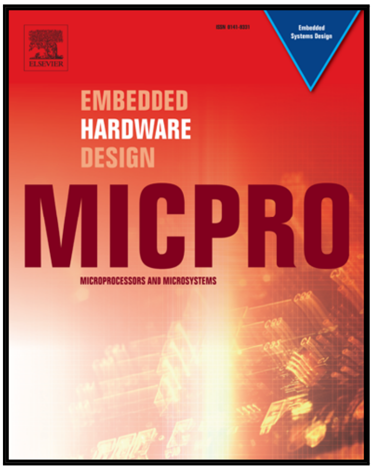
Accepted Manuscript



Real-time ECG Monitoring using Compressive sensing on a Heterogeneous Multicore Edge-Device

Hamza Djelouat, Abbes Amira , Faycal Bensaali, Christos Kotronis, Elena Politi, Mara Nikolaidou, George Dimitrakopoulos

PII: S0141-9331(19)30039-0
DOI: https://doi.org/10.1016/j.micpro.2019.06.009
Reference: MICPRO 2839

To appear in: *Microprocessors and Microsystems*

Received date: 18 January 2019
Revised date: 18 June 2019
Accepted date: 18 June 2019

Please cite this article as: Hamza Djelouat, Abbes Amira , Faycal Bensaali, Christos Kotronis, Elena Politi, Mara Nikolaidou, George Dimitrakopoulos, Real-time ECG Monitoring using Compressive sensing on a Heterogeneous Multicore Edge-Device, *Microprocessors and Microsystems* (2019), doi: https://doi.org/10.1016/j.micpro.2019.06.009

Real-time ECG Monitoring using Compressive sensing on a Heterogeneous Multicore Edge-Device

Hamza Djelouat[a], Abbes Amira[a,c,], Faycal Bensaali[a], Christos Kotronis[b], Elena Politi[b], Mara Nikolaidou[b], George Dimitrakopoulos[b]

[a]*College of Engineering, Qatar University, Doha, Qatar, P. O. Box: 2713*
[b]*Department of Informatics and Telematics, Harokopio University, Athens, 17671, Greece*
[c]*Faculty of Technology, De Montfort University, Leicester, UK*

Abstract

In a typical ambulatory health monitoring systems, wearable medical sensors are deployed on the human body to continuously collect and transmit physiological signals to a nearby gateway that forward the measured data to the cloud-based healthcare platform. However, this model often fails to respect the strict requirements of healthcare systems. Wearable medical sensors are very limited in terms of battery lifetime, in addition, the system reliance on a cloud makes it vulnerable to connectivity and latency issues. Compressive sensing (CS) theory has been widely deployed in electrocardiogramme ECG monitoring application to optimize the wearable sensors power consumption. The proposed solution in this paper aims to tackle these limitations by empowering a gateway-centric connected health solution, where the most power consuming tasks are performed locally on a multicore processor. This paper explores the efficiency of real-time CS-based recovery of ECG signals on an IoT-gateway embedded with ARM's big.LITTLE^{TM} multicore for different signal dimension and allocated computational resources. Experimental results show that the gateway is able to reconstruct ECG signals in real-time. Moreover, it demonstrates that using a high number of cores speeds up the execution time and it further optimizes energy consumption. The paper identifies the best configurations of resource allocation that provides the optimal performance. The paper concludes that multicore processors have the computational capacity and energy efficiency to promote gateway-centric solution rather than cloud-centric platforms.

Keywords: ambulatory ECG monitoring; heterogeneous multicore solution; compressive sensing; and edge computing.

1. Introduction

Internet of things (IoT)-based healthcare applications have become more attractive recently. After all, the growing older-adult population and unhealthy lifestyle trends are leading to the spread of chronic diseases, escalating demand for continuous clinician supervision and rising medical management costs [1]. An ambulatory real-time monitoring system can provide a low cost, efficient effective and convenient medical solution while maintaining the independence of the patients and improving their quality of life.

IoT-based remote monitoring systems consist of small wireless sensors continuously measuring signals of interest and transmitting them to a nearby gateway using low-power communication protocols [2]. The gateway further transmits the captured information to the IoT-platform (e.g. cloud) where data processing is conducted and useful information are extracted to meet the demands of the end users [2].

Nevertheless, wireless transmission of big data takes a toll on the battery of wearable medical devices. In addition, processing the data remotely makes the system vulnerable to connectivity and latency issues, and transmitting sensitive medical information to remote servers raises serious privacy concerns [3, 4, 5]. Subsequently, in most cases, this model fails to grant a reliable and robust solution that meets the quality of service required.

This paper presents some of the results obtained as part of the ongoing project "Embedded multi-core systems for multi-critical applications on the Internet of Things Era (**EMBIoT**)" [6, 7, 8, 9, 10, 11]. The paper is dedicated to addressing some of the aforementioned limitations by exploring mutlticore processors in the design of real-time power-efficeint gateway-centric connected health applications. To this end, the paper explores the use of the CS scheme on ECG signals acquisition and reconstruction [12, 13, 14].

CS is an emerging acquisition technique that merges the sampling and sensing stages into a single operation that allows an optimal recovery of the data using a fewer number of measurements than that required by the Shannon-Nyquist theorem. Several studies in the literature have shown the benefits of CS in terms of power consumption optimization [15].

Furthermore, in this paper, we also tackle latency and privacy issues related to cloud-based models by investigating the efficeiency of moving the processing from cloud to local heterogeneous multicore edge devices. Rather than a continuous stream of data, signals are reconstructed, classified, and analyzed locally. Then, the extracted information is transmitted to remote servers to be evaluated by the end-user. This approach can potentially address the shortcomings of remote health monitoring systems by removing dependence on clouds and consequently their transmission and computing latency, adding additional security and privacy measures locally before data is transmitted over the internet, and reducing the cost of cloud infrastructures by distributing computational-load to ubiquitous mobile-devices.

