

Mobile Solution for Three-tier Biofeedback Data Acquisition and Processing

Orlando R. E. Pereira

Department of Informatics
University of Beira Interior,
Covilhã, Portugal
opereira@ubi.pt

Paulo A. C. S. Neves

Institute of Telecommunications /
DI, Univ. of Beira Interior, PT
EST, Polytec. Inst. of Castelo Branco
Castelo Branco, Portugal
pneves@est.ipcb.pt

Joel J. P. C. Rodrigues

Institute of Telecommunications /
Department of Informatics
University of Beira Interior,
Covilhã, Portugal
joel@ubi.pt

Abstract— Wireless sensor networks span from military applications into everyday life. Body sensor networks greatly benefit from wireless sensor networks to answer the biofeedback challenges in healthcare applications. In such applications, data is of fundamental importance, it must be reliable and within easy reach. However, most solutions rely on a personal computer to process and display sensor data. In this paper we propose a mobile solution that draws on three-tier body sensor networks to dramatically improve data accessibility, through the use of a Java and Bluetooth-enabled mobile phone. The mobile tool features data monitoring and presentation. This approach allows data visualization by the patient or medical staff without a portable computer or specific monitoring hardware. We hope to contribute to the adoption of biofeedback for early detection of health abnormalities and lower the budget that governments spend each year in healthcare.

Keywords— *Body Sensor Networks; Sensor Networks; Healthcare applications of Sensor Networks; m-Health; Biofeedback*

I. INTRODUCTION

A wireless sensor network (WSN) is composed of up to thousands of smart sensing nodes. These nodes have a processing unit with memory, a wireless transceiver, sensing electronics (sensor and analog-to-digital converter), and a power supply (battery).

Born on military applications, wireless sensor networks span over to different application fields, namely with the availability of hardware at affordable prices. These networks pose significant challenges over the traditional computer ones, like limited energy, which limits network lifetime.

The application of WSNs to health care becomes more than a vision, since technology is available today [1]. A sensor network that senses health parameters becomes a body sensor network (BSN) [2-5]. Nodes of BSNs are directly attached to human body, so special care must be exercised. In certain scenarios a BSN is expected to continuously operate for several days without user intervention, gathering patient data, so energy constraints apply to the smart sensor node [6]. Since typically smart sensor's redundancy is not used, a sensor that stops working results in a lost health parameter [7, 8]. One of the approaches is to defer data aggregation and acquisition to a

more powerful node, called sink node, instead of relying on the smart sensor nodes to process raw data.

One of the goals of a BSN is to provide biofeedback, the capacity to continuously monitor health parameters such as heart beat rate, arterial blood pressure [9], and body temperature in an unobtrusive and efficient way [10], thus monitoring health parameters in behalf of the patient.

This paper focuses on one of the design decisions on the deployment of a BSN: data processing and presentation. A BSN without a mean to extract data is basically useless. We rely on a three-tier BSN, with smart sensor nodes, a sink node and a presentation device to provide an integrated solution for biofeedback.

The remainder of the paper is organized as follows. Section II provides some insight on related work, while section III presents the global system architecture and the importance of the mobile solution. Section IV elaborates on the mobile solution, while section V describes the mobile tool user's interface. The deployment and validation is left for section VI. Finally, section VII concludes the paper.

II. RELATED WORK

In this section we surface some related approaches to biofeedback and compare them to our proposal. In [7], authors present a phone-centered body sensor network. This approach uses a mobile phone as the sink for the BSN, and a wrist device as the user interface hardware. This approach relies on a mobile phone to take care of sensor data acquisition, which may not be adequate given the sensitive nature of data and reliability of the phone itself for such purposes.

A Wireless body sensor network (WBSN) for healthcare monitoring is presented in [11], where a sink and a central system is used. The sink acts as the "middleman" between the WBSN and the central system. The central system is a standard personal computer and is intended for hospital facilities rather than for personal use.

