Disruption-Tolerant Wireless Sensor Networking for Biomedical Monitoring in Outdoor Conditions

Frédéric Guidec, Djamel Benferhat, Patrice Quinton

Disruption-Tolerant Wireless Sensor Networking for Biomedical Monitoring in Outdoor Conditions

Monitoring the Cardiac Activity of Marathon Runners using DTN Techniques

Frédéric Guidec · Djamel Benferhat · Patrice Quinton

Abstract Off-the-shelf wireless sensing devices open up interesting perspectives for biomedical monitoring. Yet because of their limited processing and transmission capacities most applications considered to date imply either indoor real-time data streaming, or ambulatory data recording. In this paper we investigate the possibility of using disruption-tolerant wireless sensors to monitor the biomedical parameters of athletes during outdoor sports events. We focus on a scenario we believe to be a most challenging one: the ECG monitoring of runners during a marathon race, using off-the shelf sensing devices and a limited number of base stations deployed along the marathon route. Field experiments conducted during intra-campus sports events show that such a scenario is indeed viable, although special attention must be paid to supporting episodic, low-rate transmissions between sensors carried by runners and roadside base stations.

1 Introduction

The concept of Wireless Biomedical Sensor Network (WBSN) opens up new opportunities for biomedical monitoring, such as the long-term, continuous monitoring of patients in a clinical environment or at home [1, 2,3].

In a typical deployment scenario, one or several wireless sensors are attached to a patient, and a wireless base station is installed in this patient's surroundings. This base station can either store the data received from the sensors, or it can forward these data directly to a remote site, such as a physician's desktop computer or a hospital's monitoring center. In any case, since the sensors are wireless the patient can move freely around the base station, while an endless stream of data flows from the sensors he/she is carrying to the base station. This freedom of movement is however limited by the short transmission range of the wireless sensors. Indeed, most sensors include low-power radio transceivers (such as IEEE 802.15.4/ZigBee transceivers or, less frequently, IEEE 802.15.1/Bluetooth transceivers), with which actual transmission ranges are usually between a few meters indoor and up to a hundred meters outdoor.

In traditional scenarios involving wireless biomedical sensors and a base station, it is commonly assumed that the transmission link between sensor and base station is continuously available and reliable. Transmission protocols can actually tolerate transient link disruptions without data loss, but the general assumption is that frequent, long-term disruptions should never occur while a patient's health status is being monitored. Such an assumption holds when a patient does not move much around the base station, as is the case in a hospital environment or at home. Yet there are other circumstances when the connectivity between sensor and base station can be seriously disrupted by the patient's mobility.

In this paper we investigate the possibility of using off-the-shelf wireless sensors to monitor the health of highly mobile people in outdoor conditions. Our main motivation is to confront the possibilities offered by currently available sensors with the requirements of a demanding biomedical application, in order to assess if such an application can indeed be implemented using existing devices and technologies. To achieve this goal we focus on a scenario we consider as a most challenging one: monitoring the cardiac activity of runners during

F. Guidec · D. Benferhat
IRISA - Université de Bretagne-Sud, France

P. Quinton
IRISA - École Normale Supérieure de Rennes, France

a marathon race. The underlying idea is that if biomedical monitoring can be performed in such a challenging scenario, then similar solutions can also be designed and implemented for less constrained situations.

This paper provides a synthesis of results we presented in September 2012 at the 7th International Conference on Body Area Networks (BODYNETS'12) in Oslo [4], and in December 2012 at the 6th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI'12) in Madrid [5]. Both communications addressed the same marathon scenario, which is the test-case we consider in project CoMoBioS (Communicating Mobile Biometric Sensors). This scenario is described in Section 2. Related work pertaining to biomedical monitoring and to delay/disruption tolerant networking is discussed in Section 3. Section 4 provides an overview of the sensors we use in project CoMoBioS. The constraints presented by such sensors in order to meet the requirements of the marathon scenario are discussed in Section 5. In Section 6 we report on a first field experiment we conducted in order to check whether IEEE 802.15.4 (ZigBee) technology can be used to support the transmissions between sensors worn by runners and roadside base stations. Results of this experiment were presented at BODYNET 2012. They show that 802.15.4 transmissions can hardly meet the requirements of the marathon scenario. An alternative approach was therefore proposed, and presented at UCAmI 2012.This new approach is presented in Section 7. It involves using Android smartphones as relays between ECG sensors and roadside base stations, and IEEE 802.11 (Wi-Fi) transmissions on the smartphone-to-base-station segment. Field experiments confirm that it is a lot more viable, as it allows to transmit ECG episodically with no data loss. Section 8 presents new results that were obtained using 3G transmissions instead of Wi-Fi transmissions. These results show that although 3G transmission may appear to be the "obvious" solution to collect biomedical data on marathon runners, this solution is quite power-greedy and is therefore not necessarily the most effective one. Section 9 discusses power consumption issues, and Section 10 concludes this paper

2 Description of the Marathon Scenario

The scenario we consider as a test case is defined as follows: we assume the cardiac activity of athletes must be monitored using off-the-shelf sensors featuring an ECG sensing element during a marathon race. This particular scenario was selected because runners must cover a long distance during a marathon, and that distance clearly exceeds the limited radio range of the low-power
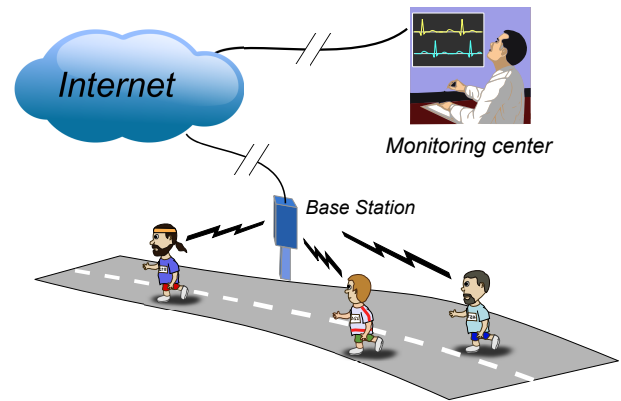
Fig. 1 Illustration of ECG monitoring for marathon runners, using wearable sensors and roadside base stations

radio transceivers available on most current sensor platforms. Besides, since runners in a marathon all follow exactly the same route, a number of base stations can be deployed along that route (see Fig. 1).

A base station (BS) is typically a unit that features a low-power radio interface, and at least one wired or wireless interface for long-distance transmissions (typically a broadband access to the Internet). The first radio interface is used to receive data from the sensors carried by marathon runners, and the second one is used to forward these data to a remote site (for example the closest medical aid station, or a physician's desktop, laptop, or smartphone). Data received from the sensors can be processed locally on the BS before being forwarded to the monitoring site, although that is not a requirement.

Transmissions between sensors carried by runners and roadside base stations must rely on low-power transceivers. IEEE 802.15.4 (ZigBee) transceivers would be ideal candidates for this purpose, since this standard has been developed specifically for low-power, low-bitrate transmission. Besides, many off-the-shelf sensors include such transceivers. Another option would be to use the IEEE 802.11 (Wi-Fi) standard, which provides higher bitrates and a longer transmission range, but at the price of a significantly higher power consumption. This option would not be our first choice, though, since current off-the-shelf sensors usually do not feature Wi-Fi interfaces.

