present it will never record the signal. This means that nothing will ever be recorded by the mobile ECG sensor and the algorithm can not run. For this reason, new frequency channel reporting must be designed in our RF platform beacon mote code manually.

We have also corrected another problem in original MoteTrack algorithm: all RSSI scaling algo-

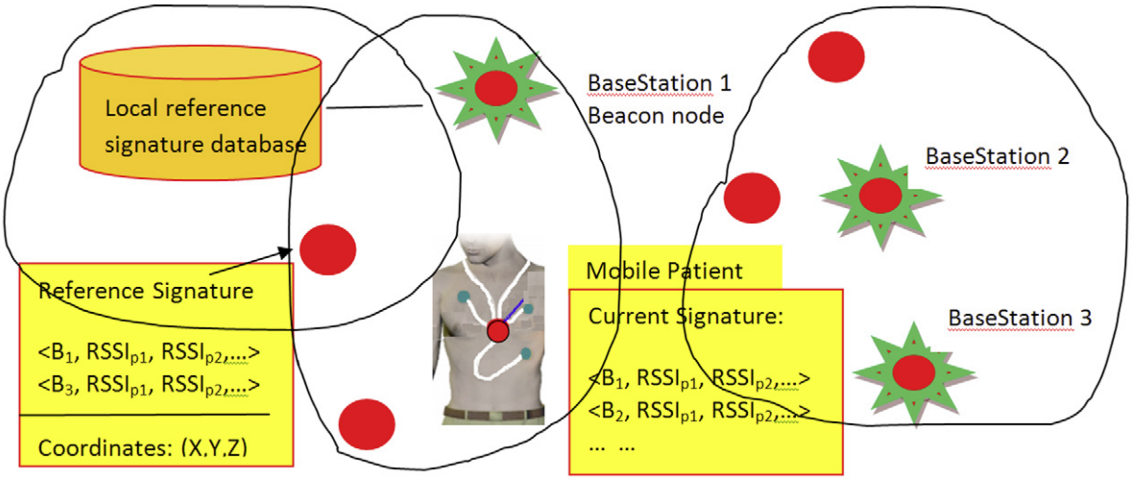


Fig. 16. Example of stored reference signatures [24].

rithms in MoteTrack depend on the detection of battery power level in voltage supplied to the mote, which means that it becomes less accurate as time passes and the batteries drain.

Once these enhancements were made, the MoteTrack program could be run in our RF board and patients' location data could be recorded. The first step towards implementing the MoteTrack algorithm is to determine the field in which tracking of a mobile mote would be desired. Beacon mote code must be altered to indicate how often each beacon must transmit their signal, which contains the beacon identification that is used by mobile motes to determine their locations. This length of time is defined by the FREQ_LISTEN_PERIOD constant and must be updated whenever the algorithm is run. This code must be compiled and loaded to the beacon motes. In the process of loading beacons, one must provide a unique identification number for each one. These beacons must then be placed throughout the environment at predetermined locations which are recorded as coordinates in a map. To make MoteTrack use this map properly the METERS_PER_PIXEL parameter must be defined. This parameter is the conversion between meters and pixels on the map. This is used by MoteTrack to place dots on the map to indicate mobile mote locations. The environment used for this experiment was the Department of Computer Engineering at Rochester Institute of Technology. See Fig. 17 for our chosen beacons and their locations.

Once the beacons have been placed, the next step is to produce a reference signature database. This is accomplished by programming a mobile mote to collect reference data and provide that data to an attached laptop computer. To program a mobile mote to collect reference signatures, the length of time that the mobile mote will record the transmissions from the beacons it can see for each reference signature point must be defined. This is the DATA_COLLECTION_PERIOD parameter. Once this is set, the mobile mote mode DATA_COLLECTION must be defined preventing the mote from operating in normal mode, and instead will forward all collected data directly to the attached computer.

This code must then be compiled and loaded onto the mobile mote. Once that has been completed, a serial forwarding program must be started to ensure proper communication between the mote and the laptop computer. Finally, the data collection program may be started. This program requires the mobile mote to be moved to various locations within the environment. At certain locations the user may indicate a position on the map, and start the data collection process for that location. This will cause the mobile mote to record power strength and identification numbers from each beacon it is within range of. These signatures are recorded as .dat files to be used later in generating the database.

A .dat file is produced for each and every location for which a signature is recorded. These .dat files must be combined into one and provided to a program provided by MoteTrack that generates two database files to be used in the main MoteTrack program's algorithms. Using these reference signatures, the algorithm can estimate location based

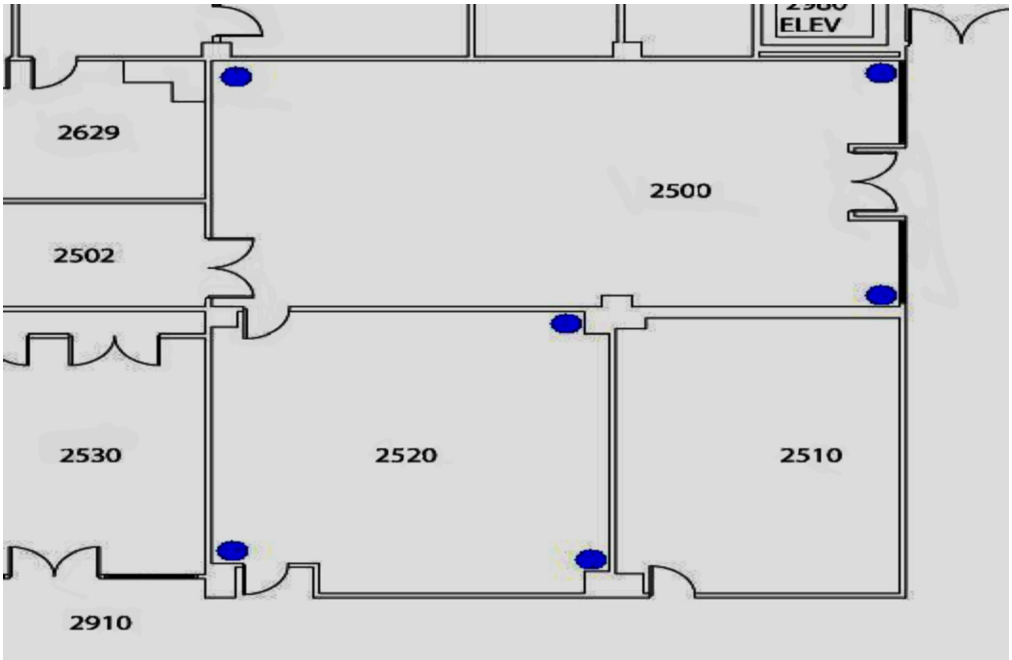


Fig. 17. Patient tracking experiment field and beacon placements (at RIT).

on the power strength and beacon identification. This data collection process prevents the use of MoteTrack in hostile environments. It is simply not feasible to deploy beacon motes at predetermined locations and create a reference signature database. The system has now been set up. It is now ready for mobile mote tracking.

4.2.2. Discussion on positioning accuracy

Compared to some other schemes, such as using RTT measurements or propagation model and triangulation theory [54–57], the advantages of the above Mote-track based positioning scheme include the following a few aspects: (1) Distributed, self-organized processing: RF-based location tracking is a well-studied problem [54]. However, existing approaches to RF-based localization are centralized (i.e., they require either a central server or the user's roaming node, such as PDA or laptop, to compute the user's location) and/or use a powered infrastructure. MoteTrack uses a decentralized approach to computing locations that runs on the programmable beacon nodes, rather than a back-end server. (2) Reliability: most previous approaches are brittle in that they do not account for lost information, such as the failure of one or more transmitters, or perturbations in RF signal propagation. As such, existing approaches are inappropriate for safety-critical medical applications. MoteTrack employs a dynamic radio signature distance metric that adapts to loss of information, partial failures of the beacon infrastructure, and perturbations in the RF signal. (3) Fault tolerance: the location signature database is replicated across the beacon nodes themselves in a fashion that minimizes per-node storage overhead and achieves high robustness to failure.