Authors of [12] present an interesting approach to healthcare monitoring using a wearable wrist device. However, it does not constitute a BSN since a single device is used with multiple sensors (light, motion, audio and

temperature) and Bluetooth connection technology to a personal computer or mobile phone. E-watch features a LCD to view some limited information (128x64 pix monochrome) with a graphic view (although limited by the screen resolution), and some added functionalities like a calendar.

Another solution to the growing problem of older adults care comes from [13]. The authors present the telecare concept and provide some information about a pilot study. The need for thresholds to achieve flexibility and support a broad range of individuals is also mentioned.

Several approaches exist with different design goals, but the same purpose: to provide health information to the patient, to the medical staff or for both. In this work we aim to provide a convenient mean to sense bio-signals, process it and show health information. Since nowadays almost everyone carries around a mobile phone, the choice seems obvious. Moreover, the large majority of mobile phones today feature Bluetooth connection and Java support, the necessary conditions to run the mobile tool.

III. SYSTEM ARCHITECTURE

This section presents the system architecture based on a three-tier Body Sensor Network (BSN). Figure 1 shows a scenario that illustrates the global system architecture. A sink node aggregates the values of the different sensors, on a per-channel basis. It can be viewed as a "middleman" between the sensor network and the outside world, providing constant monitoring of health parameters.

The mobile device connects to the sink via Bluetooth, providing a very convenient and broad range of devices for data processing and visualization.

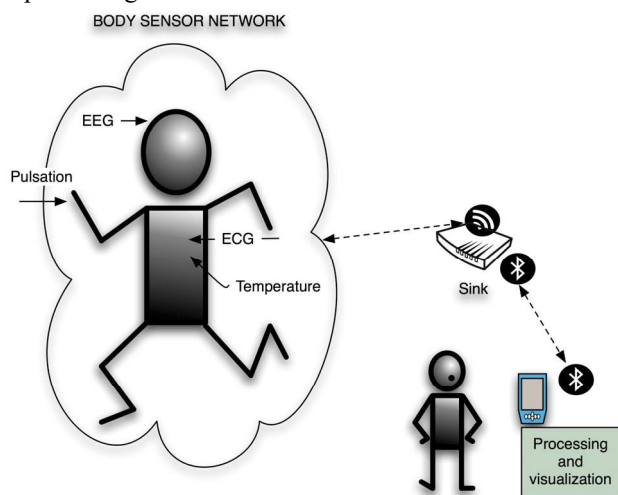


Fig. 1. Illustration of the global system architecture.

The proposed three-tier architecture allows separation of system functionalities. The BSN is responsible for health parameter's sensing, the sink is responsible for data aggregation from all sensors, fault tolerance, and outside communication, while the mobile device is responsible for data processing, calibration and presentation. As a result, if the

mobile device malfunctions or is not present, data is not compromised. Also, the mobile tool is abstracted from the BSN technologies involved, being it wired communication, wireless 2,4GHz radios or other.

With this three-tier architecture system, development, testing and validation are easier to accomplish. The BSN is fire walled from the outside world, and can be deployed independently of the processing and visualization device. In the BSN prototype we used, the sink is very portable and can be placed on the patient's belt for instance, resulting in a true portable solution.

IV. MOBILE TOOL

The mobile tool is the main focus of this paper. This tool directly communicates with the sink of the BSN through a Bluetooth connection. The sink has a unique password that is hard coded, without this password it is not possible to start Bluetooth communication.

Robustness, convenience and user-friendly interface were the main concerns for this application. Since medical data is very sensitive and vital signs are critical to human life and behavior, the application must be robust. The tool must perform robust Bluetooth connection to the sink device, thus checking for errors constantly, with a very tight control over the communication.

The use of a mobile device brings significant convenience advantages over a portable personal computer, due to the inherently form-factor of both systems. Moreover, nowadays, almost everyone has a mobile phone or a Personal Digital Assistant (PDA). However, convenience must also be addressed in the software. After an initial calibration due to individual threshold parameters that are unique, the application runs by itself, continuously collecting data. Furthermore, the same mobile device can be applied to gather different patient's data.