With 802.15.4 transceivers the average transmission range is around 30 meters for outdoor transmissions. With 802.11 transceivers the average range is around 100 meters outdoor.

A BS deployed along a marathon route would thus cover about 60 meters of that route using an 802.15.4 transceiver, and about 200 meters with an 802.11 transceiver. In the first case no less than 700 base stations would be required to ensure a full coverage of the 42.2 km
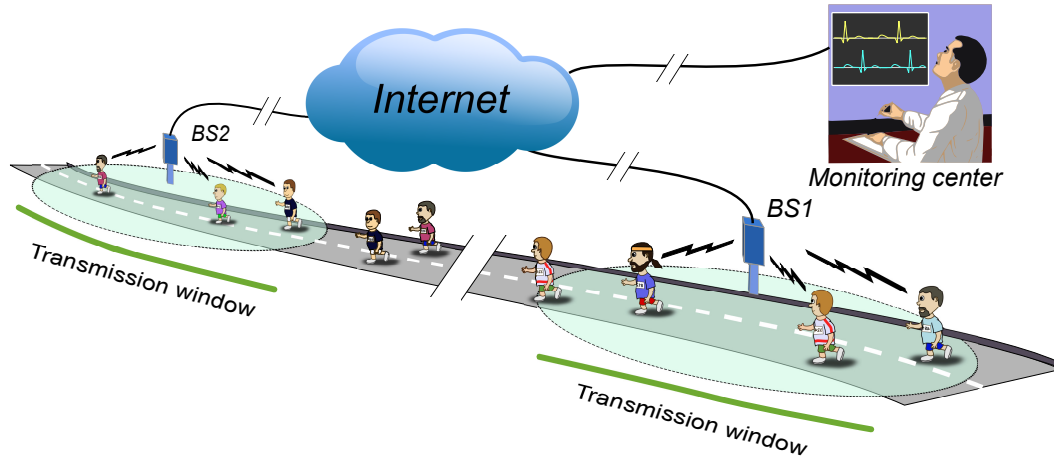
Fig. 2 Illustration of disruption-tolerant ECG monitoring for marathon runners, each base station covering only a small part of the marathon route

route, while "only" 200 base stations would be required in the second case.

In any case, deploying hundreds of base stations in order to cover a marathon route is hardly an option, for both organizational and financial reasons. The approach we propose is based on the idea that only a sparse coverage of the route needs to be ensured, using a reasonable number of base stations. A disruption-tolerant solution for data gathering must therefore be implemented, using the *store and forward* principle. This principle is the foundation of *Disruption-Tolerant Networking (DTN)*: a mobile node that is temporarily disconnected from the network can *store* data (or messages) in a local cache, carry these data for a while, and *forward* them later when circumstances permit [6]. In our scenario, the ECG sensor carried by a runner captures data continuously and stores these data locally. Whenever the runner passes by a BS, a transient radio contact occurs between the sensor and that BS. This contact is exploited by the sensor to upload data to the BS, which in turn can relay these data to the monitoring center (see Fig. 2). The distance between successive base stations and the speed of the runner determine how often "fresh" data can be sent to the monitoring center. According to cardiologists, a physician monitoring the cardiac activity of marathon runners should receive updated data for each runner at least every 5 to 10 minutes, in order to be able to detect arrhythmias and prevent incidents. Considering the pace of an average runner this implies that base stations should be placed about 1 to 2 km apart. With this approach, the marathon route can be covered satisfactorily with about 30 base stations. These base stations could typically be deployed on or near medical aid tents.

3 Related Work

As mentioned in Section 1 wireless biomedical sensor networks open up new opportunities for biomedical monitoring. Many projects have addressed the long-term, continuous monitoring of patients in a clinical environment or at home [2,3]. In the latter case the cost and inconvenience of regular visits to the physician can be avoided, or at least significantly reduced.

Projects in the mHealth (Mobile Health) line usually recognize that patients should not always be compelled to stay at home or in a clinical ward while being monitored. The solutions they propose usually rely either on dedicated base stations that must be deployed specifically for that purpose [7], or on 2.5/3G technologies [8,9,10]. In both cases the general assumption is that biomedical sensors worn by patients can send data whenever necessary to a remote site for data recording or analysis. In other words end-to-end connectivity between sensors and remote site is a prerequisite, which is sometimes hardly met in real conditions.

The concept of *Delay/Disruption-Tolerant Networking (DTN)* has been introduced as a means to cope with challenging situations where continuous end-to-end connectivity in a network cannot be guaranteed [11]. Work along that line was originally targeted at Inter-Planetary Networking (IPN), where the prime concern is to tolerate long delays and predictably-interrupted communications over long distances. Yet it was rapidly recognized that the *store and forward* principle can prove useful in many other kinds of challenging environments and application fields. Indeed transmission disruptions and delays can also be encountered in terrestrial wireless networks.

During the last decade many projects have thus been initiated in order to apply the DTN concept in

a variety of application fields, such as tactical military communication, emergency relief operations, environment and wildlife monitoring, or vehicular networking [12]. When mobile nodes are involved in the network, the *store and forward* principle can actually be extended to the *store, carry and forward* principle: mobility becomes an advantage, as it allows mobile nodes to carry messages physically (and potentially over long distances) before forwarding them to another node. This approach makes it possible to fill the gap between non-connected parts of the network, allowing remote nodes to communicate even though no temporaneous end-to-end connectivity is ever achieved between these nodes.

Delay- and disruption-tolerant solutions for sensor-based applications have already been proposed several times in the literature [13,14,15,16], but to the best of our knowledge the potential of this approach for eHealth or mHealth applications has not been investigated much so far. Yet [17] confirms that health services based on DTN techniques could notably be appreciated by health workers, especially for those working in low resource settings.

4 Overview of SHIMMER Sensors

In this project we use SHIMMER platforms with ECG expansion modules in order to acquire biomedical data on runners (see Fig. 3). The SHIMMER platform features an 8 MHz TI MSP430 micro-controller with 10 kB RAM, 16 kB EEPROM, 48 kB flash memory (for program code). Two radio modules are included, that both operate in the 2.4 GHz ISM band: an IEEE 802.15.4/ZigBee compliant CC2420 transceiver, and a WML-C46A class 2 IEEE 802.15.1 (Bluetooth) transceiver. Data acquisition is performed on up to 8 channels through a 12-bit AD converter. A 2 GB micro-SD card provides storage capacity for data logging, and the platform is powered by an integrated 250 mAh Li-Ion battery.

The SHIMMER platform is mostly dedicated to recording and transmitting physiological and kinematic data [18]. Several kinds of expansion modules are therefore available, including physiological sensors such as ECG (electrocardiography), EMG (electromyography) and GSR (galvanic skin response) sensors, as well as kinematic sensors for 3-axis angular rate sensing and 3-axis low field magnetic sensing.