On the other hand, it also has some drawbacks. For instance, it may be prone to large change of antenna placement/orientation, and the slightest change to the beacon infrastructure requires repetition of the training procedure. However, the above shortcomings are not significant: first, all RF-based positioning schemes have variable accuracy when antenna performance changes a lot; second, the signature database has low-cost self-replication protocol such that the training procedure has low-overhead.

5. ECG data mining

Here we will discuss our data mining algorithms for the ECG data collected from RF sensors.

Feature extraction is a commonly used term in image processing and pattern recognition. It is a form of dimensionality reduction that locates points of interest from a multidimensional space. In the scope of this research, feature extraction is conducted by applying wavelet analysis techniques to patient data, thus providing ECG characteristic point detection capabilities. Since most recently published detectors are based on standard database libraries, this real-time application is an attempt to expand the horizons of current research efforts. It also offers a significant function extension to existing vital sign monitoring systems and brings them one step closer to medical care realization.

Although MATLAB has proven itself to be a very powerful instrument in both academia and industry, it does not provide command line support for its functions and libraries outside the MATLAB working environment. This is particular cumbersome for its intended applications in this research. MATLAB and its wavelet toolbox provide a good option for the desired signal wavelet analysis and feature extraction, but the patient data are passed in automatically via a Java application in a complete separate working environment. While MATLAB does provide Java Virtual Machine support, it is not possible the other way around to access MATLAB functions from a Java program outside. To maintain the real-time behavior of this application, the patient data must be passed into the MATLAB workspace promptly for signal processing.

The solution to the above problems is to setup a MATLAB server establishing a connection to the localhost that enables communication within the workstation. A number of additional files are required to make this work classified into the server side and the client side. The MATLAB server is based on a small application named MatlabControl.java developed by Kamin Whitehouse during his studies at University of California, Berkeley. This is a Java program intended to access MATLAB commands while running inside the MATLAB working environment. This is made possible by MATLAB's support for the Java Virtual Environment and the abilities to execute normal Java programs.

The MATLAB server is based on the MatlabControl file. It establishes a localhost connection and awaits communication from the outside programs. Upon receiving messages, it either redirects them to the appropriate MATLAB functions via MatlabControl.java, or responds with a predefined solution back to the awaiting clients. One of the problems that exist with running a Java program inside the MATLAB environment is the fact that MATLAB provides only one single thread, therefore the termination of any Java application initiated from inside MATLAB would also exit MATLAB as well.

The client side of program is incorporated into the Flavor RIT application by reading patient data from Flavor RIT and communicates it to the MATLAB server via the established localhost connection. However due to the continuous input of patient data from the mobile platforms, it is impossible to send all of them at the same time, especially during times when there are more than one connected mobile platform. The design choice was to only send in data associated with the currently selected in Flavor RIT for wavelet analysis after every 600 packets have been collected. This provides a meaningful mediation for data processing and data displaying. A sample extraction result is shown in Fig. 18.

To improve the ECG classification accuracy in terms of identifying different types of abnormal heart beats, we have investigated the theory of Support Vector Machine (SVM), which has been proved to be able to minimize the probability of misclassifying yet-to-be-seen patterns [16,17].

Our SVM algorithms are based on the biology signals data mining principle in [58]. The basic procedure of SVM algorithm is as follows [16,18,58]: Considering the problem of separating the set of training vectors belonging to two separate classes:

$$S = \{(x,y)|\{(x_1,y_1),(x_2,y_2)\dots(x_L,y_L)\},\\ x \in R^n,\ y \in (-1,1)\}. \tag{5}$$

The above vectors are said to be optimally separated by the hyperplane if they are separated without error and the distance between the closest vector to the hyperplane is maximal. We can then transform the input data into a higher dimensional feature space to enable linear classification. Specifically we can define an appropriate kernel function in the input space in place of the dot product in the high dimensional feature space. Next, we can formulate the dual of the convex quadratic programming problem to obtain the unique global solution for the classifier.

To apply the above SVM theory, we need to extract some dominant features from ECG data to serve as the SVM classification vectors. Wavelets analysis is well known for its feature extraction efficiency. The Wavelet Transform of a function f is a convolution product of the time series with the scaled and translated kernel, and is given by:

$$W_{S,X0} = \int_{-\infty}^{+\infty} \frac{1}{s} \cdot \Psi\left(\frac{x-x_0}{s}\right) \cdot f(x)\mathrm{d}x, \tag{6}$$

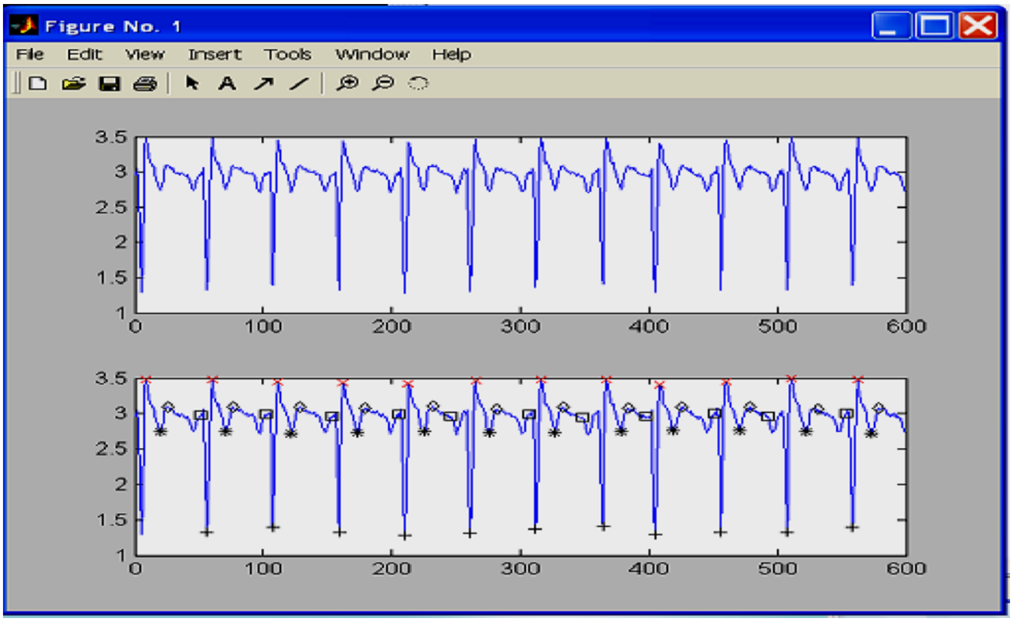


Fig. 18. Data displayed every 600 packets.

where s is a scale parameter and x_0 is a space parameter.

