Journal of Biomedical Informatics 59 (2016) 1–14

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Resource efficient data compression algorithms for demanding, WSN based biomedical applications

Christos P. Antonopoulos *, Nikolaos S. Voros

Technological Educational Institute of Western Greece, Computer and Informatics Engineering Department, National Road Antiriou-Ioanninon, 30020 Antirio, Greece

ARTICLE INFO

Article history:
Received 21 January 2015
Revised 16 June 2015
Accepted 29 October 2015
Available online 7 November 2015

Keywords:
Biomedical signal processing
Data compression algorithms
Wireless sensor networks
Performance evaluation
ECG/EEG signal compression
Epilepsy monitoring

ABSTRACT

During the last few years, medical research areas of critical importance such as Epilepsy monitoring and study, increasingly utilize wireless sensor network technologies in order to achieve better understanding and significant breakthroughs. However, the limited memory and communication bandwidth offered by WSN platforms comprise a significant shortcoming to such demanding application scenarios. Although, data compression can mitigate such deficiencies there is a lack of objective and comprehensive evaluation of relative approaches and even more on specialized approaches targeting specific demanding applications. The research work presented in this paper focuses on implementing and offering an in-depth experimental study regarding prominent, already existing as well as novel proposed compression algorithms. All algorithms have been implemented in a common Matlab framework. A major contribution of this paper, that differentiates it from similar research efforts, is the employment of real world Electroencephalography (EEG) and Electrocardiography (ECG) datasets comprising the two most demanding Epilepsy modalities. Emphasis is put on WSN applications, thus the respective metrics focus on compression rate and execution latency for the selected datasets. The evaluation results reveal significant performance and behavioral characteristics of the algorithms related to their complexity and the relative negative effect on compression latency as opposed to the increased compression rate. It is noted that the proposed schemes managed to offer considerable advantage especially aiming to achieve the optimum tradeoff between compression rate-latency. Specifically, proposed algorithm managed to combine highly completive level of compression while ensuring minimum latency thus exhibiting real-time capabilities. Additionally, one of the proposed schemes is compared against state-of-the-art general-purpose compression algorithms also exhibiting considerable advantages as far as the compression rate is concerned.

1. Introduction

Epilepsy comprises a disease of profound social significance while it represents one of the most important medical challenges, troubling human kind throughout its history. Although significant research effort has been devoted for many decades, relatively little advancements can be reported in understanding, analyzing, identifying, categorizing and treating it. In this direction [1] offers a review on the results of the Global Campaign against Epilepsy aiming to shed light on the challenges and difficulties people with epilepsy and their families are facing. In addition, in [2] it is clearly depicted that it is also important to look into issues like stigmatization, social exclusion, medical or psychiatric comorbidities. Another aspect related to the economical side effects of epilepsy and respective treatments is discussed in [3–6].

Toward gaining a better understanding, advancements of Wireless Sensor Networks (WSN) are attracting increasing interest both by academia and industry in the area of Epilepsy study. On one hand this is indicated by the various relative projects using WSN advancements to study epilepsy [7] and home health care [8]. On the other hand, increased interest is noticed in lots of research efforts ranging from study of specific WSN medical applications [9] to performance study of WSN networks [10,11] and efficient handling of large volumes of data [12]. In that respect long term, non-intrusive monitoring of patients, during an extended period of time, has been used for many years to extract valuable conclusions and indications. In [13] extensive reports on intensive EEG/video monitoring are presented. Authors [14] EEG monitoring is used to offer a quantitative review of seizure risk in specific considerations, while [15] focuses on epileptic and non-epileptic disorder distinction. However, relative studies in diverse environments through adequate WSN equipment are expected to offer significant insights and advancements. In that context European Research

* Corresponding author.
E-mail address: cantonopoulos@teimes.gr (C.P. Antonopoulos).

http://dx.doi.org/10.1016/j.jbi.2015.10.015
1532-0464/

projects are devoted in Epilepsy study and monitoring using ultra low power wireless platforms [7,16]. Additionally, platforms specifically designed and implemented to wirelessly aggregate respective modalities are attracting intruding attention as indicated in efforts [17,18] where a wireless neural interface is the main objective. Also in respective efforts are presented focusing on EEG data aggregation [19–21]. Such implementations enable medical personnel to perform accurate, secure and non-intrusive monitoring and study of phenomena not possible through conventional approaches.