The ECG expansion module we use in this project provides RA-LL (Right Arm - Left Leg) and LA-LL (Left Arm - Left Leg) input leads. The RA-LA (Right Arm - Left Arm) lead can then be calculated based on the other two leads. Sampling is performed on each
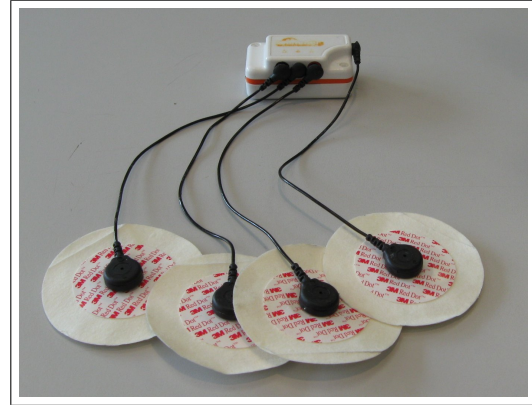
Fig. 3 A SHIMMER platform, with an ECG expansion module and electrodes

RA-LL and LA-LL channel by the 12-bit A/D converter, and the sampling frequency can be adjusted up to 1 kHz. ECG sampling on two channels therefore produces a continuous stream of data, at a rate that can reach up to 24 kbps.

Like many other off-the-shelf sensor platforms the SHIMMER platform is driven by TinyOS, a free and open-source component-based operating system targeting wireless sensor networking [19]. TinyOS applications are built in nesC (a dialect of the C language optimized for low memory consumption) out of event-based software components, some of which present hardware abstractions and others higher-level abstractions such as packet communication, routing, sensing, actuation and storage.

5 Requirements and Constraints

Compared with many other sensor-based applications that only produce data episodically, ECG monitoring is a rather demanding application. Indeed a stream of data is produced continuously, at a rate that can reach a few tens of kbps. Since our objective is to transmit ECG data in short bursts whenever a marathon runner passes by a roadside base station, the question is therefore to determine if the requirements of ECG monitoring can be balanced with the constraints of episodic, low-rate, and short-range transmissions.

In order to answer this question it is necessary to evaluate the exact requirements of ECG monitoring on the one hand, and the constraints presented by SHIMMER sensors for outdoor data transmission on the other hand.

5.1 Requirements of ECG Monitoring

ECG monitoring is usually performed with a 500 Hz to 1 kHz sampling frequency, and the SHIMMER platform's A/D converter has a 12-bit resolution. In such conditions the bitrate of the data stream produced by the platform's 2-channel ECG module ranges between 12 kbps and 24 kbps. If needed several options can be considered in order to reduce this figure:

- Using lower sampling frequency and resolution: a 200 Hz sampling with 8-bit samples (on each channel) would for example produce a 3.2 kbps data stream. Such parameters may of course alter the quality of the ECG data stream, but signal reconstruction techniques can be used on the receiver side in order to compensate for this low quality [20].
- Compressing ECG data before storage and transmission: an important constraint here is to implement an algorithm that does not exceed the computation power of the SHIMMER platform's microcontroller, such as that proposed in [21].
- Processing ECG data on the SHIMMER platform, and transmitting reports and alerts rather than the whole data stream: a recognition module for cardiac arrhythmia is proposed in [22], and delineation algorithms for the automatic detection of the major ECG characteristic waves are described in [23]. The algorithms proposed in both papers have a low computational complexity, so they can run on resource-constrained platforms such as the SHIMMER platform.

5.2 Constraints Presented by SHIMMER Platforms

The IEEE 802.15.4 transceivers included in SHIMMER platforms theoretically allow a 250 kbps transfer rate. Yet this transfer rate is the maximal signaling rate that can be achieved on the radio channel. The actual transfer rate available at application level is of course significantly lower than that signaling rate.

In order to clarify the real potential of SHIMMER platforms for data transmission in our marathon scenario, we conducted a series of preliminary experiments. We developed a simple base station by associating a netbook with a Crossbow TELOS-B mote. The TELOS-B mote is very similar to the SHIMMER platform, except that it does not come with a collection of expansion modules for biomedical monitoring. In that case the IEEE 802.15.4 transceiver of the mote was used to receive data from the SHIMMER sensor, and forward these data directly to the netbook through a USB link. It is worth mentioning that we did not investigate IEEE 802.15.1 (Bluetooth) transmissions during

these preliminary experiments, for we considered that the 10.25 second inquiry procedure required for discovering and pairing Bluetooth devices was not compliant with a scenario that involves runners passing rapidly by a base station.

- Power consumption: we observed that a SHIMMER sensor with an ECG expansion module can run for almost 10 hours on its built-in battery, while storing ECG data on the micro-SD card and sending these data continuously on the wireless channel. Further details are given in Section 9.
- Radio range: as mentioned before the average radio range around an 802.15.4 transceiver is around 30 meters.
- Transmission bitrate: according to the tests we conducted with SHIMMER and TELOS-B platforms the achievable transfer rate between sensor and BS cannot exceed 50 kbps.

The last figure of 50 kbps is surprisingly low compared to the standard's 250 kbps signaling rate. Yet this is the maximal bitrate we observed, and this result is actually consistent with other results mentioned in the literature [24] and in the TinyOS forum. Indeed it appears that the architecture of the SHIMMER and TELOS-B platforms both present a transmission bottleneck, which lies in the connection between the micro-controller and radio transceiver. Although the CC2420 radio transceiver can send and receive frames at 250 kbps on the radio channel, these frames can only be transferred to or from the micro-controller at a very limited rate. This is an important disadvantage for our marathon scenario, which requires that a single base station be able to receive data streams from several ECG sensors in the same timespan.

6 Field Experiment using IEEE 802.15.4 Transmissions

6.1 Experimental Conditions

The preliminary experiments mentioned in Section 5 gave us a crude idea of what can be expected from sensors and base stations in a marathon scenario, but we decided to get a proof-of-concept in more realistic conditions. A field experiment was conducted during an intra-campus sports event that occurred in September 2011 on the Ker Lann campus in Bruz (France). A 3.9 km running race was organized during that event, and three volunteers (two students and a professor) were equipped with ECG-enabled SHIMMER sensors on that occasion. Four base stations (BS1 to BS4) were deployed along the running route (Fig. 4). This route
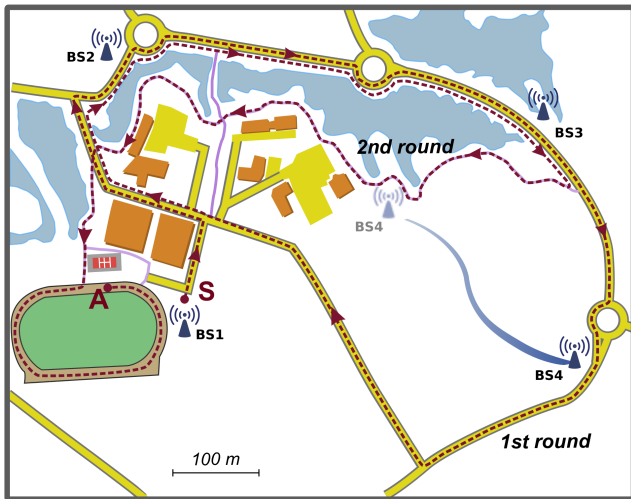
Fig. 4 Running route and location of base stations during the 3.9 km running race at Ker Lann campus

Fig. 5 One base station (i.e. netbook + TELOS-B mote) installed on the roadside during the running race at Ker Lann campus

was a loop, so the runners passed two times near each base station. BS4 had to be moved between the first and second round, since the second round was shorter than the first round. The distance between successive base stations was about 500 meters.