Within the aforementioned context, this paper qunatify CS-based ECG reconstruction on a heterogeneous multicore IoT-gateway using two real-time recovery algorithms: the subspace pursuit (SP) and orthogonal matching pursuit (OMP).

A thorough analysis of OMP's real-time performance was performed in [16] and [17]. This study employs an ARM big.LITTLE™ processor used in modern smartphones and provides a detailed analysis to evaluate the performance of the OMP algorithm on multiple cores.

Our experimental results show the superiority of SP under low signal distortion criteria, the energy efficiency on increasing the number of cores allocated to the reconstruction task, and the configurations where big cores—commonly known for being power-hungry—consume less energy per reconstructed window than the LITTLE energy-efficient cores.

The rest of the paper is organized as follows. Section 2 provides a general overview of the EMBIoT project, whereas, section 3 illustrates the criticalities that a remote monitoring system should address. Section 4 gives a brief overview about CS and the reconstruction algorithms used. The hardware components as well as the communication protocols used to monitor the ECG signals have been explained in section 5. Section 6 overviews the experimental setup and the evaluation metrics used. Hardware implementation results are presented and discussed in section 7. Section 8 concludes the paper.

2. EMBIoT Project

The EMBIoT project aims to incorporate embedded system solutions to IoT-based connected health applications. Embedded systems grant a reliable level of connection and optimized cooperation between different system components [18, 19, 20]. In the design of connected health solutions, conventional approaches are no longer sufficient in terms of computaion perfromance and real-time performance. For instance, single-core processors used as computing resources are running out of business, thus, exploring multi-cores can open a wider range of opportunities for more reliable platforms. In addition, with tremendous amount of medical data collection/transmission/processing, advanced algorithmic approaches will help to achieve real-time, energy efficient solutions.

The EMBIoT project goal is to provide a sustainable system design to accommodate the different requirements for modern connected health applications. Subsequenlty, EMBIoT will focus on proposing a breakthrough of the developed multicore technological solutions in IoT-based connected health, where real-time and mixed-criticality are important issues to address.

Connected health platforms are expected to demand a high level of criticalities. Therefore, EMBIoT aims to address these challenges by developing solutions to achieve the following requirements:

- Reconfigurability and adaptability of solutions.
- Affordable design cost by utilizing multicores platforms similar to the ones used in commercial smartphones.

- Management of multi-critical services and applications in dynamically changing, real-time environments.
- Scalability of the proposed algorithms for fusion, classification and compressive sensing.

The EMBIoT conceptual framework is depicted in Figure 1. EMBIoT consists of two main layers: the application layer and the technology layer. The application layer is based on the IoT, comprising both wearable systems and smart environments. Wearable systems applications are related to (i) the medical record of patients, (ii) their daily life activities and (iii) sensors capturing their vital signs. Smart environments may include buildings (e.g. Hospitals, robotic surgery rooms, houses), and vehicles (e.g. Ambulances). The technology to be used in order to build such applications will be based on multi-core platforms, where the challenges are numerous: dealing with node failures, carrying out accurate and efficient data fusion and effectively configuring the network are some of the critical issues to deal with.

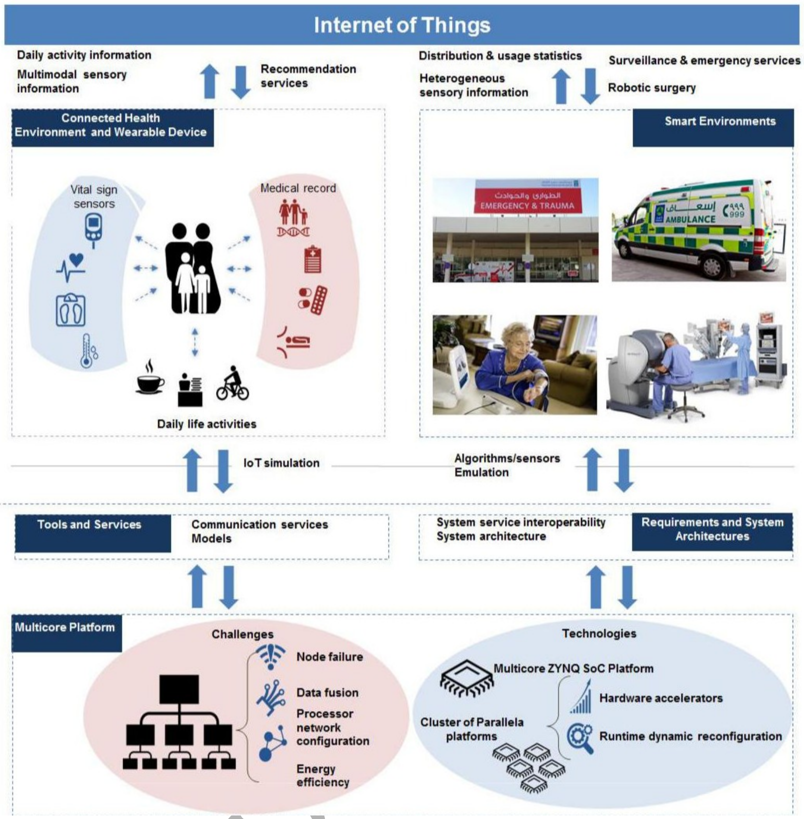


Figure 1: Overall EMBIoT Framework

3. EMBIoT Mixed Criticalities

As mentioned before, IoT-based remote health monitoring systems are mainly composed of hardware (e.g. networked sensors, smartphones, etc.) and software (e.g. specialized operating systems, linear algebra libraries, etc.) that can execute different criticality applications [21].