In terms of user-friendliness the display size and orientation, together with a mobile phone keypad are the main limitations here. The vertical instead of horizontal display dictates a guided user interface (GUI) rather different from personal computer programming. The tool adapts itself to different screen sizes, scaling font size (in three sizes – small, medium and large) and graphical scale (based on screen size and text occupation). To achieve platform compatibility, no proprietary software development kit (SDK) is used and keypad use is limited to numerical and directional keys only (keys present on every mobile phone).

The mobile tool is divided into two main functional blocks. The first one is responsible for the Bluetooth communication with the sink node, featuring connection establishment and management. The second one deals with data processing and presentation to the user.

To increase the robustness of the mobile tool two kinds of exceptions can be thrown in execution time: system exceptions (the ones that the system must handle) and user

exceptions (the ones that the user can help overcome or is simply notified of). In Figure 2 we present a diagram with the basic blocks of the tool, with indication of the possible exceptions that can be thrown.

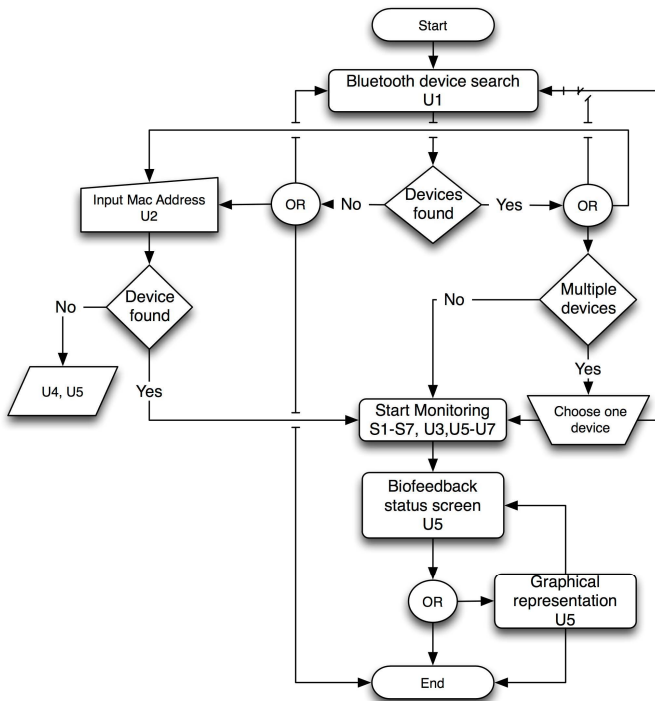


Fig. 2. Mobile application software diagram with possible thrown exceptions.

On the system exceptions, exception *S1* occurs when the device is not paired (a condition necessary to use Bluetooth connection), while exception *S2* occurs when the sink is already in acquisition mode, transmitting data to the mobile device. System exception 3 (*S3*) occurs when the mobile device tries to receive data and the sink's not yet in acquisition mode, *S4* is thrown when an attempt to open a already opened communication port is issued, and exception 5 (*S5*) occurs when communication is tried with the port closed. When a communication is in progress and the application tries to close the port exception *S6* is thrown, and finally, *S7* is thrown when the communication parameters are not correctly set.

On the user exceptions, four are related to the Bluetooth connection, *U1* when the Bluetooth adapter cannot be found, *U2* when the sink MAC address cannot be found, *U3* when Bluetooth passkey is not correct, and *U4* when no sink device could be found. Exception *U5* occurs when Bluetooth connection is interrupted, or sink battery is low/exhausted or even sink has been shut down. Exception *U6* is thrown when the firmware of the sink device is not supported under the current mobile tool implementation and finally exception *U7* occurs when the sink port cannot be opened.

The mobile tool starts with the sink device search. The search results in three possible occurrences: a single device is found, multiple devices are found, or no device is found.

Monitoring can be started on any of the found devices, but a single sink device (BSN) can be monitored at a given time. In

our current implementation, data flows in frames with support up to eight different data sources (channels). By default, the mobile tool captures continuously a specified number of frames, for instance 500.

In terms of data visualization, two types of screens were defined: a Biofeedback status screen (by default) and a graphical representation (similar to a oscilloscope of the monitored signal).