During this field experiment at Ker Lann campus each base station was composed of a netbook connected to a TELOS-B mote (see Fig. 5). The netbooks were not connected to a remote monitoring center on that occasion, since our motivation was primarily to observe how ECG data could be collected from the SHIMMER sensors as the runners passed close to a base station. Each base station therefore simply recorded the data obtained from passing sensors in flash memory, and the data recorded by all four base stations were reassembled and analyzed after the race was over.

6.2 Protocol for Data Acquisition and Transmission

We developed specific code in nesC in order to ensure the acquisition, storage, and transmission of data between SHIMMER sensors and TELOS-B platforms. The main features of this code are detailed below.

6.2.1 Data Acquisition on a SHIMMER Sensor

The acquisition of ECG data on each SHIMMER sensor is performed on the two 12-bit channels (RA-LL and LA-LL leads), with 500 Hz sampling frequency. The 12 kbps data stream hence produced is compressed on-the-fly, using a simple differential compression algorithm that lowers the bitrate to about 6 kbps. The data stream is then packetized in small bundles, each bundle containing a 34-byte header (including the sensor's identifier and a local timestamp), and 80 bytes of compressed ECG data. A bundle can thus fit in a single 802.15.4 data frame (whose size cannot exceed 128 bytes). After its creation a bundle is stored as a file in the SHIMMER sensor's micro-SD card, from which it can be retrieved to be transmitted during radio contacts with a base station.

6.2.2 Data Transmission between SHIMMER Sensor and Base Station

Since each base station (using a TELOS-B mote as an 802.15.4 transceiver) can have to interact with several passing SHIMMER sensors at the same time, some form of medium access control is required in order to avoid frame collisions on the radio channel. We therefore designed and implemented a simple coordination protocol, whereby a base station can allocate time slots to each sensor in range for data transmissions. This protocol is strongly inspired from the GTS (Guaranteed Time Slot) allocation method defined in the ZigBee specification [25]. Each base station periodically broadcasts a beacon frame, which allows neighbor SHIMMER sensors to detect its presence. The interval between two successive beacons is split in two parts: a Contention Access Period (CAP), and a Contention-Free Period (CFP). During the CAP sensors can notify the base station of their presence and request the allocation of a time slot for data transmission. During the CFP each sensor can use its allocated time slot to upload bundles of data to the base station, with no interference from other sensors. Since all sensors do not necessarily have the same amount of data (that is, the same number of data bundles) to upload to the base station, the number of available data bundles is included in the request a sensor sends to the base station during the CAP. The

base station can thus adjust the duration of the CAP time slots assigned to neighbor sensors proportionally to the amount of data they need to upload. Information about the allocation, ordering and duration of time slots is notified to all neighbor sensors at once, using a single frame that is broadcast by the base station at the end of the CAP and just before the CFP.

Each bundle of ECG data can fit in a single data frame, so no fragmentation is required. MAC-level data frame acknowledgement is enabled during the CFP: after sending a data bundle a sensor receives an ACK frame, that confirms that the data bundle has been received and accepted by the base station. If the ACK frame is not received the same data frame is sent again after a timeout. Upon receiving an ACK for a data frame the corresponding bundle remains in the micro-SD filesystem, but it is tagged as "transmitted" so the sensor will not try to upload this bundle again (to the same base station or to the next one).

Several strategies can be devised in order to determine which data bundles should be sent first when a SHIMMER sensor establishes a connection with a nearby base station. An option is for example to preserve the chronological ordering of data bundles, uploading the oldest bundles first. For the field experiment we decided to favor the transmission of "fresh" ECG data first, and to fill the gaps by uploading older bundles whenever possible. The transmission algorithm running on the sensors was therefore implemented in such a way that "real-time" bundles (i.e. those produced during a radio contact between sensor and base station) were uploaded to the base station first, and the time remaining during each GTS time slot was used to upload "old" bundles (i.e. bundles that had been stored on the sensor's micro-SD card, and that had not been uploaded to a base station yet). A monitoring system receiving ECG data from a marathon runner could thus display the current heart activity of the runner, and optionally allow a user to rewind the ECG stream in order to display past events.

6.2.3 Data Collection on a Base Station

Besides serving as a coordinator for wireless medium access, the base station receives ECG data bundles from passing sensors. As mentioned before, each bundle includes an identifier of the source sensor and a timestamp that is associated with the data when they are packetized. The SHIMMER platform does not include any real-time clock, so timestamping is performed based on a local timer that ticks every 100 ms. When a data bundle is sent by a sensor to a nearby base station, the
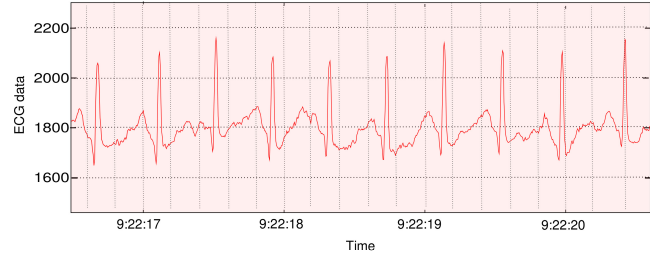
Fig. 6 Example of ECG data collected from a runner's sensor during the race

duration since this bundle was recorded is calculated, and this duration is inserted in the bundle's header in place of the record time. When the data frame is received by the base station the actual time of the bundle's production is calculated based on the current time (according to the base station's system clock), on the duration specified in the bundle's header, and on an estimation of the time required to transmit the data frame between sensor and base station (this transmission can be estimated quite accurately since data frames are transmitted during a CFP period, when no backoff mechanism is used).

Every bundle of ECG data thus received by a base station contains an indication of where and when it was produced. Each base station can therefore record data bundles for deferred analysis, or transmit these bundles to a remote site with no risk of data mixup or disordering.

6.3 Results

During this experiment the three runners covered the 3.9 kilometers in about 22 minutes, and each sensor produced about 2.5 MB of ECG data (that is, about 65.000 compressed bundles) during that time.

Figure 6 shows an excerpt of the ECG data stream that was collected from one of the sensors during the race. This data stream would probably need some noise reduction processing, but as such it is exploitable by a cardiologist.