Healthcare is traditionally a domain requiring mostly safety-critical core components, since human lives can be jeopardized by a faulty implementation. Other components or functions can be mission-critical (i.e. a failure may affect the credibility of certain organizations -such as hospitals-, putting them at great risk), as well as operation-critical. Dealing with such criticalities, while developing the IoT components of a health system, is of significant importance. Remote elderly monitoring system (REMS), which will be used as a case study in this paper, can be characterized as a mixed-criticality real-time health system that supports multiple components and functions, each with a different criticality.

In this work, deploying heterogeneous multicore edge-devices, like an IoT-gateway, can address mixed criticalities, such as the real-time management of patient health data, the energy consumption of such devices, as well as security and privacy issues.

In particular, time is a safety criticality for health monitoring components. Gateways are responsible for the robust real-time treatment of sensor-collected data (e.g. ECG signals). Thus, such devices must possess the efficiency and technical capabilities to encapsulate a large number of functions and computationally complex tasks i.e. receive signals from all sensors, reconstruct, classify and compress them, while operating in real-time. Otherwise, the system is considered to be faulty since the real-time behavior is compromised.

Tied to the real-time performance, the energy consumption of such devices is operation-critical and must remain low in order to have a more sustainable IoT-based health system. Deploying heterogeneous multicore architectures on IoT components, for example, an ARM's big.LITTLE heterogeneous multicore platform, can couple high performance with energy efficiency. Finally, security and privacy are mission-critical. It is important to avoid errors (e.g., corruptions) to the collected signals and ensure that they are safely delivered to the intended receiver (e.g. the gateway), while the patient data needs to remain private. In order to meet this criticality, solutions like migrating the signal reconstruction to a local edge IoT-gateway rather than transmitting them directly to the Cloud, can address such issues. Figure 2 illustrates a graphical representation of the performance, energy and security criticalities that a REMS gateway component must address.

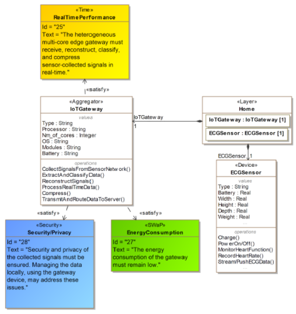


Figure 2: REMS criticalities

4. Compressive Sensing

In CS, signals that are sparse in certain domains can be captured using fewer measurements than that required by the Nyquist rate [12]. A signal $\boldsymbol{x} \in \mathbb{R}^n$ is sparse if it can be represented in some basis as:

$$\boldsymbol{x} = \sum_{i=1}^{n} \boldsymbol{s}_i \boldsymbol{\psi}_i = \boldsymbol{\psi} \boldsymbol{s} \tag{1}$$

where only few elements of the vector $\boldsymbol{s} \in \mathbb{R}^n$ are nonzero. The matrix $\boldsymbol{\psi} \in \mathbb{R}^{n \times n}$ represents the domain where the signal $\boldsymbol{x}$.

CS operation is non-adaptive sensing and can be represented as follows:

$$\boldsymbol{y} = \boldsymbol{\phi x} = \boldsymbol{\phi} \psi s = \boldsymbol{A s} \tag{2}$$

where $\boldsymbol{y} \in \mathbb{R}^m$ is the measured signal (compressed) and $m << n$. $\boldsymbol{\varphi} \in \mathbb{R}^{m \times n}$ is a sensing matrix and $\boldsymbol{A} = \boldsymbol{\varphi \psi} \in \mathbb{R}^{m \times n}$ is a matrix linking the sparse representation of $\boldsymbol{x}$ with the compressed signal $\boldsymbol{y}$.

In order for the CS acquisition to be efficient and capture the salient information about the signal, the sensing matrix $\boldsymbol{\varphi}$ should: 1) meet the restricted isometry property (RIP) and 2) be incoherent with the sparsifying basis $\boldsymbol{\psi}$ [22]. Consequently, designing a sensing matrix is not a trivial task, but fortunately, it has been shown that a $\boldsymbol{\varphi}$ can satisfy both conditions with an arbitrary basis if its elements are drawn from an independent identically distrusted (i.i.d) values such as those obtained from a Gaussian or a Bernoulli distribution [22].

In order to estimate the original signal $\boldsymbol{x}$ from the compressed data $\boldsymbol{y}$, different approaches can be adopted such as convex optimization and greedy algorithms. The latter are widely considered for real-world applications as they exhibit a simple implementation architecture, a fast convergence to the solution and a sub-optimal recovery. The most well known greedy algorithms are OMP [23], compressive sensing matching pursuit (CoSaMP) [24]and SP [25]. It is worth mentioning that in this paper both OMP and SP have been used to reconstruct the compressed acceleration data.

5. System Setup

The paper presents a REMS. First, the proposed platform collect compressed ECG data from patients. Afterwards, the compressed ECG is transmitted to the gateway via a Bluetooth connection. At the gateway, the data is reconstructed and classified to detect abnormalities in the patient's heart beats. The proposed platform relies on the- Shimmer3 device ™ [26] for data acquisition and ODROID XU4 [27] for data processing. The overall system framework is shown in Figure 3.

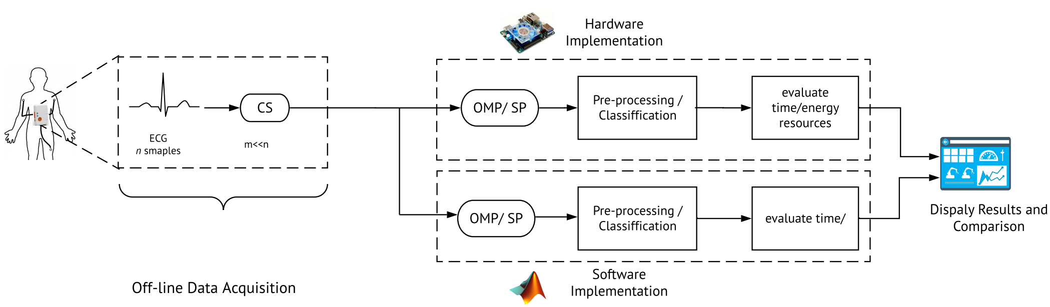


Figure 3: Proposed framework

5.1. Shimmer device^{TM}:

presents a low power and a lightweight sensing device used to acquire both ECG and acceleration data to empower two different applications, namely, ECG abnormality detection and automatic fall detection. The data acquired by the shimmer are compressed before transmission following CS techniques, where different compression modes can be performed. On the sensing level, different constraints have to be met such as security, low power transmission, etc.