To find out the "features" (i.e. the singularity points) of the above wavelet function, here we introduce the concept of "Local Holder Exponent (LHE)" [16]. The LHE of a function $f()$ at the point x_0, denoted as $h(x_0)$, is defined as the largest exponent such that there exists a polynomial $P_n(x)$ of order n satisfying the following condition for a in a neighborhood of x_0:

$$|f(x) - P_n(x - x_0)| \leqslant C \cdot |x - x_0|^h. \tag{7}$$

In fact, the polynomial $P_n(x)$ is the n-order Taylor series of $f(\cdot)$ in the neighborhood of x_0. If $h(x_0) \in [n,n+1]$, then s in Eq. (2) is n times (but not $n+1$ times) differentiable at point x_0. We can thus see that $h(x_0)$ measures the "singularity" level (i.e. "irregularity") of function $f(\cdot)$ at the point x_0. A larger $h(x_0)$ indicates a better regularity. It also characterizes the local scaling range of a function. Thus we can exploit the distinct scaling behavior of different ECG signals to classify ECG time series. Since the Wavelet analysis provides a way to analyze the local behavior of a time series, the local behavior of $f(\cdot)$ is mirrored by the following wavelet characteristics:

$$W_{s,x0}(f) \propto |s|^{h(x0)}, \quad s \to 0^+. \tag{8}$$

Hence, based on the Log–Log plot of the wavelet "Amplitude vs. scale a", we can then extract the local LHE $h(x_0)$. In fact, it has been shown that Wavelets can remove polynomial trends that could cause previously used box-counting techniques to fail to quantify the local scaling of the signal [19].

Definition 1. Wavelet Transform Modulus Maxima (WTMM): To reduce the regular wavelet analysis redundancy and calculation complexity, WTMM [18] proposes to change the "continuous" sum over space (see Eq. (2)) to a "discrete" sum over the local Maxima of $|W_s, x_0(f)|$. Denote $Z(s,q)$ as a partition function, and Ω (s) as the set of all Maxima [16] at scale s, then WTMM can efficiently use the following "Space-Scale" partitioning:

$$Z(s,q) = \sum_{\Omega(s)} |W_{s,x0}(f)|^q \quad \text{, and} \quad Z(s,q) \propto s^{\tau(q)}, \tag{9}$$

where $\tau(q)$ represents a scaling range. We have the following relationship between the singularity strength $h(q)$, the spectrum of singularities (denoted as $D[h(q)]$) and $\tau(q)$: (using the Legendre Transformation theorem in [16]):

$$h(q) = \frac{\mathrm{d}\tau(q)}{\mathrm{d}q}; \quad D[h(q)] = q \cdot h(q) - \tau(q). \tag{10}$$

The importance of WTMM lies in its Maxima Lines (MLs): For any LHE $h(x_0)$, there is at least one ML that points towards x_0. For any fractal signals, the number of MLs will diverge in the limit $s \to 0+$ [16].

Although WTMM provides efficient estimation for "Global" scaling of ECG time series, it has been shown that the "Local" scaling analysis could provide more relevant information on feature extraction [20]. The idea of "Local" scaling analysis can be summarized as follows (for details, see [20]):

First, let us define a function $G(s)$ as follows (through the partition function Z(s,q), see Eq. (5)):

$$G(s) = \sqrt{Z(s,2)/Z(s,0)}. \tag{11}$$

Then the Mean LHE (denoted as $\bar{h}$) is determined by:

$$\bar{h} = \frac{\log[G(s)] - C}{\log(s)}, \tag{12}$$

where C is a constant depending on the ECG amplitude normalization ratio.

Through the Struzik Multiplicative Cascade Model [20], and using $s = 1$ in the wavelet analysis, we can estimate the LHE (denoted as $\hat{h}(x0)$) at singularity x_0 as the slope (see Eq. (9)):

$$\hat{h}(x_0, s) = \frac{\log(|W_{s,x0}(f)|) - (\bar{h} \cdot \log(s) + C)}{\log(s) - \log(s_L)}, \tag{13}$$

where S_L is the length of the entire wavelet ML (Maxima Line) Tree.

Wrapper algorithm for ECG Feature Reduction: Even though the wavelet analysis and LHE can provide us a series of ECG features, it is necessary to increase the accuracy of the induction algorithms through the reduction of parameters. Here we use Wrapper approach in [16] to conduct a search in the wavelet space. Our Wrapper algorithm [16] includes a "State" that is a vector of LHE, an initial state (we set to empty), a heuristic evaluation through fivefold cross-validation (repeated multiple times with a small penalty for every ECG feature), and a Hill-climbing search algorithm.

To validate our LHE/WTMM-based feature extraction and classification, we have used the following ECG data sets: (1) 50 Normal Sinus Rhythms (NSR) recorded from real ECG sensors; (2) Other Arrhythmia came from PhysioNet [21], which provides a set of databases that group records of one or more digitized ECG signals, as well as a set of their corresponding beat and rhythm annotations. Especially, we have used (a) PhysioNet MIT-BIH Noise Stress Test Database that contains typical noises in ambulatory ECG recordings, and (b) PhysioNet MIT-BIH Arrhythmia Database, which is used to study the different types of arrhythmias.

Regarding Arrhythmia, we have chosen the following four types: (1) Paced rhythm; (2) Atrial Fibrillation; (3) Nodal (A-V junctional) rhythm; and (4) Ventricular fibrillation. For each of the five rhythms (i..e Normal (NSR), Paced, A-Fib, Nordal, and V-Fib), we have used the following procedure (see Fig. 19) to extract the WTMM LHEs that will be used for the input vectors of Support Vector Machine model.

Please note that Step 3a (in Fig. 20) does not directly use the "single-value" Holder exponents since we have used statistical analysis based on large

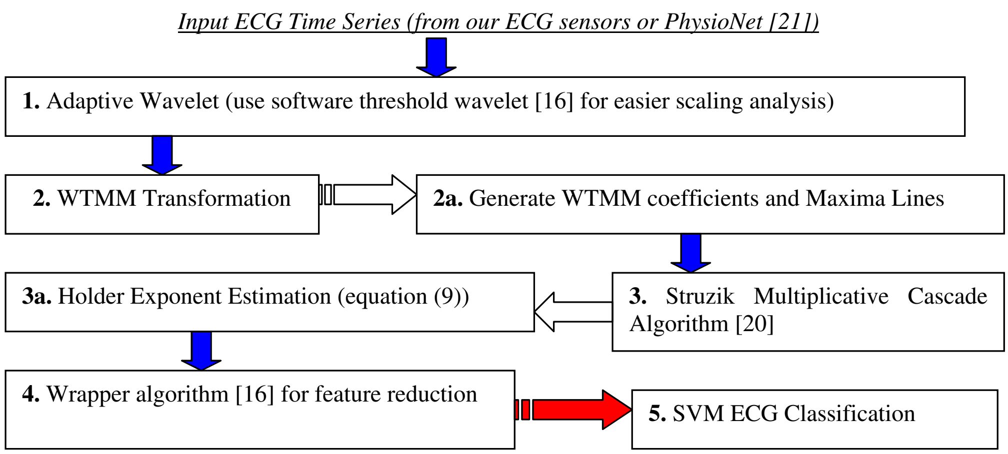


Fig. 19. ECG data series feature extraction software components.

amount of MIT-BIH arrhythmia record flows (each record flow has 10-second of ECG data series). Thus we have calculated the Probability Densities of different LHEs and then fitted those densities into a Gaussian model. The LHEs for the five rhythms were found to be in the range of (−0.5, 1.5). We then divided this range into 10 sub-ranges and took the 10 mid-points (Fig. 21) of those 10 sub-ranges in the Probability Density Function. We have used multiples runs of fivefold cross validation in Step 4.