During the race our prime motivation was to observe if the data bundles produced continuously on each sensor could actually be transmitted when the sensor established radio contact with one or another base station. Figure 7 shows the timeline of transmissions between the three sensors (S1 to S3) and the four base stations (BS-1 to BS-4). More precisely it shows the radio contacts as each sensor passed close to a base station (lines with arrowheads), and the amount of data that were uploaded to the base station during that contact. For example, a radio contact was established between
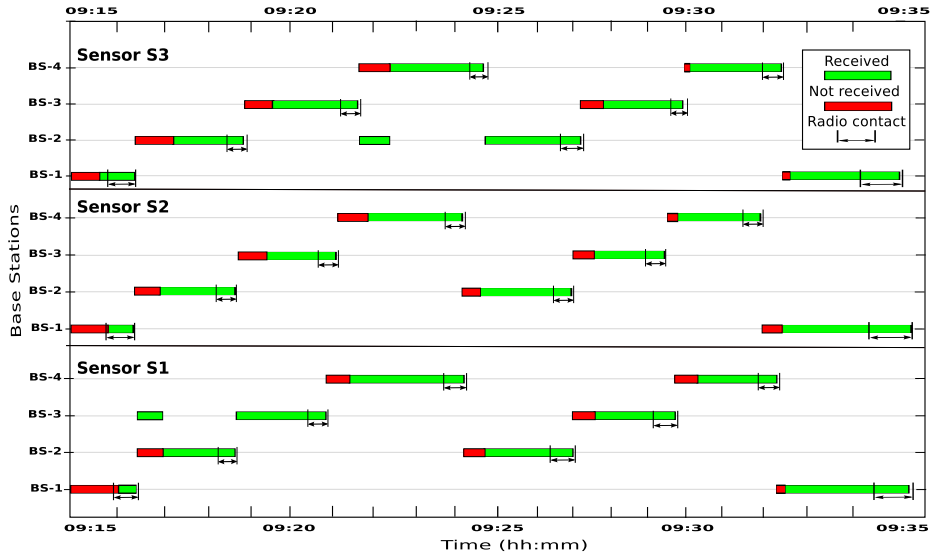
Fig. 7 Timeline of data transmissions during the first field experiment (at Ker Lann campus)

S1 and BS-1 between 09h15m55s and 09h16m22s. It can be observed that during this 27-second contact only a small amount of data was uploaded from S1 to BS-1, and none of the data acquired before the radio contact was uploaded to BS-1. About two minutes later S1 established a radio contact with BS-2, and during this contact 82% of the data produced since the former contact was uploaded to BS-2. The next contact was established between S1 and BS-3, and this time S1 managed to upload to BS-3 all the data it had produced since its contact with BS-2, plus 17% of the data that it had failed to transmit to BS-2.

By observing carefully the timing of the radio contacts between the sensors and base stations, it can be observed that the data uploading process was more effective when a base station only had to interact with one or two sensors simultaneously. In contrast, when a sensor had to deal with the three sensors (as happened at the beginning of the race when all three sensors passed close to BS-1 at the same time) only a fraction of the data could be uploaded to the BS.

During the race the duration of radio contacts ranged between 11 seconds and 48 seconds, with an average value of 19 seconds. During these contacts the sensors managed to upload 79% of their data to the base stations. The remaining 21% of data bundles were not lost, though, since they were stored on each sensor's micro-SD card and could be collected after the race.

6.4 Discussion

Globally the results of this field experiment involving 802.15.4 transmissions between sensors and base sta-

tions confirm that the protocol we implemented can indeed tolerate transient connectivity, and is resilient to connectivity disruptions. However they also show that in spite of this disruption-tolerant procotol not all data acquired during the race could be uploaded to the base stations.

A major outcome of this experiment is that it clearly showed the limits of outdoor ECG data acquisition using short-range, low-rate 802.15.4 transmissions. Although this experiment was conducted with only three sensors, and although the distance between successive base stations was rather short (about 400 meters instead of the 1 or 2 km required during a real marathon race), only a fraction of the data acquired on each sensor could be collected by base stations during the race.

Of course the amount of data produced on each sensor could certainly be reduced, as explained in Section 5, by adjusting the frequency and resolution of ECG acquisition. A more efficient compression algorithm could for example be implemented, provided the code of this algorithm could hold in the SHIMMER platform's 48 kB flash memory. Our current code (which handles data acquisition, compression, packetization, storage, and transmissions) already has a 45 kB footprint. Replacing the simple differential compression algorithm it contains by a more efficient compression algorithm without exceeding the SHIMMER's capacity would be quite a challenge. Indeed, at runtime about 22% of the processing time is used for data acquisition, 28% for data compression and packetization, 16% for data storage (writing to and reading from the micro-SD card), 28% for data transmission (which is an important task but only occurs when the sensor gets close

to a base station), and the remaining 6% is spent performing other minor tasks.

7 Field Experiment using IEEE 802.11 Transmissions

7.1 Experimental Conditions

The field experiment conducted at Ker Lann campus revealed that 802.15.4 transceivers can hardly meet the requirements of our marathon scenario. In order to get around this problem we started investigating an alternative approach whereby transmissions between a runner and nearby base stations are based on the IEEE 802.11 (Wi-Fi) standard. Unfortunately, like most biomedical sensors the SHIMMER platform does not include an 802.11 transceiver. Each runner must therefore carry a smartphone, which serves as a relay between his/her sensor and nearby base stations. Besides, since most smartphones feature an IEEE 802.15.1 (Bluetooth) transceiver but no IEEE 802.15.4 (ZigBee) transceiver, the sensor and the smartphone must be linked using Bluetooth.

In this new configuration the ECG data stream produced by the sensor is thus transmitted directly and continuously to the smartphone through a Bluetooth RFCOMM link. The smartphone processes this data stream (packetization + compression + storage + optional signal analysis), and the upload of data bundles from a smartphone to roadside base stations is performed using Wi-Fi wireless links. Base stations take the form of standard Wi-Fi access points, with broadband connectivity to the monitoring center.

7.2 Protocol for Data Acquisition and Transmission.

In order to investigate this new approach we again developed specific code in nesC for the SHIMMER sensors, and a Java application for Android smartphones. With the protocol described in Section 6.2 the SHIMMER sensor was responsible for data acquisition, compression, packetization, and episodic transmission to nearby base stations. With this new protocol the sensor is only responsible for data acquisition. All the other operations are delegated to the smartphone.

7.2.1 Data Acquisition on a SHIMMER Sensor

As mentioned above the ECG data stream produced on the sensor is transmitted on-the-fly to the smartphone through a Bluetooth RFCOMM link.

7.2.2 Transmission between SHIMMER Sensor and Android Smartphone

Each SHIMMER sensor must be paired with a specific smartphone, and two paired devices must of course be carried by the same marathon runner. The Java application we designed for Android smartphones allows a user to locate nearby SHIMMER sensors, and to pair the smartphone with one or several of these sensors, using secured pairing if desired. The possibility for a single smartphone to collect data from several sensors is a provision for future work: several sensors may thus be attached to a single athlete, so different kinds of data can be collected simultaneously.

Once a smartphone is paired with a sensor, an RFCOMM link is established between them. Through this link the smartphone can control the sensor, and send simple commands in order to adjust the sampling frequency or resolution, to start or stop the data acquisition, etc. When data acquisition is enabled on a sensor, a continuous data stream is sent to the smartphone through the RFCOMM link.