Software wise, the Shimmer operates on a C-based firmware called LogAndStream [28]. To enable a CS-based ECG acquisition, the firmware has been modified accordingly.

The new firmware will allow the user to execute the following commands:

- Set_Sampling_Rate: Sets the Shimmer sampling frequency
- SET_SENSORS: Activates and deactivates the appropriate sensor (ECG, Accelerometer)
- SHIMMER_COMMANDS: Takes 10 Bytes and configures the shimmer parameters such as gain(refer to ECG document)
- SET_WINDOW_AND_MODE: Sets the transmission mode (real time, 1 second, 1.5 second) and the processing mode (raw, adjusted, high quality compression and low quality compression)
- START_STREAMING_COMMAND: Start the streaming
- STOP_STREAMING_COMMAND:End the streaming

After making the necessary modification on the code, the firmware has been compiled using Texas Instruments Code Composer studio (TI 4.4.8 compiler). The result of the compilation is a text file that can be uploaded to the Shimmer using the provided dock and Consensys [29].

5.2. *ODROID XU4* ^{TM}

is a heterogeneous multicore platform (HMP) featuring ARMs big.LITTLE$^{\mathrm{TM}}$ technology. The ODROID-XU4 presents a modern processing platform with a enhanced computing resources. The XU4 is equipped with a Samsungs Exynos 5422 octa-core processor which is embedded with a cluster of four Cortex-A15 cores (big) and a cluster of four Cortex-A7 cores (LITTLE). Furthermore, the XU4 architecture provides an optimized processing features leading to an efficient power consumption behaviour. Subsequently, leveraging the big.LITTLE$^{\mathrm{TM}}$ technology, the XU4 can efficiently utilize a maximum of its eigth cores to manage computationally intensive tasks [30] as shown in Figure 4.

The XU4 plays the role of an edge computing platform to link the acquired data from the sensor to the cloud. On the XU4, the received compressed data has to be reconstructed and then classified to detect a different abnormal event, i.e, ECG arrhythmia classification and automatic fall detection. Consequently, the XU4 will provide the user with a wide range of applications each with a set of constraints, requirements and defined quality of service (QoS).

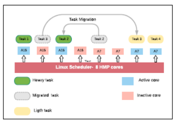


Figure 4: Task management within the ODROID XU4 platform [30]

5.3. *REMS Implementation*

In order to establish the connection between the Shimmer and the XU4, a complex software tool has been developed based on both C++ and Python. The developed application, namely, RemsClient, permits to connect and configure multiple Shimmer devices, process their data efficiently, and display them to the user. The main Python program is RemsClient.py and is considered the Parent Application, this app will contain the GUI of the program and has high-level control over the programs below it. The Parent is capable of spawning independent Child processes each reading from a Shimmer device independently. The Child can spawn its own C++ based process if it needs it. The C++ process can quickly perform complex data processing (such as CS signal reconstruction) or data classification.

The communication between the Shimmer and the RemsClient established in the new firmware is performed as follows:

1. Shimmer initializes global variables by setting all buffer arrays to zero;
2. It waits for the RemsClient to send its configuration commands (discussed earlier);
3. After being configured, it remains idle until the user starts the streaming process;
4. When the stream starts, the firmware samples requested ECG until the buffer is full;
5. When full, the Shimmer will prepare a Bluetooth message to be sent to the RemsClient after processing the data if that is required;
6. If the user did not stop the stream, the Shimmer will send the data to the RemsClient and empty the buffer;
7. Repeat 4 to 6 until the stream is stopped.

The principle of operation of the Parent is based on two classes, the Shimmer Class and the ChildManager Class. The Shimmer Class allows the parent to create a Shimmer object per physical Shimmer device. Then, the parent can use the Shimmer Class methods to adjust the Shimmer configuration, allowing to ignore the low-level details of the configurations and simply call high-level methods. Afterwards, the configured Shimmer object can be passed to a ChildManager object, the ChildManager object has methods that allow to start a Child process, read data from it, and terminate it, again giving a high-level interface for control. Both classes have been filled with error handles and exceptions, so it would be impossible to call a method or a function incorrectly.

Once the Parent spawns a Child, it passes the Shimmer object. The Child uses the Shimmer object to access the configuration selected by the user and connect to and configure the physical Shimmer device. When the stream starts, the Child coordinates the data between the Shimmer, C++, and the Parent.

There are two messaging protocols in the client, the first being Python Parent to Python Child which is simple, relaying on a custom parent Child message class and Pythons multiprocessing pipes. The other messaging system is between the Python Child and C++, relying on the ZMQ library and protocol buffer.

The ZMQ library is common to Python and C++. It allows to start a Python server and a C++ client and connect them to each other through an inter process communication (IPC) socket. The socket however only accepts strings or serialized information, meaning that vectors and lists (a common type for our data) cannot be sent directly through a ZMQ socket.

This serialization issue can be solved by using a common data format such as Java Script Object Notation (JSON). But since JSON serialization and deserialization is slow and our system is real-time sensitive, we use Google's Protocol Buffers (or Protobuf). Protobuf allows us to write a Proto file outside of Python and C++. This proto file contains custom messaging classes. The custom messaging classes can be complied into C++ classes and Python classes, and hence, now we have common objects between both languages. Protobuf also provides us with serialize and deserialize methods. To summurize, a message from Python to C++ requires the following:

1. Start a ZMQ sever in Python.
2. Connect to ZMQ server from C++.
3. To send data from Python to C++. Create a protobuf class message, serialize it and send it through the socket.
4. To receive the data in C++, receive the serialized protobuf message, and deserialze it using C++ version of this class.