WTMM coefficients were computed at scale [1:20] and the LHEs were estimated at scale 1. Both the leads were used for classification purpose. We can see that there are very few non-diagonal numbers present. The diagonal values represent the correct identification of the respective rhythms. Another important observation is that all the all the arrhythmia rhythms are very well separable. In the right-bottom (6×6) matrix, all the non-diagonal numbers are zero (or negligible).

$$Confuse\_Matrix = \begin{array}{cc} & \begin{array}{ccccc} NSR & Paced & A-fib & Nordal & V-fib \end{array} \\ \begin{array}{c} NSR \\ Paced \\ A-fib \\ Nordal \\ V-fib \end{array} & \begin{bmatrix} 67.3 & 0.89 & 0.13 & 1.12 & 0.89 \\ 0.77 & 19.35 & 0 & 0 & 0 \\ 1.31 & 0 & 9.98 & 0 & 0 \\ 0.91 & 0 & 0 & 20.14 & 0 \\ 1.11 & 0 & 0 & 0 & 3.41 \end{bmatrix} \end{array}. \tag{14}$$

Our SVM-based classification results are shown in Fig. 23. In that diagram, we have also compared our classification performance to two of the best ECG classification algorithms, i.e. Bayesian Classifier [23] and Decision Tree [22]. Although the accuracy for NSR is similar between ours and others, the accuracy to identify arrhythmia is higher in our scheme. More importantly, our algorithm can use WTMM/Wrapper to efficiently extract multiple features from a "large-scale" ECG database within a reasonable small calculation time.

The below equation shows the Confuse Matrix (for all Arrhythmia, not including NSR) where the

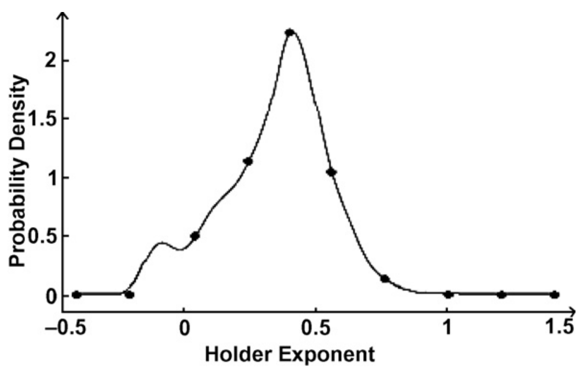


Fig. 20. Probability density–LHEs.

6. Trustworthy medical transmission

6.1. Security requirements in MANET-based tele-cardiology networks

Medical Security is important in the healthcare organizations of all over the world. For instance, US HHS issued patient privacy protections as part of the Health Insurance Portability and Accountability Act of 1996 (HIPAA) [34]. HIPAA included provisions designed to encourage electronic transactions and also required new safeguards to protect the security and confidentiality of health information. Most health insurers, pharmacies, doctors and other health care providers were required to comply with these federal standards beginning April 14, 2003 [34].

To protect the two important aspects of cardiac patient "privacy" in MASN systems, i.e. (1) confidentiality, i.e. only the source/destination can understand the medical data through crypto-keys, and (2) integrity, i.e. no data falsifying during transmission, we need to apply strong end-to-end security mechanisms to the cardiac data packets that are transmitted between any two network entities (such as between a patient's sensor and a physician's server). On the other hand, in a practical community/hospital tele-cardiology system that is based on sensor network architecture, we should consider

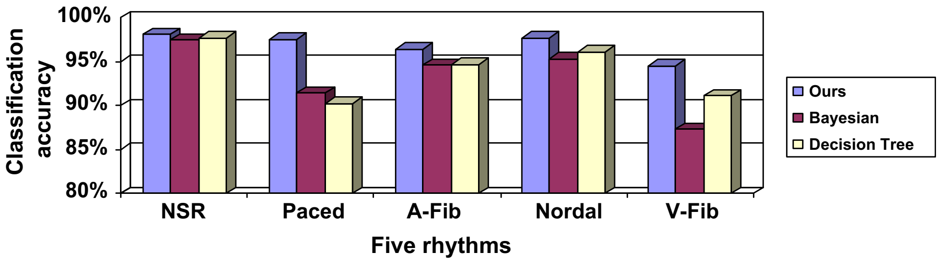


Fig. 21. Normal/Arrhythmia classification accuracy.

the following two constraints when designing privacy-preservation mechanisms:

(1) Low-energy/low-overhead security protocols: A major concern in medical security protocols design is energy-efficiency. Our experiments [9,10] have shown that most of sensor battery is consumed in radio communications instead of in ECG signal processing or sensing (see Fig. 19, a pie graph). Therefore, the security protocols should not use too many message exchanges between patients' sensors and network. Moreover the security schemes should be of low-complexity. Therefore Symmetric-crypto could be a better choice than traditional Asymmetric-crypto based on public/private keys having high computational overhead.
(2) Multi-hop vs. Single-hop security: We should use multi-hop wireless relay among patients instead of single-hop communications (i.e., direct patient-doctor wireless forwarding) due to the following reasons: First, by deploying a multi-hop data forwarding network, packets can be routed around radio obstructions in a community. While a single-hop, i.e. long-distance (>100 m), line-of-sight radio communications may not be possible. Second, Packet forwarding via multiple mall radii transmissions requires less energy than a single large radius transmission for radio communications [11,12]. The energy savings afforded by multi-hop forwarding would help conserve sensor batteries.

6.2. Security design for "one-hop" ECG data transmission

Security in each individual hop is the prerequisite of the multi-hop MASN security. As the starting point of our security research, we have implemented a low-energy, low-overhead security scheme for one-hop (e.g., patient-to-doctor) wireless communications [12,13]. As shown in Fig. 22, the security software is built in both the sensor and the Mote Interface Board (MIB) that serves as the transition gateway between a sensor and a server.

Our one-hop security mechanism uses the following two security primitives: (1) IV: Initialization vec-

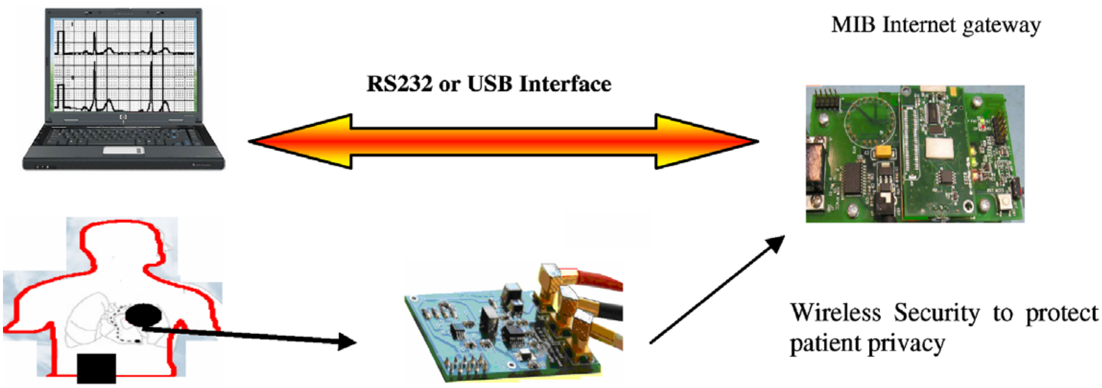


Fig. 22. Tele-cardiology MANET security: single patient case.

tors (IVs). One implication of semantic security is that encrypting the same plaintext two times should give two different ciphertexts. The main purpose of IVs is to add variation to the encryption process when there is little variation in the set of messages. (2) Block cipher choice. Triple-DES [13] is too slow for software implementation in embedded medical PDAs or sensors. We found RC5 [13] and SkipJack [14] to be most appropriate for embedded microcontrollers. Although RC5 is slightly faster, it is patented. Also, for good performance, RC5 requires the key schedule to be pre-computed, which uses 104 extra bytes of RAM per key. Because of these drawbacks, we selected Skipjack.