The code we designed for both SHIMMER sensors and Android smartphones can tolerate transient disruptions in RFCOMM links. For example, if paired sensor and smartphone get disconnected for a while, they strive to re-establish the connection, and data transmission (if enabled) resumes as soon as the connection is re-established.

The data stream received by the smartphone is packetized in bundles, which are then stored in the smartphone's SD-card, awaiting for transmission to the monitoring center. Each bundle consists of a header and a payload. The header includes an identifier of the source sensor and a timestamp. The payload is simply a byte array that contains a sequence of data bytes received from the sensor. The size of this byte array depends on the data acquisition frequency and resolution on the sensor, as well as on the period set for data bundling. For example, data acquisition on two 12-bit channels with 500 Hz sampling produces a continuous data stream at 12 kbps. Assuming a bundle is produced every 20 seconds on the smartphone, each bundle contains a 30 kB payload.

7.2.3 Transmission between Smartphone and Roadside Base Station

As mentioned in the former section a base station is typically a standard Wi-Fi access point with broadband connectivity to the Internet. When a runner passes close to a base station, the smartphone he/she is carrying detects the access point and tries to associate with it. If

the association succeeds, the smartphone authenticates if needed, and sends a DHCP request in order to obtain IP parameters from a DHCP server. Once the IP connectivity is obtained, the Android application we designed starts uploading data bundles to a remote server that is the entry point of the monitoring center.

The transmission of data bundles between smartphone and server is performed using UDP datagrams. Each bundle of ECG data easily fits in a single datagram, so no fragmentation is required at this level. An ARQ (Automatic Repeat reQuest) mechanism is however implemented in order to prevent any loss of data bundle.

When a smartphone carried by a runner establishes a connection with a base station these devices are still far away from each other (see Fig. 2). The quality of the wireless link between them is usually quite low, so sending data bundles too hastily in such conditions could yield a high level of data loss. In order to prevent this problem our Android application implements a rate control mechanism that is inspired from TCP's slow-start mechanism. At the beginning of a contact window –when the smartphone has just received IP parameters from the DHCP server– a simple stop-and-wait method is used: the smartphone only sends one bundle, and waits for an acknowledgement that this bundle has been received by the server. Once this acknowledgement is received the next data bundles are sent with a go-back-N method: up to N bundles can be sent in a row, before receiving the acknowledgement of the first of these bundles. The width of the sliding window (N) can be adjusted dynamically as the runner moves closer to the access point, then farther from this access point. Experience confirms that the combination of both ARQ methods allows an efficient use of the contact window between smartphone and base station, and does not induce a high level of transmission failures at the beginning of a contact window.

7.3 Results

A new field experiment was conducted on the Tohannic campus in Vannes (France) in Spring 2012, in order to observe how our system performs with 802.11 transmissions on the runner-to-base-station segment. Three volunteers were equipped with ECG-enabled SHIMMER sensors and HTC Wildfire S smartphones, and two base stations (BS1 and BS2) were placed about 1 km apart along the running route. These base station were standard Wi-Fi access points. They were both placed on a window-ledge, and connected to the campus LAN. The runners had to run around the campus, passing twice close to each base station.

Figure 8 shows the timeline of transmissions between the smartphones carried by the three runners (S1 to S3) and the two base stations. The intervals with arrowheads depict radio contacts between smartphones and base stations, and the duration of each contact is indicated in the figure.

Let us examine the transmission timeline for S1. This smartphone was installed together with a SHIMMER sensor on a runner around 10:58. Both devices were activated immediately, so S1 started collecting bundles of data from that time on. Once the three runners were ready to go, they walked together to the start line. Since BS1 was located near that line a connection S1 established a connection with B1, and started uploading to the remote server all the bundles it had recorded since its activation. At 11:04, the three runners started running. The connection between S1 and BS1 was therefore interrupted, after a 105 second contact window during which 21 data bundles had been uploaded to the server. Around 11:10 S1 established a connection with the second base station. This new contact window lasted 40 seconds, and this time 19 bundles were uploaded by S1 (17 of these bundles had been produced since S1 lost contact with BS1, and 2 new bundles were produced while S1 was in contact with BS2). As the runner carrying S1 continued running around the campus, S1 later established a connection again with BS1 (around 11:19), and then with BS2 (around 11:28), which was installed close to the finish-line.

7.4 Discussion

During this field experiment involving 802.11 transmissions between runners and base stations, no data bundle was lost, or failed to reach the remote server. This is of course a major improvement over the first experiment, which revealed the limitations of 802.15.4 transmissions.

The configuration involving smartphones that serve as relays between SHIMMER sensors and roadside base stations obviously meets the requirements of our marathon scenario. Additional experiments have been conducted in order to assess the scalability of this approach. Each of the three volunteers involved in the previous experiment has been equipped with a backpack containing 4 smartphones, and went running around the campus again. The Android application running on these smartphones was configured so as to run in simulation mode, producing dummy data bundles at the same rate as if real bundles were received from a SHIMMER sensor. With this configuration we could verify that when the three runners (hence the 12 smartphones) passed close
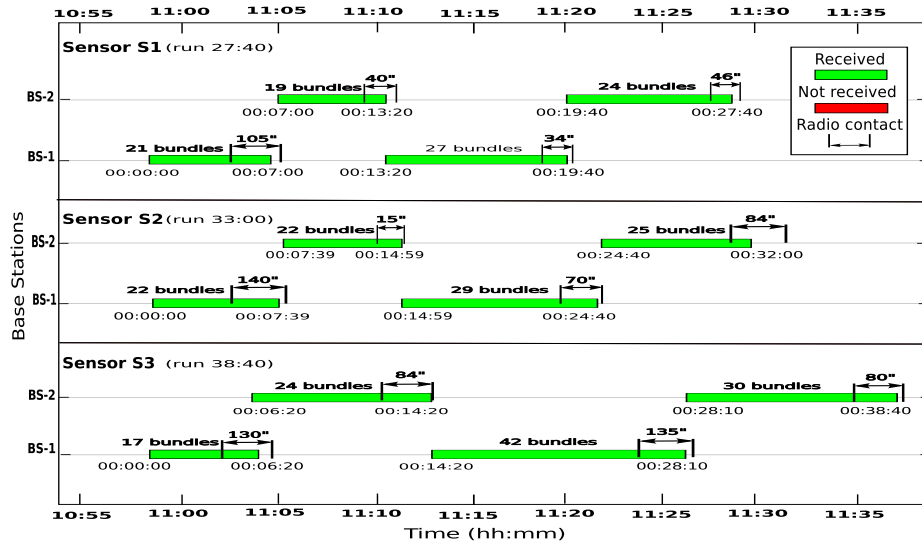
Fig. 8 Timeline of data transmissions during the second field experiment (at Tohannic campus)

to an access point simultaneously, the smartphones managed to associate with this access point and upload their data bundles during the contact window.