The tool uses Connected Limited Device Configuration (CLDC) version 1.1 and Mobile Information Device Profile (MIDP) version 2.0. The CLDC presents a stripped version of J2SE (Java 2 Standard Edition) focused on devices with limited computational resources such as mobile phones, PDAs and pagers, among others. MIDP supplies a specific framework for development on embedded systems, such as mobile phones and PDAs, augmenting the CLDC functionality.

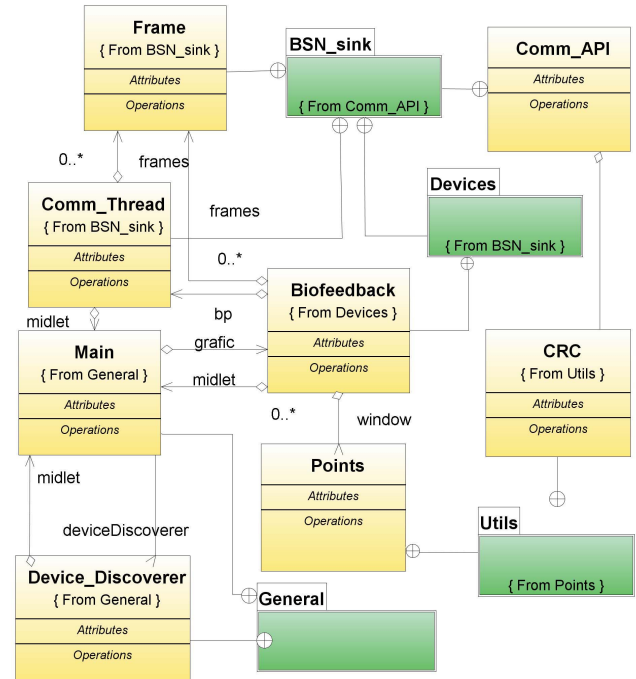


Fig. 3. Mobile tool class diagram (simplified).

Figure 3 presents the simplified class diagram of the mobile tool, excluding the exception handling classes for the sake of simplicity. Application execution starts at the *Main* class with the midlet generation, and goes into the *Device Discoverer* class that is responsible for Bluetooth device search. After the selection of a single device to communicate, the communication must be established. Class *Biofeedback* allocates space for data through the *Frame* class and connection to the *BSN_sink*. If successful, data can be received from the sink. Data is received in a cyclic form: read the next frame; perform Cyclic Redundant Checks (CRC) on frame's data, and present data on the display. Meanwhile, class *Comm_Thread* is instantiated every time new frames are needed. Class *Comm_Thread* is instantiated every time a new

sequence of frames starts. The number of frames is user-programmable. Finally, class *Points* is instantiated when a virtual window for graphical calculation is needed.

We believe that this tool presents enough functionality for everyday use. Nevertheless, sometimes the collected data must be stored and further analyzed afterwards. In order to support this feature, the mobile tool includes persistent storage of collected data in a file inside the mobile device. The file type, in order to be easily imported to spreadsheet programs, is a text type with coma-separated values. The filename reflects the date and time of the first data sample, thus avoiding file overwrite problems, and at the same time being self-explanatory.

V. USER INTERFACE

In terms of the mobile tool functionality, Figure 4 illustrates a screenshot of the instantaneous signal value of one sensor channel, with two areas: a text information area and a graphical signal representation.

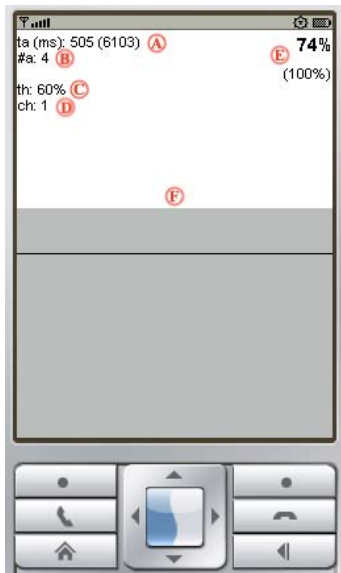


Fig. 4. Mobile tool screenshot of data presentation (text form).