6. Experimental Setup

The main objective of this work is to quantify the signal reconstruction performance of OMP and SP. We consider multiple signal dimensions, the time and energy they require to run on ARM's big.LITTLE$^{\text{TM}}$ HMP at different processing configurations.

The evaluation of the reconstruction performance is carried out in MATLAB. A set of ECG records (103, 106, 109, 113, 116, 119, 208, 217, 219, 221) from the MIT-Arrhythmia database [31] have been used. The percentage root-mean-square difference (PRD) metric is used to evaluate the reconstructed signal quality and it is defined as follows:

$$\mathbf{PRD} = \frac{\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2}{\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_2} \times 100 \tag{3}$$

where $\tilde{\boldsymbol{x}}$ is the mean of the original signal. To evaluate the effect of the signal dimensions on **PRD**, four signal lengths, $n = [1.0\ 1.5\ 2.0\ 2.5] \times f_s$, are considered. The compression ratio is defined as $CR = n/m$ and the average **PRD** is computed using 300 consecutive windows from each record (total $= 3000$ segments).

ECG signals are not continuously transmitted to cloud for visual inspection but are locally analyzed for the majority of the time, and hence medical-grade ECGs are only required at the request of the end user or in case of emergencies. Consequently, we define two PRD quality targets as follows:

- High quality (HQ) target suitable for clinician inspection.
- Low quality (LQ) target that allows for accurate data classification and feature extraction.

The HQ target is defined based on [32], where clinicians rated ECG signals at varying distortion levels, concluding that $\mathbf{PRD} \leq 9$ is "good" for visual inspection. The LQ target is defined empirically by quantifying abnormality classification accuracy at different signal distortion levels. Two heart beat classes are used for classification: normal beat (N) and premature ventricular contraction beat (V), and four records (106, 119, 208, 221) were tested. The average accuracy of the K-nearest neighbor (KNN) and the extend-nearest neighbor (ENN) algorithm in detecting the class of heart beats in the aforementioned records is presented in Figure 5. The results reveal that signals reconstructed with $\mathbf{PRD} \leq 25$ provide a near identical or slightly deteriorated (≈ 2 %)classification accuracy in comparison to the original signal (**PRD** = 0). Thus, the LQ target is set as **PRD** ≤ 25 %.

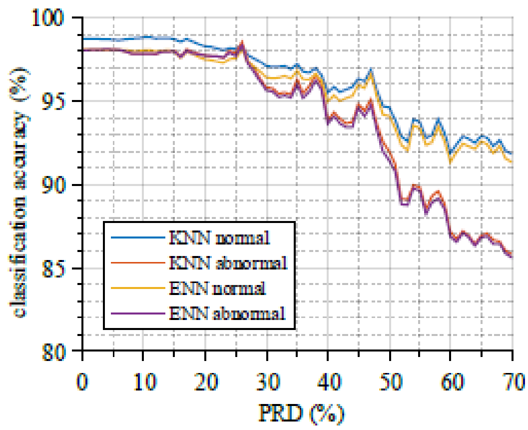


Figure 5: ENN abnormality detection accuracy versus ECG signal distortion in PRD

Real-time performance and energy consumption are assessed using Hard-Karnel's Odroid XU4 board [33]. The XU4 operates Ubuntu 16.04 (Linux 3.10 + armv7) and the algorithms were written in C++ using the `Armadillo` (v8.3) library [34] for linear algebra and complied with `gcc` (v5.4).

The reconstruction time is computed over a 50 consecutive trails at a particular signal dimension with variable core type, number, and frequency. Since increasing the signal window length increases both computational complexity of reconstruction and time-gap between two consecutive segments, the average reconstruction time is normalized over a real-time window defined $T_r = n/f_s$. Power consumption is measured and averaged using a digital wattmeter in series with the boards power supply. Afterwards, the energy required for reconstruction is defined as average power $\times$ reconstruction-time.

ACCEPTED MANUSCRIPT

7. Results

7.1. Reconstruction Quality

Figure 6 presents a comparison between OMP and SP in terms of PRD using diffrent ECG signal dimensions.

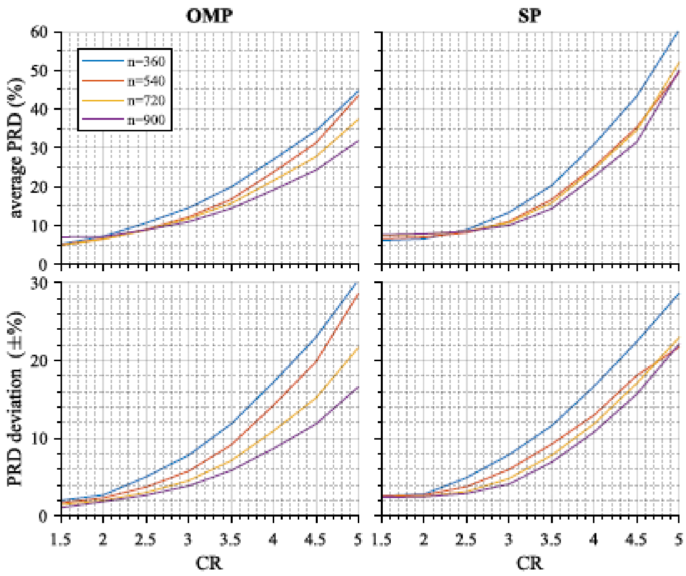


Figure 6: Signal reconstruction performance of OMP and SP at different ECG window dimensions

OMP performs better than SP at most CRs, however, SP has lower reconstruction error at CRs= $[2.5-3.5]$. Generally, higher segment lengths decrease reconstruction error as it reduces both average **PRD** and its deviation. This can be more clearly observed in Table 1 where the highest possible CR that provides the targeted **PRDs** with a probability of 98% is computed. The data concludes that SP allows for higher compression considering the HQ criterion, while for the LQ target OMP is preferred. Increasing n from $(360 \rightarrow 900)$ renders approximately a 10% and a 9% improvement in signal compressibility which is beneficial for the sensing node but will also increase the computational complexity and energy required for reconstruction at the gateway. For the remainder of this paper, every window size will be compressed with its corresponding highest possible CR, and HQ and LQ will refer to signals reconstructed using SP and OMP respectively.