It is difficult to directly measure energy consumption of security mechanisms from sensors. We have thus resorted to an accurate simulator called Power Tossim [15] where hardware peripherals (such as the radio, EEPROM, LEDs, and so forth) are instrumented to obtain a trace of each device's activity during the simulation run. Through the obtained real-time traces of the current drawn in our SkipJack-based Symmetric crypto and RSA-based Symmetric crypto [13], we have computed the energy consumption of major components (such as CPU idle, CPU active, Radio, etc.) in sensors (see Table 2).

From Table 1, we can see that for the two most important components, i.e. CPU active and Radio transmission, our proposed security scheme shows significant power-saving improvements over RSA security scheme (the energy efficiency is improved by 92% and 154%, respectively).

Table 1
Security energy consumption comparisons

(in milli-joules mJ)	Skipjack	RSA
CPU active	26	51
Radio	1002	2542
Memory access	11	25
Total	1680	3360

6.3. Wireless cardiac data transmission security: "multi- patient" case

To get closer to the real tele-cardiology MANET scenario, we have extended the above single-patient transmission security to a multi-patient case.

It is challenging to securely deliver data from a remote ECG sensor to an Internet Gateway through multi-hop transmission as it requires integration of the security scheme with energy-efficient MASN routing protocols. As shown in Fig. 23, we partition patients' sensors into a number of "clusters". In each cluster, exactly one sensor is chosen as the cluster head (CH). Thus each sensor only needs one-hop communication to send the ECG signals to its CH, which searches for a neighboring CH for data relay to the Gateway. This cluster-based concept has also been used in many hierarchical routing MASN protocols to save energy [51,52]. To avoid the battery overusing in a CH, the selection of CH could be rotated periodically among the sensors belonging to the same cluster.

We have used the aforementioned SkipJack to achieve Intra-cluster Security (i.e. inside each cluster). For secure the data transmission between clusters, an Inter-Cluster Session-Key (SK) is used (see Fig. 24). A new SK is periodically distributed to all CHs by the Gateway. All new SKs are derived from a one-way hash function $H(0)$. The Gateway first pre-computes a long one-way sequence of keys: $\{SK_M, SK_M - 1, \ldots, SK_n, SK_n - 1, \ldots, SK_0\}$ (size $M \gg n$), where $SK_i = H(SK_{i+1})$. Initially only SK_n (instead of the whole M-size key sequence) is distributed to each CH. Then a CH can utilize $H(0)$ to figure out $SK_{n-1}, \ldots, SK_0$. The n keys $\{SK_n, SK_{n-1}, \ldots, SK_1\}$ are stored in a local key buffer. However, SK_0 is not in the buffer because it is used for the current data packet encryption /decryption. After the initial SK_n delivery, the Gateway periodically sends $SK_{n+1}, SK_{n+2}, \ldots, SK_M$ (one key distribution in each period) to all CHs.

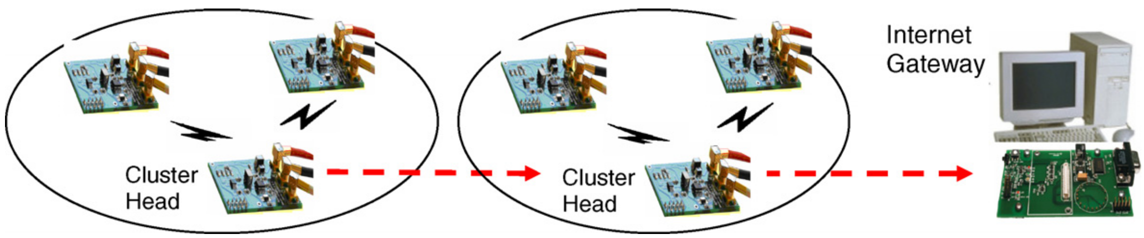


Fig. 23. Multi-patient case: cluster-based security.

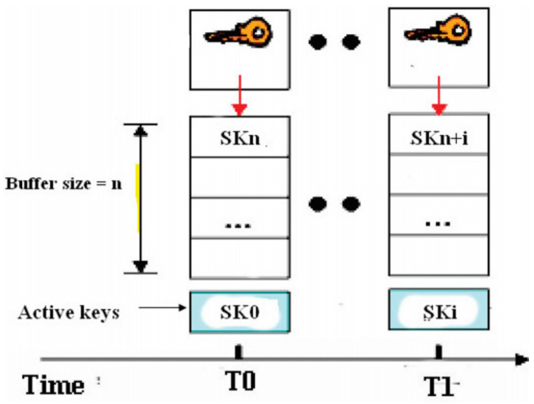


Fig. 24. Key-chain among CHs.

After receiving a new SK, the CH keeps applying $H(0)$ to it for some time, in order to find a key match in its key buffer. For instance, assume that a CH receives a new key SK_j and its key buffer already holds n SKs as follows:

$\{SK_i, SK_{i-1}, \ldots, SK_{i-n+1}\}$.

If $H(H(H\ldots(H(SK_j)))) \notin \{SK_i, SK_{i-1,} \cdots SK_{i-n+1}\}$ (15)

the authentication fails and the SK_j will be discarded. Otherwise, if the authentication is successful, the key buffer is shifted one position and the SK shifted out of the buffer is pushed into the "active key slot" to be used as the current SK (Fig. 25). The empty position is filled with a new key SK', derived from the received SK_j through H which meets the following two conditions:

$$SK\prime = H(H(H(\ldots H(SK_j)))), \quad \text{and} \quad H(SK') = SK_i. \tag{16}$$

6.4. Security analysis

(1) Gateway attacks: Because the distribution of new SKs is managed by the Gateway, it is possible for an attacker to compromise the Gateway and thus attack any future SK disclosures. Thanks to the SK buffer, there is a delay between receiving the new SK and actually using it. If the distribution interval is Δ' (i.e., the re-keying period) and n is the buffer length, the new SK will not be used until $n \times \Delta'$ later. As long as we can detect the Gateway compromise within $n \times \Delta'$ time interval and renew SKs, the cardiac data packets will maintain security performance.

(2) SK attacks among CHs: The attacker may modify the transmitting SK, inject phony SK, or use wireless channel interference to damage security packets. Our scheme can easily defeat these attacks. Thanks to the one-way characteristics of the hash function keys, any false SKs cannot pass the authentication test, that is, after L times ($L \leqslant n$) of using hash function, if we still can NOT satisfy the following formula, we will regard that it is a false SK: (in the following formula, SK_{FAKE} is a false SK and SK_{NOW} is the currently used SK.)

$$\underbrace{H(H(\ldots(H(}_{L}SK_{FAKE})\ldots))) = SK_{NOW}. \tag{17}$$

(3) Cardiac Packet attacks (such as faking the ECG data): Our scheme defeats it through SK re-keying every Δ', and inclusion of Sensor_ID and per-packet IV (which will also be updated from packet to packet) in the generation of key-streams to counter the key-stream reuse problem.

(4) Main-in-the-middle attacks: Our scheme can also defeat Main-in-the-middle attacks (where an attacker fools the CHs as if he/she were a legal CH). Our strategy is to perform a transmission of MAC in the re-keying procedure as follows:

$$Gateway \rightarrow CH: \quad E(\Delta'|n|SK_0|MAC(\Delta'|n|SK_0)). \tag{18}$$

To test our MASN security performance, we have collected the following statistics: within a certain amount of network data transmitted between all ECG sensors and a medical server, how much percentage of data belong to MASN security control messages (such as keying messages). The remaining data thus belong to other purposes (such as raw ECG values, MASN routing protocols, etc.). Fig. 25 shows our MASN security overhead collected from 1-hop away sensors (i.e. there is only one radio link between the ECG sensor and the server), 2-hop away sensors (i.e. ECG sensors need to travel 2 radio links to reach the server), and 3-hop away.