Further experiments involving a larger number of runners covering a longer distance should of course be conducted, but considering the high bandwidth available with Wi-Fi transmissions it can be expected that hundreds of runners can be monitored simultaneously using this approach (although hundreds of runners passing by the same access point at the same time may exceed its capacity).

One of the drawbacks of this configuration is of course that runners might be reluctant to carry a smartphone in an armband, in addition to the SHIMMER sensor. A SHIMMER unit with its ECG expansion module weighs about 22 grams. In contrast a smartphone usually weighs between 100 and 200 grams.

8 Field Experiment using 3G Transmissions

8.1 Experimental Conditions

Besides using episodic Wi-Fi transmission to upload ECG data bundles to a remote monitoring center, the Android application we designed for the smartphone can also rely on 3G transmission for data upload.

In this configuration data bundles can be uploaded as soon as they are produced by the smartphone, provided 3G connectivity is effective whenever a new bundle is ready to be sent. Otherwise bundles are stored on the bundle until connectivity is restored.
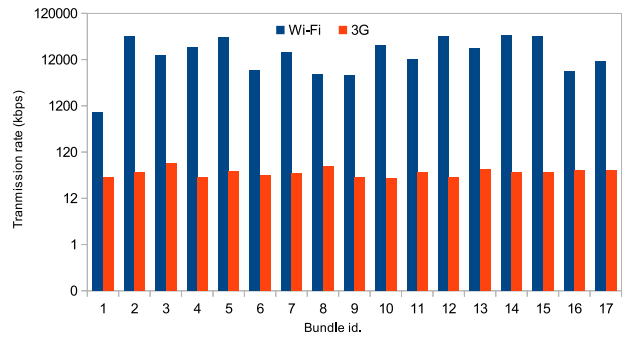
Fig. 9 Transmission rates observed with Wi-Fi and 3G for 16 kB bundles (logarithmic scale)

8.2 Results

A field experiment was again conducted on the Tohannic campus in order to compare how our system performs when using either Wi-Fi transmissions or 3G transmissions. Two volunteers were equipped with our system, which was configured so as to rely on Wi-Fi access points for the first volunteer, and on 3G transmissions for the second one.

During this experiment the volunteers ran side by side around the campus. Both monitoring systems were configured so as produce a 16 kB bundle every 20 seconds, and we measured the time required for sending each bundle and receiving the corresponding acknowledgement.

The results are presented in terms of transmission rates in Figure 9 (with a logarithmic scale). As expected we observe transmission rates of several Mbps

for Wi-Fi transmissions. More specifically, the bitrate observed with the Wi-Fi solution ranges from 850 kbps to 4 Mbps, with an average value around 2 Mbps. For 3G transmissions we expected to observe transmission rates of at least 200 or 300 kbps, but the figures we obtained are significantly lower. Indeed, the bitrate ranges from 35 kbps to 70 kbps, with an average value around 45 kbps.

8.3 Discussion

During this experimentation, the smartphone using 3G transmission maintained its connection with nearby cell-phone towers continuously (in HSPDA/3G+ mode) as its carrier ran around the campus. It is worth mentioning that this campus is well covered by cell phone networks. During a real marathon race the connectivity to a 3G network would probably be less stable, and the store-carry-and-forward algorithm implemented in the Android application would prove useful in such conditions.

Another issue is that the dependability of a 3G network can hardly be guaranteed in a mass crowd event such as a marathon race, during which hords of spectators –most of them using their cell phone intensively– usually gather along the route followed by the runners. In contrast the approach that requires the deployment of about 30 Wi-Fi access points with standard WPA authentication guarantees that these access points are used only for biomedical monitoring.
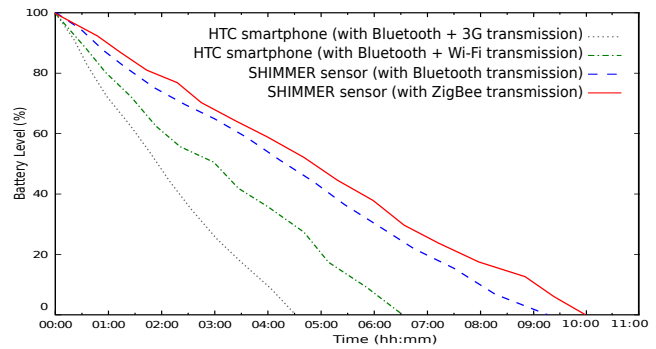
9 Power Consumption

The autonomy of the devices carried by runners might be an issue during a marathon race. An autonomy of at least 6 hours is required, so marathon runners can be equipped with a monitoring system long before they actually start running, and so the monitoring continues after they have passed the finish line of the marathon.

In order to evaluate whether this issue is a critical one we measured the evolution of the battery level on a smartphone and on a SHIMMER sensor, for each possible configuration considered in this article.

Figure 10 shows that the SHIMMER platform with its ECG expansion module can run for about 9 to 10 hours, while acquiring ECG data continuously, storing these data on the micro-SD card, and sending these data on the ZigBee or Bluetooth wireless channel.

In contrast an Android HTC Wildfire smartphone maintaining one Bluetooth connection with a SHIM-MER sensor and uploading bundles of data to a remote

Fig. 10 Power consumption observed on the SHIMMER sensor and HTC Wildfire smartphone

monitoring center can deplete its battery quite rapidly. If this smartphone relies on episodic Wi-Fi connections with nearby access points its battery is empty after about 6 hours. If this smartphone relies instead on 3G transmissions its battery is empty after about 4 hours.

An autonomy of 4 to 6 hours is barely sufficient for a marathon race, but of course other models of smartphones may run far longer in similar conditions (the Wildfire S model is known to have very little autonomy).

These results however show that a monitoring solution relying on 3G transmissions tends to be more power-greedy than a solution relying on Wi-Fi transmissions. Further experiments must be conducted in order to clarify this point. Indeed, the amount of power consumed for 3G transmission changes dynamically, as the transceiver continuously adjusts its transmission power and protocols in order to reach the base station it is associated with.

10 Conclusion

Off-the-shelf wireless sensing devices such as the SHIM-MER platform open a wide range of perspectives for health monitoring. Yet because of the limited computation and transmission capacities of such platforms most applications considered to date imply either ambulatory data recording or real-time data streaming. In the latter case, ubiquitous continuous end-to-end connectivity is expected to support data transmissions between sensors worn by patients and a remote monitoring center. With disruption-tolerant networking another approach can be considered, whereby data are captured and stored continuously on the sensor platform, and transient connectivity with one or several base stations is used opportunistically to upload data to a remote site.