	OMP				SP			
n	360	540	720	900	360	540	720	900
HQ	2.77	3.06	3.00	3.20	**2.85**	**3.13**	**3.04**	**3.28**
LQ	**4.1**	**4.43**	**4.47**	**4.61**	3.78	4.12	4.08	4.30

Table 1: Highest CR that achieves the LQ and HQ targets with probability of 98%

7.2. Real-time Performance and Energy Consumption

In CS, several parameters govern the performance of the reconstruction algorithm when implemented on the XU4. these parameters are the signal dimension, number of cores running the process as well as their operating frequencies

In here, we only presented the cases where $n = 360$ and $n = 900$. Normalized reconstruction time for OMP and SP is shown in Figure 7. Obtained results show that it is possible to achieve real-time recovery using the lowest number

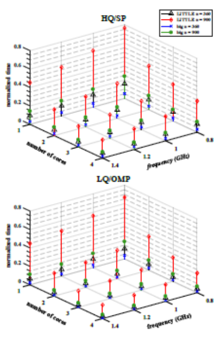


Figure 7: Normalized time of HQ and LQ ECG signal reconstruction using SP and OMP at different signal lengths, core type, number of cores, and core frequency

of cores at its minimum frequency ((1 × A7 at 0.8 GHz) for the largest signal dimension, i.e, $n = 900$.

Furthermore, implementation on the A15 processors speeds up the reconstruction process to a factor $[2 - 4]$ compared to A7-based implementation. It is worth noting that the number of cores used is the most dominant factor in speeding up the process. Whereas, the frequency of the cores does not show to affect the speed of the recovery algorithm. For instance, using $2 \times$A7 at 0.8 GHz is 25% faster than running it on a $1 \times$ A7 at 1.2 GHz.

Moreover, increasing the cores' frequency from 0.8 GHz to [1.0 1.2 1.4] GHz results in a [20 35 47]% and a [17 31 41]% faster reconstruction using SP and OMP respectively. Whereas, increasing the number of cores used from 1 to [2 3 4] results in a [37 56 68]% and [44 71 86]% speed up for SP and OMP respectively.

Figure 8. shows the energy consumption per window for diffrent procesor configurations. The following observation can be drawn:

- Intuitively, power consumption increases as more cores are used.
- The optimal result in terms of power consumption is obtained using $1 \times$A15 at 0.8 GHz for the case of $n = 360$. For a highest dimensional signal, $1 \times$ A15 at 0.8 GHz renders the best results, although replacing the A7

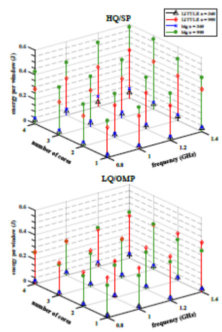


Figure 8: Energy required for HQ and LQ ECG signal reconstruction using SP and OMP at different signal lengths, core type, number of cores, and core frequency

by A15 processor increase the power consumption only by a factor of 1.2, however, it speeds up the process by 3.2 factor.

- The LQ scenario shows an interesting result where A15 cores are more energy-efficient than the A7 cores for different combinations of core's "number-frequencies". In addition, the more A7 used, the less power is consumed. This can be explained by the fact that parallelization of OMP on A7 cores results in a significant decrease in computational-time, enough to compensate for the increased power required for the extra cores.
- For LQ reconstruction, the combination $1 \times$ A15 at 0.8 GHz could be selected as the best configuration as it exhibits the less power consumption with a fast reconstruction time.

8. Conclusions

This paper presents a solution to IoT connected health based on edge-computing devices embedded with HMPs. Such solution addresses the latency and privacy issues that can be bottleneck the reliability of healthcare systems.

The EMBIoT project objective to empower a complete solution to healthcare application by relying on multicore platforms and advanced algorithmic approaches such as CS and machine learning techniques. The aim is to address issues related to power efficiency, real-time performance, reliable and secure communication. EMBIoT focuses also on the identification and modelling criticalities of healthcare IoT systems.

The presented work demonstrates the efficiency of CS-based ECG reconstruction implemented on an ARM big.LITTLE^{TM} processor. The analysis has been conducted using different signal sizes as well a various combination of the core's "number-type-frequency". The performance has been evaluated in terms of energy consumption and reconstruction quality/execution time.

Moreover, multicore processors have shown their ability to reconstruct ECG segments of up to 2.5 sec using the minimum computational resource. The obtained performance that the more cores used for the reconstruction process, the lower reconstruction time and is more energy-efficient than increasing the cores' frequency.

HMPs are able to efficiently process computationally complex tasks in real-time and can be a viable solution to a more sustainable IoT-based remote health monitoring system.

Acknowledgments

This paper was made possible by National Priorities Research Program (NPRP) grant No. 9-114-2-055 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Xiao-Fei Teng, Yuan-Ting Zhang, C. Poon, P. Bonato, Wearable Medical Systems for p-Health, IEEE Reviews in Biomedical Engineering 1 (2008) 62–74. doi:10.1109/RBME.2008.2008248.