As we can see from Fig. 25, it takes higher security control overhead when the ECG data has more bytes in each communication unit (i.e. a network packet). And more hops away means higher security

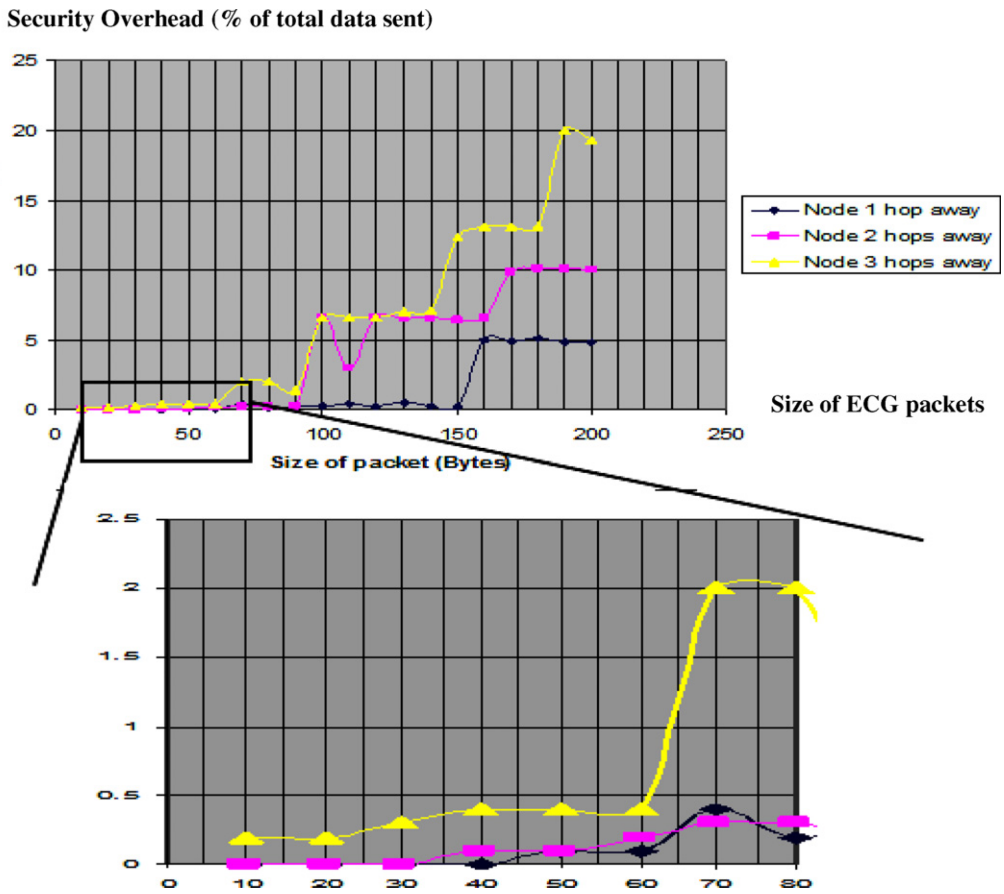


Fig. 25. MASN security overhead (percentage of total data is used for security control).

overhead since it is more difficult to control a longer distance ECG sensor.

7. System integration

After we have discussed different pieces of our MASN system in the above sections, this section will discuss the integrated MASN deployment procedure when used in practical cardiac monitoring systems. Our current MASN system includes dozens of mobile ECG sensors with RF wireless communication boards (deployed in different rooms), patient simulators (to simulate ECG data generation), one receiving station (to receive all ECG data and record them into a medical database) and the workstation (with the medical database). The patient simulator is used to generate ECG signals for testing purposes, which eliminates the inconvenience of having a live testing subject. The receiving station receives the patient and communicates with the workstation via the USB port. The Flavor RIT software picks up the ECG signal, displays it onto the screen, and then sends them via the MATLAB server for ECG feature extraction. The MATLAB program segments the data into 600-point packets and applies feature extraction algorithms to one segment at a time.

When used in practical cardiac monitoring applications, the patient simulators will be replaced by real cardiac patients who can move around (in our testbed, we hold each ECG sensor and walk around to simulate patient mobility). The current sensor-to-sensor RF communication range is around 90 m, which is good enough for a real nursing home/ hospital scenario with many cardiac patients in a large building. As we mentioned in Introduction section, those patients' ECG sensors can use hop-to-hop relay communication to reach a long distance until finally getting the data to the "receiving station" that is connected to a medical server.

Next we will exemplify some instructions on installing our MASN system software. The first step

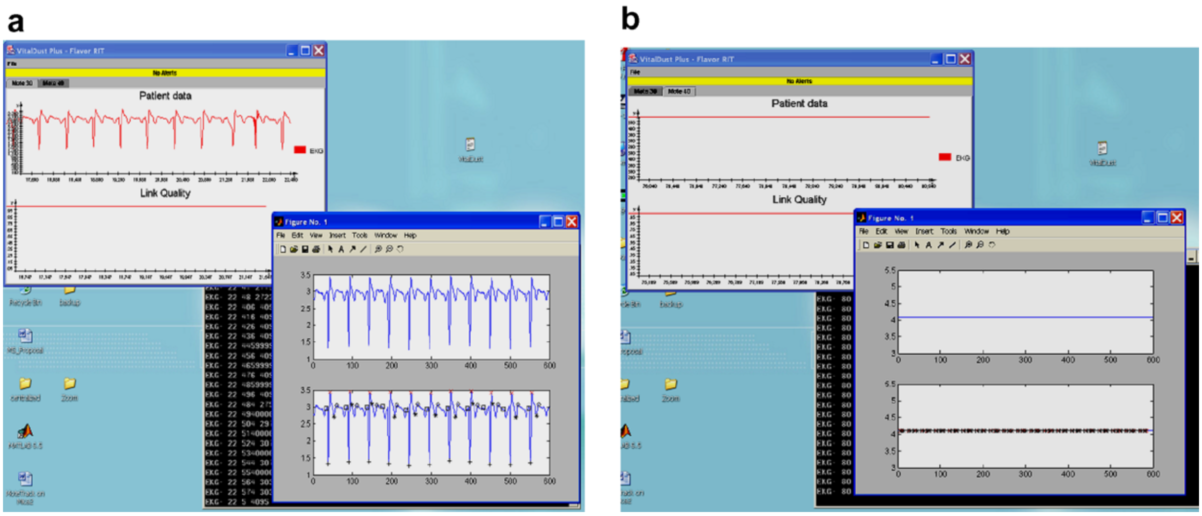


Fig. 26. Feature extraction: (a) Tab Mote30; (b) Tab Mote40.

is to start the Medical server with Matlab-based ECG processing software and database management software. Due to the using of a sensor mote as the receiving station, it is important to configure the data port for successful data transmission. The receiving mote is running the generic program TOSBase, which may be found with any TinyOS distribution. The first step is to find out the actual port number assigned to the ECG sensor board by using the command motelist. The next step is to configure the MOTECOM system variable by issuing the command export MOTECOM=serial@COM3:telos. This associates the MOTECOM variable with the serial communication port COM3 at the data rate defined by telos, which is 57,600.