Following the figure, we have in **A** the amount of time the signal overcomes the specified threshold and, in parenthesis, the maximum continuous time that the signal overcome the threshold. Next, **B** represents the number of times the signal overcome the threshold, and **C** the percentage of the current specified threshold. The number shown in **D** shows the current channel (sensor) that is being read and, in **E**, the current signal percentage and, in parenthesis, the maximum value obtained. Finally, **F** shows a graphical representation of the instantaneous signal value (use of a shaded box: the greater the box, the greater the signal) and the threshold line (in the figure, the signal overcomes the threshold signal).

Figure 5 shows an example of the second type of screen that

the mobile tool offers. The great majority of the area is occupied by a graphical, over time, representation of the sampled signal. Inside the graphic, **A** shows the maximum value the signal can take, and **B** the minimum one. Part **C** informs on the sampling rate in seconds. In the current implementation the sink sends 1000 frames per second. This sampling time is calculated dividing the number of frames by 1000. As a result, the shown value represents a sampling time of 0.502s or 502 frames per second. In **D** we have the following three values: the maximum signal value, the average signal value and the minimum signal value, and **E** shows the current signal value. In **F** we have the graphical scale for both graphic dimensions (volts in y-axis and seconds in x-axis). Finally, in **G** the screen shows the channel we are monitoring and the raw value of the signal on the sink.

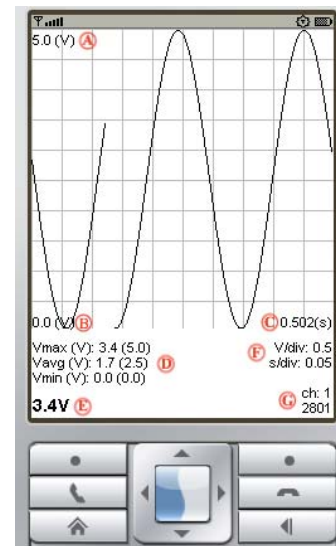


Fig. 5. Mobile tool screenshot of data presentation (graphical form).

To create the graphic shown in Figure 5, the raw signal is received from the sink (an 8 bit value, which corresponds to 4096 different values). Then we perform the calculation of the bottom line of the graphic, based on screen resolution and text occupancy. A transformation function is used to map the 4096 values on a zero to one scale. The normalized value is multiplied by the bottom position of the chart, and a dot represents the current normalized value.

VI. DEPLOYMENT AND VALIDATION

We deployed the mobile tool on several different devices, such as Nokia 5200 (Java-enabled phone, series 40 Symbian device UI, 2006), Nokia 6280 (Java-enabled phone, series 40 Symbian device UI, 2005), Nokia 6630 (Symbian OS V.8.0, Symbian series 60 UI, 2004), Nokia N80 Internet Edition (Symbian series 60 3rd edition, 2006) and, finally, Nokia N95 8GB (Symbian OS 9.2, S60 release 3.1, 2007). The application ran effortlessly on every one, without any issues.

Our main concern was with application performance on the more limited devices (Nokia 5200 and Nokia 6280). On the Nokia 5200, a call was made while the application was running, and the data continuously flowing from the sink. When the call ended, the data was available and presentation went smoothly as expected.

The mobile tool uses approximately 80KB of persistent storage space, and 6KB of RAM. In the first run of the application it takes around 17 seconds to perform an initial device discovery. After the first run the application takes about 4 seconds to establish the communication with the device.

Using two mobile devices to communicate with the same sink, the first one that reaches the sink communicates, while the other is blocked and generates a *S4* exception (communication port already open). However, after the device closes communication, other devices can communicate with the sink. This is a limitation of the current sink implementation, not the mobile application itself.

In terms of battery consumption, the application does not pose significant drain when compared to a scenario where Bluetooth connection is used. However, the constant use of screen backlight can pose significant battery drain (a scenario similar to the use of a GPS navigation software).

We validated the results against a PC application and the results are, as expected, exactly the same; since the data source is the same. The advantage here is the elimination of a PC to view the monitored signals.