[2] D. Minoli, Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2m Communications, Wiley, Hoboken, New Jersey, 2013.

[3] M. M. Baig, H. Gholamhosseini, M. J. Connolly, A comprehensive survey of wearable and wireless ECG monitoring systems for older adults, Medical & Biological Engineering & Computing 51 (5) (2013) 485–495. doi:10.1007/s11517-012-1021-6.

[4] A. R. Nambiar, N. Reddy, D. Dutta, Connected health: Opportunities and challenges, IEEE, 2017, pp. 1658–1662. doi:10.1109/BigData.2017.8258102.

[5] F. Samie, V. Tsoutsouras, L. Bauer, S. Xydis, D. Soudris, J. Henkel, Computation offloading and resource allocation for low-power IoT edge devices, IEEE, 2016, pp. 7–12. doi:10.1109/WF-IoT.2016.7845499.

[6] H. Baali, H. Djelouat, A. Amira, F. Bensaali, Empowering technology enabled care using iot and smart devices: A review, IEEE Sensors Journal 18 (5) (2018) 1790–1809.

[7] M. Al Disi, H. Djelouat, C. Kotroni, E. Politis, A. Amira, F. Bensaali, G. Dimitrakopoulos, G. Alinier, Ecg signal reconstruction on the iot-gateway and efficacy of compressive sensing under real-time constraints, IEEE Access 6 (2018) 69130–69140.

[8] H. Djelouat, H. Baali, A. Amira, F. Bensaali, Iot based compressive sensing for ecg monitoring, in: Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2017 IEEE International Conference on, IEEE, 2017, pp. 183–189.

[9] M. Al Disi, H. Djelouat, A. Amira, F. Bensaali, The accuracy and efficacy of real-time compressed ecg signal reconstruction on a heterogeneous multicore edge-device, in: 2018 21st Euromicro Conference on Digital System Design (DSD), IEEE, 2018, pp. 458–463.

[10] M. Al Disi, H. Baali, H. Djelouat, A. Amira, F. Bensaali, C. Kontronis, G. Dimitrakopoulos, G. Alinier, An efficient compressive sensing method for connected health applications, in: Proceedings of SAI Intelligent Systems Conference, Springer, 2018, pp. 365–373.

[11] H. Djelouat, H. Baali, A. Amira, F. Bensaali, Joint sparsity recovery for compressive sensing based eeg system, in: Ubiquitous Wireless Broadband (ICUWB), 2017 IEEE 17th International Conference on, IEEE, 2017, pp. 1–5.

[12] A. M. R. Dixon, E. G. Allstot, D. Gangopadhyay, D. J. Allstot, Compressed Sensing System Considerations for ECG and EMG Wireless Biosensors, IEEE Transactions on Biomedical Circuits and Systems 6 (2) (2012) 156–166. doi:10.1109/TBCAS.2012.2193668.

[13] C. Karakus, A. C. Gurbuz, B. Tavli, Analysis of Energy Efficiency of Compressive Sensing in Wireless Sensor Networks, IEEE Sensors Journal 13 (5) (2013) 1999–2008. doi:10.1109/JSEN.2013.2244036.

[14] F. Chen, A. P. Chandrakasan, V. M. Stojanovic, Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors, IEEE Journal of Solid-State Circuits 47 (3) (2012) 744–756. doi:10.1109/JSSC.2011.2179451.

[15] H. Mamaghanian, N. Khaled, D. Atienza, P. Vandergheynst, Compressed Sensing for Real-Time Energy-Efficient ECG Compression on Wireless Body Sensor Nodes, IEEE Transactions on Biomedical Engineering 58 (9) (2011) 2456–2466. doi:10.1109/TBME.2011.2156795.

[16] D. Bortolotti, A. Bartolini, M. Mangia, R. Rovatti, G. Setti, L. Benini, Energy-Aware Bio-signal Compressed Sensing Reconstruction: FOCUSS on the WBSN-Gateway, IEEE, 2015, pp. 120–126. doi:10.1109/MCSoC.2015.34.

[17] D. Bortolotti, M. Mangia, A. Bartolini, R. Rovatti, G. Setti, L. Benini, Energy-Aware Bio-signal Compressed Sensing Reconstruction on the WBSN-gateway, IEEE Transactions on Emerging Topics in Computing (2016) 1–1doi:10.1109/TETC.2016.2564361.

[18] H. Baali, H. Djelouat, A. Amira, F. Bensaali, X. Zhai, Efficient hardware implementation of the l 1regularized least squares for iot edge computing, in: Ubiquitous Wireless Broadband (ICUWB), 2017 IEEE 17th International Conference on, IEEE, 2017, pp. 1–5.

[19] H. Baali, X. Zhai, H. Djelouat, A. Amira, F. Bensaali, Inequality indexes as sparsity measures applied to ventricular ectopic beats detection and its efficient hardware implementation, IEEE Access 6 (2018) 9464–9472.

[20] H. Djelouat, X. Zhai, M. Al Disi, A. Amira, F. Bensaali, System-on-chip solution for patients biometric: A compressive sensing-based approach, IEEE Sensors Journal 18 (23) (2018) 9629–9639.

[21] A. Burns, R. Davis, Mixed criticality systems-a review, Department of Computer Science, University of York, Tech. Rep (2013) 1–69.

[22] E. Candes, M. Wakin, An Introduction To Compressive Sampling, IEEE Signal Processing Magazine 25 (2) (2008) 21–30. doi:10.1109/MSP.2007.914731.

[23] J. A. Tropp, A. C. Gilbert, Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit, IEEE Transactions on Information Theory 53 (12) (2007) 4655–4666. doi:10.1109/TIT.2007.909108.

[24] D. Needell, J. A. Tropp, Cosamp: Iterative signal recovery from incomplete and inaccurate samples, Applied and computational harmonic analysis 26 (3) (2009) 301–321.