After having configured the environment, it is now possible to start our Flavor RIT program by typing in vitaldust.gui.java.VitalDust in the command line, and the system would be up and running as shown in Fig. 26. Once the first 600 data points have been collected, the data would be sent out for feature extraction and a MATLAB window would appear with the associated characteristic points.

8. Conclusions

The objective of this research was to take advantage of the modern low-cost, low-power sensor and wireless communication technology, to create a telecardiology sensor network (MASN) for remote ECG monitoring purposes. Our MASN system can provide continuous vital sign monitoring capabilities without the exhaustion of any manpower. In fact, it is intended to give support to the current health care environments and free up medical professionals for more urgent functions. By automating the vital sign monitoring process, the most updated information for all patients are made available at all times. Based on wireless sensor network technology, there are the wearable mobile platforms distributed to the patients of concern. These mobile platforms are responsible for gathering patient vital sign using a 3-Lead ECG monitoring system. The gathered data are transmitted wirelessly over radio to the receiving station connected to a workstation where the data are processed. ECG Feature extraction/classification techniques are applied to the patient data and the characteristic points of interests extracted. These data provide meaningful information for the diagnosis of possible cardiovascular diseases. This is especially useful for extended recordings of ECG signals where human processing is not only time consuming, but also error prone. In addition to these functionalities, the system is designed to also provide security measurements against malicious attacks and stealing of patient information. Finally a future expansion possibility is studied for patient location tracking. This is an important expansion because the original intention of this system is to decrease the amount of time required for medical response to patients in need. By having the exactly patient locations in hand, it is possible to further reduce the response time.

In a nut shell, although there have been many research efforts in both of the fields of vital sign monitoring and ECG signal feature extraction. Most of them stay theoretical at the best. This research marks an attempt to bridge the two research fields by providing a product that is more realizable and would directly benefit the consumers in the medical field.

Acknowledgements

The authors in the Telemedicine Sensor Network Research Group at RIT sincerely appreciate the following support activities during this research: (1) NSF Cyber Trust program (CNS-#0716211 and CNS-#0716455) on telemedicine security research, (2) Sensorcon Inc. that provides three years of biochemical/medical sensor program support to us; (3) Cisco Inc. that provides University Research Program (URP) funding to us; (4) RIT Faculty EArly Development (FEAD) funding on telemedicine research. All the proposed development and concepts do not represent the opinions of the supporting sources.

References

[1] M.G. Hunink, L. Goldman, A.N. Tosteson, M.A. Mittleman, P.A. Goldman, L.W. Williams, J. Tsevat, M.C. Weinstein, The recent decline in mortality from coronary heart disease 1980–1990, The effect of secular trends in risk factors and treatment, J. Am. Med. Assoc. 277 (2007) 535–542.

[2] L.A. Short, E.H. Saindon, Telehomecare rewards and risks, Caring 17 (42) (1998) 36–40.

[3] T. Martin, E. Jovanov, D. Raskovic, Issues in wearable computing for medical monitoring applications: a case study of a wearable ECG monitoring device, in: Proc. International Symposium on Wearable Computers (ISWC), Atlanta, Georgia 2000, 43–50.

[4] Wikipedia, the free encyclopedia: on definition of MEMS technology: see <http://en.wikipedia.org/wiki/MEMS> (visited in June of 2007).

[5] CardioNet Inc. has developed an integrated technology and service—mobile cardiac outpatient telemetry (MCOT)—which enables heartbeat-by-heartbeat, ECG monitoring, analysis and response, at home or away, 24/7/365, On CardioNet project details, please see: <http://www.cardionet.com/>.

[6] R. Fensli, E. Gunnarson, O. Hejlesen, et al., A wireless ECG system for continuous event recording and communication to a clinical alarm station, in: Proc. IEEE EMBS, San Francisco, CA, USA, September 1–5, 2004, pp. 1–4.

[7] R. Fensli, E. Gunnarson, T. Gundersen, A wearable ECG-recording system for continuous arrhythmia monitoring in a wireless tele-home-care situation, in: Proc. IEEE International Symposium on Computer-Based Medical Systems, Dublin, June 23–24, 2005.

[8] T. Gao, D. Greenspan, M. Welsh, R.R. Juang, A. Alm, Vital signs monitoring and patient tracking over a wireless network, in: Proc. 27th Annual International Conf. of IEEE EMBS, Shanghai, China, September 2005.

[9] Fei Hu, Yu Wang, Hongyi Wu, Mobile telemedicine sensor networks with low-energy data query and network lifetime considerations, IEEE Trans. Mobile Comput. 5 (4) (2006) 404–417.

[10] Shruti Lakdawala (Thesis Advisor: Dr. Fei Hu), Low-power wireless sensor hardware/software platform for heart disease monitoring: an intelligent data processing interface, Master Degree Thesis, Department of Computer Engineering, Rochester Institute of Technology, Report ID: RIT-2005-309971, May 2005.

[11] L. Kleinrock, J. Silvester, Spatial reuse in multihop packet radio networks, Proc. IEEE 75 (1) (1987) 156–167.

[12] M.B. Srivastava, Energy efficiency in mobile computing and networking, Mobicom Tutorial 7 (August) (2000).

[13] Bruce Schneier, Applied Cryptography, second ed., John Wiley & Sons, 1996.

[14] SkipJack specifications: <http://jya.com/skipjack-spec.htm>.

[15] V. Shnayder et al., Simulating the power consumption of large-scale sensor network applications, in: Proc. SenSys, November 2004.

[16] Anuradha J. Joshi, Ph.D. Progress Report, Data mining of biomedical signals, Department of Computer Science and Engineering, Indian Institute of Technology, Bombay Mumbai, August 27, 2005.

[17] C. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Disc. 2 (2) (1998) 121–167.

[18] S.R. Gunn, Support vector machines for classification and regression, Technical Report, University of Southampton, May 1998.

[19] A. Arneodo, E. Bacry, J.F. Muzy, The thermodynamics of fractals revisited with wavelets, Physica A: Statist. Theor. Phys. 213 (1995) 232–275.

[20] Z.R. Struzik, Determining local singularity strengths and their spectra with the wavelet transform, Fractals 8 (2) (2000).

[21] The Research Resource for Complex Physiologic Signals, 2004, September 8. PhysioNet [Online]. Available from: <http://www.physionet.org>.

[22] R. Le Blanc, Quantitative analysis of cardiac arrhythmias, CRC: Critical Rev. Biomed. Eng. 14 (1) (1986) 1–43.

[23] T.M. Mitchell, Mach. Learn. (1997).

[24] K. Lorincz, M. Welsh, MoteTrack: a robust, decentralized approach to RF-based location tracking, in: Proc. International Workshop on Location and Context-Awareness (LoCA 2005) at Pervasive 2005, May 2005.

[25] Wikipedia, the free encyclopedia, on the definition of microsensors, see <http://www.wikipedia.com> (visited on June of 2007).

[26] M. Welsh, B. Chen, et al., CodeBlue: wireless sensor networks for medical care, Division of Engineering and Applied Sciences, Harvard University, 2006.

[27] On all wireless sensor network products (including motes, sensor boards, gateway, etc.) from Crossbow Inc., please see: <http://www.xbow.com> (visited in June 2007).

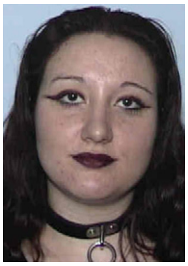
[28] On the Chipcon Inc., RF transceiver products, please see <http://www.chipcon.com> (visited in June 2007).

[29] One of the largest chip production company – Texas Instruments, see <http://www.ti.com> (visited in June 2007).