In order to illustrate this approach we investigated a challenging scenario: the ECG monitoring of runners during a marathon race. A field experiment conducted during a campus sports event, using SHIMMER platforms for data acquisition and IEEE 802.15.4 transmissions to upload episodically ECG data to roadside base stations, has revealed that 802.15.4 transmissions –though appealing at first glance– can hardly meet the requirements of the marathon scenario. An alternative approach has then been considered, using Android smartphones as relays between ECG sensors and roadside base stations. With this approach data acquired by the SHIMMER sensor are transmitted continuously to the smartphone through a Bluetooth RFCOMM link. The smartphone processes this data stream, and uploads data bundles episodically to Wi-Fi access points placed on the roadside. A field experiment conducted with this configuration confirms that capturing and transmitting ECG data during a running race is indeed feasible with off-the-shelf devices and technologies. Since many current mHealth projects rely on 3G transmissions for data collection, we compared how our system performs when using either Wi-Fi or 3G transmissions for data upload. The solution involving 3G transmission proves more power-greedy, which may be a problem since our marathon monitoring system should be able to run continuously for about 6 hours in order not to be a trouble for runners. The 3G option may additionally be less scalable during a marathon event, as thousands of people use their cell phones simultaneously and tend to saturate the neigbouring cell towers. Further investigation is needed in order to clarify this point.

In the near future we plan to deploy our system during a real marathon race (possibly the next edition of the Baie du Mont Saint-Michel marathon, in France), with dozens of runners carrying ECG monitoring systems.

Acknowledgements

References

1. Chen, M., Gonzalez, S., Vasilakos, A., Cao, H., Leung, V. C. M.: Body Area Networks: a Survey. Journal of Mobile Networks and Applications **16**(2), ACM/Springer (April 2011) 171–193

2. Alemdar, H., Ersoy, C.: Wireless Sensor Networks for Healthcare: a Survey. Computer Networks **54**(15) (2010) 2688–2710

3. Konstantas, D., Herzog, R.: Continuous Monitoring of Vital Constants for Mobile Users: the MobiHealth Approach. In: 25th Annual International Conference of the IEEE EMBS. (2003) 3728–3731

4. Benferhat, D., Guidec, F., Quinton, P.: Disruption-Tolerant Wireless Sensor Networking for Biomedical Monitoring in Outdoor Conditions. In: 7th International Conference on Body Area Networks (BODYNETS'12), Oslo, Norway, ACM Digital Library (September 2012) 1–7

5. Benferhat, D., Guidec, F., Quinton, P.: Cardiac Monitoring of Marathon Runners using Disruption-Tolerant Wireless Sensors. In: 6th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI'12). LNCS, Vitoria-Gasteiz, Spain, Springer (December 2012) 1–8

6. Fall, K.: Messaging in Difficult Environments. Technical report, Intel Research Berkeley (2004)

7. Babovic, Z., Crnjin, A., Racocevic, G., Stankovic, M., Peric, Z., Cirkovic, I., Damjanovic, I., Milutinovic, V.: ProSense Research Activities in Belgrad In: 9th International Conference on Telecommunication in Modern Satellite, Cable, and Broadcasting Services (TELSIKS'09). IEEE (2009) 291–294

8. Konstantas, D., Jones, V., Herzog, R.: MobiHealth innovative 2.5-3G mobile services and applications for healthcare. In: Proceedings of the Eleventh Information Society Technologies (IST) Mobile and Wireless Telecom. (2002) 43–52

9. Tsiknakis, M., Katehakis, D.G., Orphanoudakis, S.C.: A Health Information Infrastructure Enabling Secure Access to the Life-Long Multimedia Electronic Health Record. In: CARS. (2004) 289–294

10. Klompmaker, F., Workowski, A., Thronicke, W., Ostermair, F., Willemsen, D., Hoffmann, J.D.: User Centered Design of an Interactive Mobile Assistance and Supervision System for Rehabilitation Purposes. In: 3rd International Conference on Wireless Mobile Communication and Healthcare (MobiHealth'12), Paris, France, Springer (November 2012) 1–10

11. Fall, K.: A Delay-Tolerant Network Architecture for Challenged Internets. In: Proceedings of ACM SIGCOMM03. (August 2003)

12. Voyiatzis, A.G.: A Survey of Delay- and Disruption-Tolerant Networking Applications . Journal of Internet Engineering **5**(1) (June 2012) 331–344

13. Pisztor, B., Musolesi, M., Mascolo, C.: Opportunistic Mobile Sensor Data Collection with SCAR. In: In Proc. IEEE Int'l Conf. on Mobile Adhoc and Sensor Systems (MASS07), IEEE Press (2007) 1–22

14. Jain, S., Shah, R., Brunette, W., Borriello, G., Roy, S.: Exploiting Mobility for Energy Efficient Data Collection in Wireless Sensor Networks. MONET **11**(3) (2006) 327–339

15. Wang, Y., Wu, H., Lin, F., Tzeng, N.F.: Cross-Layer Protocol Design and Optimization for Delay/Fault-Tolerant Mobile Sensor Networks (DFT-MSN's). Journal on Selected Areas in Communications **26**(5) (June 2008) 809–819

16. Nayebi, A., Sarbazi-Azad, H., Karlsson, G.: Routing, Data Gathering, and Neighbor Discovery in Delay-Tolerant Wireless Sensor Networks. In: 23rd IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2009, Rome, Italy, May 23-29, 2009, IEEE CS (2009) 1–6

17. Syed-Abdul, S., Scholl, J., Lee, P., Jian, W.S., Liou, D.M., Li, Y.C.: Study on the Potential for Delay Tolerant Networks by Health Workers in Low Resource Settings. Computer Methods and Programs in Biomedicine **107**(3) (September 2012) 557–564

18. Burns, A., Greene, B., McGrath, M., O'Shea, T., Kuris, B., Ayer, S., Stroiescu, F., Cionca, V.: SHIMMER : A Wireless

Sensor Platform for Noninvasive Biomedical Research. IEEE Sensors Journal **10**(9) (2010) 1527–1534

19. Levis, P., Madden, S., Polastre, J., Szewczyk, R., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., Culler, D.: TinyOS: an Operating System for Sensor Networks. Ambient Intelligence, Springer (2005) 115–148

20. Singh, B.N., Tiwari, A.K.: Optimal Selection of Wavelet Basis Function Applied to ECG Signal Denoising. Digital Signal Processing **16**(3) (2006) 275–287

21. Hossein, M., Nadia, K., Pierre, V.: Real-Time Compressed Sensing-Based Electrocardiogram Compression on Energy Constrained Wireless Body Sensors. In: IEEE International Symposium on Circuits and Systems (ISCAS2011), IEEE CS (2011) 1–4

22. Madzarov, G., Dordevic, D.: Heartbeat Tracking Application for Mobile Devices - Arrhythmia Recognition Module. In: 32nd International Conference on Information Technology Interfaces (ITI2010), IEEE CS (2010) 585–590

23. Boichat, N., Khaled, N., Rincn, F.J., Atienza, D.: Wavelet-Based ECG Delineation on a Wearable Embedded Sensor Platform. In: 6th International Workshop on Wearable and Implantable Body Sensor Networks (BSN09), IEEE CS (2009) 256–261

24. Farshchi, S., Nuyujukian, P.H., Pesterev, A., Mody, I., Judy, J.W.: A TinyOS-Based Wireless Neural Sensing, Archiving, and Hosting System. In: 2nd International IEEE EMBS Conference on Neural Engineering, IEEE CS (2005) 671–674

25. IEEE802.15.4-2006: Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). IEEE Standard