[25] W. Dai, O. Milenkovic, Subspace Pursuit for Compressive Sensing Signal Reconstruction, IEEE Transactions on Information Theory 55 (5) (2009) 2230–2249. doi:10.1109/TIT.2009.2016006.

[26] A. Burns, B. R. Greene, M. J. McGrath, T. J. O'Shea, B. Kuris, S. M. Ayer, F. Stroiescu, V. Cionca, Shimmer–a wireless sensor platform for noninvasive biomedical research, IEEE Sensors Journal 10 (9) (2010) 1527–1534.

[27] O.-X. Lite, Odroid, heterogeneous multi-processing (hmp) solution, website available on-line at: http://www. hardkernel. com/main/products/prdt info. php.

[28] Shimmerresearch, https://github.com/ShimmerResearch/shimmer3, accessed: 2019-01-14.

[29] Shimmer documentation, http://www.shimmersensing.com/support/wireless-sensor-networks-documentation/, accessed: 2019-01-14.

[30] Odroid-xu4 description, https://www.hardkernel.com/shop/odroid-xu4/, accessed: 2019-01-14.

[31] G. Moody, R. Mark, The MIT-BIH Arrhythmia Database on CD-ROM and software for use with it, IEEE Comput. Soc. Press, 1991, pp. 185–188. doi:10.1109/CIC.1990.144205.

[32] Y. Zigei, A. Cohen, A. Katz, The weighted diagnostic distortion (WDD) measure for ECG signal compression, IEEE Transactions on Biomedical Engineering 47 (11) (2000) 1422–1430. doi:10.1109/TBME.2000.880093.

[33] ODROID — Hardkernel, http://www.hardkernel.com.

[34] C. Sanderson, R. Curtin, Armadillo: A template-based C++ library for linear algebra, The Journal of Open Source Software 1 (2) (2016) 26. doi:10.21105/joss.00026.

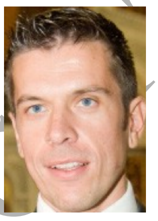
Hamza Djelouat received a BEng degree in telecommunication from the Institute of Electrical and Electronic Engineering, Boumerdes, Algeria, in 2013 and an MSc degree in electronic systems majoring telecommunication from The Ecole Militaire Polytechnique (EMP), Algiers, Algeria, in 2015. He is currently a research assistant at Qatar University. His research interests are mainly in digital signal processing and the applications of compressive sensing in wireless networks and connected health.

Abbes Amira (S'99-M'01-SM'07) received the Ph.D. degree in the area of computer engineering from Queen's University, Belfast, U.K., in 2001. He took many academic and consultancy positions, including his recent positions as Associate Dean for Research and Graduate Studies at Qatar University and also Professor of Visual Communications with the University of the West of Scotland, U.K. He published more than 300 papers in top journals and conferences and secured significant funding from government and indu. His research interests include reconfigurable computing, signal processing, IoT, connected health and security. He is an IET Fellow, HEA Fellow, Senior Member ACM.

Faycal Bensaali (S'03-M'06-SM'15) obtained a Dipl-Ing (M.Eng.) in electronics from University of Constantine and a Ph.D. in Electronic and Computer Engineering from Queen's University, Belfast. He is currently an Associate Professor in Electrical Engineering at Qatar University. Dr. Bensaali took other academic positions at the University of Hertfordshire-UK and Queen's University Belfast-UK. His research interests are mainly in embedded systems and high performance computing, custom computing using FPGAs, image and vision systems and connected health. Dr. Bensaali serves as Guest Editor of IEEE IoT journal and General Chair, Workshop Chair and TPC Member of a number of international conferences and workshops. He has authored/co-authored over 100 scientific papers in international journals and conference proceedings. He is an HEA Associate and IEEE senior member.

Christos Kotronis received his B.Sc. in Informatics & Telematics from Harokopio University in 2014 and M.Sc. in Web Engineering from the Department of Informatics and Telematics of Harokopio in 2016. He is currently a PhD candidate at the same department, under the supervision of Professor Mara Nikolaidou. He gained a three-year 2017-2019 Engineering and Technology Sciences scholarship from the General Secretariat for Research and Technology (GSRT) and the Hellenic Foundation for Research and Innovation (HFRI) for conducting his doctoral research. This research, titled "Systems-of-Systems engineering: An integrated model-driven methodology for complex systems", proposes model-driven approaches, utilizing SysML as a modeling language to automatically generate performance and cost models. Case studies on transportation, e-health, and sensor networks, are explored, taking into account the dynamic behavior of such Systems-of-Systems (SoS) under real-time conditions. His current position is teaching assistant at the Dept. of Informatics & Telematics of Harokopio University of Athens 2016.

Elena Politis holds a B.Sc. in Physics from the National and Kapodistrian University of Athens and a M.Sc in Informatics and Telematic Services with honors from Harokopio University. She is currently a PhD candidate at the department of Informatics and Telematics at Harokopio University. Her dissertation is dedicated to the Internet of Underwater Things. She speaks fluent Greek and English and has a very good of the German language. Her research interests revolve around the Internet of Things, related applications and services and optimization algorithms for autonomous underwater vehicles.



George Dimitrakopoulos has received his bachelor degree in Electrical and Computer Engineering in 2002 and his Ph.D. in 2007, from the National Technical University of Athens, and the University of Piraeus, respectively. He is currently an assistant professor at the department of Informatics and Telematics of Harokopio University of Athens (HUA) (2010), as well as a coordinator of funded projects at Infineon Technologies A.G. (2018) and a senior consultant and project manager at Intrasoft International S.A. (2015). He is the author of about 120 publications in international journals and conferences, whereas he has been involved in more than 20 internationally funded RD projects. His research interests include the design and development of safety and emergency management algorithms for smart devices, as well as the design and development of strategies for the optimization of wireless networks based on cognitive networking principles.