VII. CONCLUSIONS

This paper proposed a mobile tool for biofeedback capable of running independently on the components of the body sensor network, and it is able to run on ordinary Java and Bluetooth-enabled cellular phones. This dramatically increases its usefulness, since the patient and the doctor can retrieve important health care data in behalf of the patient.

We show that a low processing power device is able to deal with relatively complex data and present a graphical representation in real-time.

In terms of future developments, we are considering the fact that a mobile device may be used to monitor several body sensor networks, with different calibration thresholds. As a result, we will create different user profiles associated to specific sink MAC address for easier connection configuration.

We hope to contribute for further utilization of body sensor networks for biofeedback acquisition, with the easy of use that our mobile tool can provide.

ACKNOWLEDGMENTS

Part of this work has been supported by the Portuguese Science and Technology Foundation (*Fundação para a Ciência e Tecnologia*), under contract PTDC/EIA/70830/2006, Institute of Telecommunications (*Instituto de Telecomunicações*), Portugal, Plux Biosensor Engineering, Covilhã, Portugal, and Euro-NF Network of Excellence of Seven Framework Programme of EC.

REFERENCES

- [1] J. A. Stankovic, Q. Cao, T. Doan, L. Fang, Z. He, R. Kiran, S. Lin, S. Son, R. Stoleru, and A. Wood, "Wireless Sensor Networks for In-Home Healthcare: Potential and Challenges", presented at High Confidence Medical Device Software and Systems Workshop, Pennsylvania, USA, 2005.
- [2] S. Brady, D. Diamond, B. Carson, D. O'Gorman, and N. Moyna, "Combining Wireless with Wearable Technology for the Development of on-body Networks", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.
- [3] L. Huaming and T. Jindong, "Body Sensor Network Based Context Aware QRS Detection", presented at Pervasive Health Conference and Workshops, Innsbruck, Austria, 2006.
- [4] J. Luprano, J. Sola, S. Dasen, J. M. Koller, and O. Chelelat, "Combination of Body Sensor Networks and On-body Signal Processing Algorithms: the Practical Case of MyHeart Project", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.
- [5] T. Falk, H. Baldus, J. Espina, and K. Klabunde, "Plug 'n Play Simplicity for Wireless Medical Body Sensors," *Mobile Networks and Applications*, vol. 12, pp. 143-153, 2007.
- [6] R. Sarpeshkar, "Ultra Low Power Electronics for Medicine", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Boston, MA, 2006.
- [7] L. Zhong, M. Sinclair, and R. Bittner, "A Phone-centered Body Sensor Network Platform Cost, Energy Efficiency & User Interface", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.
- [8] E. K. Reilly, E. Carleton, and P. K. Wright, "Thin Film Piezoelectric Energy Scavenging Systems for Long Term Medical Monitoring", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.
- [9] C. H. Chan, C. C. Y. Poon, R. C. S. Wong, and Y. T. Zhang, "A Hybrid Body Sensor Network for Continuous and Long-term Measurement of Arterial Blood Pressure", presented at 4th IEEE/EMBS International Summer School and Symposium on Medical Devices and Biosensors, Cambridge, UK, 2007.
- [10] D. Yun, J. Kang, J.-e. Kim, and D. Kim, "A Body Sensor Network Platform with Two-Level Communications", presented at IEEE International Symposium on Consumer Electronics (ISCE 2007), Irving, TX, USA, 2007.
- [11] S.-L. Chen, H.-Y. Lee, C.-A. Chen, C.-C. Lin, and C.-H. Luo, "A Wireless Body Sensor Network System for Healthcare Monitoring Application", presented at IEEE Biomedical Circuits and Systems Conference (BIOCAS 2007), Montréal, Qc, Canada, 2007.
- [12] U. Maurer, A. Rowe, A. Smailagic, and D. P. Siewiorek, "eWatch: A Wearable Sensor and Notification Platform", presented at international Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.
- [13] A. A. Reeves, J. W. P. Ng, S. J. Brown, and N. M. Barnes, "Remotely Supporting Care Provision for Older Adults", presented at International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), Cambridge, MA, USA, 2006.