[30] On the RF and CPU chips from Ember Inc., see <http://www.ember.com> (visited in June 2007).
[31] On Zigbee Wireless Communication Standard, please see <http://www.zigbee.org> (visited in June 2007).
[32] Jason Lester Hill, System Architecture for Wireless Sensor Networks, Ph.D. dissertation, University of California, Berkeley, 2003.
[33] On TinyOS operating system, see <http://www.tinyos.net> (visited in June 2007).
[34] Office for Civil Rights – HIPAA: Medical Privacy – National Standards to Protect the Privacy of Personal Health Information, see <http://www.hhs.gov/ocr/hipaa/finalreg.html>.
[35] R.S.H. Istepanian, E. Jovanov, Y.T. Zhang, Guest Editorial: Introduction to the special section on M-health: beyond seamless mobility and global wireless health-care connectivity, IEEE Trans. Inform. Technol. Biomed. 8 (4) (2004) 405–414.
[36] E. Jovanov, J. Price, D. Raskovic, K. Kavi, T. Martin, R. Adhami, Wireless personal area networks in telemedical environment, in: Proc. 3rd International Conference on Information technology in Biomedicine, ITAB-ITIS2000, pp. 22–27.
[37] S. Park, S. Jayaraman, Enhancing the quality of life through wearable technology, IEEE Eng. Med. Biol. Mag. 22 (3) (2003) 41–48.
[38] C. Otto, J.P. Gober, R.W. McMurtrey, A. Milenkoviæ, E. Jovanov, An implementation of hierarchical signal processing on wireless sensor in TinyOS environment, in: 43rd Annual ACM Southeast Conference ACMSE 2005.
[39] D. Malan, T.R.F. Fulford-Jones, S. Welsh M, Moulton, CodeBlue: an ad hoc sensor network infrastructure for emergency medical care, in: Proc of the MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES 2004) 2004, pp. 12–14.
[40] S. Basagni, Distributed clustering algorithm for ad-hoc networks, in: Proc. International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN), 1999.
[41] S. Banerjee, S. Khuller, A clustering scheme for hierarchical control in multi-hop wireless networks, in: Proc. IEEE INFOCOM, April 2001.
[42] S. Bandyopadhyay, E. Coyle, An energy efficient hierarchical clustering algorithm for wireless sensor networks, Proc. IEEE INFOCOM 3 (2003) 1713–1723.
[43] A.D. Amis, R. Prakash, T.H.P. Vuong, D.T. Huynh, Max-min d-cluster formation in wireless ad hoc networks, in: Proc. IEEE INFOCOM, Conference on Computer Communications, March 2000.
[44] V. Kawadia, P.R. Kumar, Power control and clustering in ad hoc networks, in: Proc. IEEE INFOCOM, San Francisco, California, April 2003.
[45] C.R. Lin, M. Gerla, Adaptive clustering for mobile wireless networks, Proc. IEEE J. Select. Areas Commun. 15 (7) (1997) 1265–1275.
[46] M. Gerla, T.J. Kwon, G. Pei, On demand routing in large ad hoc wireless networks with passive clustering, in: Proc. IEEE WCNC, 2000.
[47] W.R. Heinzelman, A. Chandrakasan, H. Balakrishnan, An application-specific protocol architecture for wireless microsensor networks, Proc. IEEE Trans. Wireless Commun. 1 (4) (2002) 660–670.
[48] A.B. McDonald, T. Znati, Design and performance of a distributed dynamic clustering algorithm for ad hoc Networks, in: Proc. 34th Annual IEEE/ACM Simulation Symposium (ANSS), Seattle WA, April 2001, pp. 27–35.
[49] S. Lindsey, C.S. Raghavendra, PEGASIS: power efficient gathering in sensor information systems, in: Proc. 2002 IEEE Aerospace Conference, March 2002, pp. 1–6.
[50] O. Younis, S. Fahmy, Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach, in: Proc. IEEE INFOCOM, March 2004.
[51] A. Cerpa, D. Estrin, Ascent: adaptive self-configuring sensor network topologies, in: Proc. UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0009, May 2001.
[52] M. Demirbas, A. Arora, V. Mittal, FLOC: A fast local clustering service for wireless sensor networks, in: Proc. Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks DIWANS'04, Florence, Italy, June 29, 2004.
[53] Konrad, Lorincz, Matt, Welsh, MoteTrack: a robust, decentralized approach to RF-based location tracking, Konrad Lorincz and Matt Welsh, Springer Personal and Ubiquitous Computing, Special Issue on Location and Context-Awareness, October 2006. ISSN: 1617-4909 (Print) 1617-4917 (Online).
[54] P. Myllymaki, T. Roos, H. Tirri, P. Misikangas, J. Sievanen, A probabilistic approach to WLAN user location estimation, in: Proc. 3rd IEEE Workshop on Wireless LANs, 2001.
[55] A. Smailagic, J. Small, D.P. Siewiorek, Determining user location for context aware computing through the use of a wireless LAN infrastructure, 2000.
[56] S. Ray, D. Starobinski, A. Trachtenberg, R. Ungrangsi, Robust location detection with sensor networks, IEEE JSAC, 2004.
[57] J. Krumm, L. Williams, G. Smith, SmartMoveX on a graph—an inexpensive Active Badge Tracker, in: UbiComp 2002.
[58] Aniruddha J. Joshi, Data Mining of biomedical signals, Ph.D. Study Second Progress Report, Department of Computer Science and Engineering, Indian Institute of Technology, Bombay, Mumbai, India, August 27, 2005.
[59] Vijay K. Garg, Wireless Network Evolution: 2G to 3G, Prentice Hall PTR; first ed. (January 15, 2001) ISBN-10: 0130280771; ISBN-13:978-0130280770.
[60] Kaveh Pahlavan, Prashant Krishnamurthy, Principles of Wireless Networks: A Unified Approach, 1st ed., Prentice Hall, 2004, ISBN 8178086468.

Fei Hu received the BS and MS degrees from Shanghai Tongji University (China) in 1993 and 1996, respectively. He received the PhD degree from the Department of Electrical and Computer Engineering at Clarkson University in 2002. His PhD research was on high-performance transmission issues in wireless networks. He is currently an assistant professor in the Computer Engineering Department at RIT, New York. He served as a senior networking engineer at the Shanghai Networking Lab and Shanghai Lucent Inc. from 1996 to 1999, where he worked on several large projects on high-performance networks. He is a full Sigmaxi member, a member of the IEEE, and an IEEE chapter officer. His research interests are in ad hoc sensor networks, 3G wireless and mobile networks, and network security. His research has been supported by NSF, Cisco, Lockheed Martin, Sprint, and so on.

Mr. Meng Jiang is currently with Intel in the US. He obtained his BS and MS (duel degrees) from Computer Engineering Department at RIT (Rochester, NY, USA) in 2006. His research interests are biomedical chip design and wireless integration.



Laura Celentano is a Graduate student in the Department of Computer Engineering at RIT, USA. Their research fields are medical sensor networks and wireless security.

Yang Xiao is currently with Department of Computer Science at The University of Alabama. He was a Voting Member of IEEE 802.11 Working Group from 2001 to 2004. He is an IEEE Senior Member. Currently, he serves as Editor-in-Chief for *International Journal of Security and Networks* (*IJSN*), *International Journal of Sensor Networks* (*IJSNet*), and *International Journal of Telemedicine and Applications* (*IJTA*). His research areas are wireless networks, mobile computing, network security and telemedicine. He has published more than 190 papers in major journals (more than 50 in various IEEE Journals/magazines), referred conference proceedings, book chapters related to these research areas.