However, from an engineering point of view respective studies are based on the capability to acquire large volumes of data (digitized physiological measurements), for extended periods of time, which must be either stored locally or transmitted to an aggregation point. The two data types of paramount importance in epileptic seizure study, resulting into excessive amount of accumulated data, are *Electroencephalography* (EEG) and *Electrocardiography* (ECG) measurements. Typical acquisition devices produce samples represented as 16 bit numbers. Furthermore, a wired EEG setup is usually comprised of 64 sensors with sampling frequency up to 2.5 kHz, while ECG requires typically 4 sensors with adequate sampling frequency of a few hundreds of Hertz.

Thus, it can be easily deduced that a setup of 64 EEG sensors requires bandwidth of more than 2.5 Mbps (not considering packet headers and control data), thus posing a significant burden to WSN platforms, which typically offer extremely limited resources. This has been clearly indicated in [22], where the power consumption of critical components of a WSN node has been modeled in order to study its respective effect on the overall performance of the WSN; in [23] a relative study on the lifetime of a typical WSN node is evaluated. Relevant efforts include performance evaluation concerning time sensitive applications in specific topologies [24], and various environments like industrial test cases [25] and road tunnels [26]. The net result of all these efforts is the creation of specialized communication platforms like the ones presented in [27,28]. The above brief analysis reveals the necessity of effectively reducing the amount data that must be managed. Aiming at mitigating such inefficiencies, effective compression techniques can be valuable tools able to offer significant reduction of the data wirelessly transmitted or/and stored, without compromising information accuracy. Performance efficiency, with respect to data size reduction, is measured by compression rate corresponding to the reduction percentage achieved against the initial size. Furthermore, considering that sensor data are continuously acquired, compression must be executed on-the-fly in order to minimize CPU occupation and assure zero data loss. The latter can be caused from data overrun, which usually occurs due to the limited buffers offered by typical WSN platforms.

Another aspect that highlights the significance of this effort is the potential impact it can have at commercial level considering the respective state-of-art WSN platforms. Indeed nowadays the number of WSN platforms being able to acquire mainly ECG [29–31] and to a lesser degree EEG [31] modalities rapidly increase. However, although the number of features and algorithms offered increase (e.g. encryption algorithms is a quite common example) to the best of our knowledge none of them include efficient and specialized compression algorithms. Therefore, the respective implementations, algorithms' proposals and evaluation conducted in this paper can indeed be of high value for future development platforms.

From another perspective, similar approaches have received high research interest, as it is clearly indicated by the respective papers published in relative journals and conferences [32–35]. However, such efforts offer quite diverse functional characteristics and application domain suitability. In this paper, the main axes characterizing the proposed compression approaches are the following. On one hand design adequacy with respect to low resource communication and processing platforms is required, thus requiring low complexity and high efficiency. On the other hand, the respective design must be also adequate for epilepsy monitoring, which effectively means to exhibit high efficiency regarding compression performance of the respective demanding modalities i.e. EEG/ECG. Literature research has revealed a lack of proposals satisfying both these critical requirements' axes. Therefore following an elicitation process, which is analyzed in detail in the following section, a specific group of compression algorithms that adhere better to the aforementioned requirements has been selected. A significant contribution of this paper relates to the development of all selected approaches in Matlab environment and a consequent experimentally evaluation using real EEG/ECG medical datasets. This effort allows drawing important conclusions regarding the performance and behavior comparative analysis under common a common framework. However, the main contribution of this paper focuses on proposing novel compression schemes targeting at, on optimal "compression rate"–"compression latency" trade-off as well as maximum "compression rate". The evaluation of the proposed algorithms reveals critical advantages against already existing solutions. More specifically, the proposed research work proves to be highly efficient concerning "compression rate" as well as "compression latency". Furthermore, it exhibits critical advantages with respect to achieving the optimum trade-off between considered metrics, thus advocating the used of the proposed solutions over the already existing ones on real scenarios.

Lastly, another critical contribution significantly enhancing the added value of this paper compared to similar research efforts is the utilization of a variety of real experimental EEG and ECG datasets of high sampling frequency and high resolution, so as to achieve valid, objective and practical results. Specially [32,33] although proposing lossless approaches they don't focus on EEG/ECG signals characterized by specific attributes. Also in [34,35] although they offer significant information and background knowledge, respective evaluations are based on general data omitting the specificities of biomedical signals such as EEG/ECG. The datasets have been acquired by using actual WSN sensors as well as from publicly available databases. This aspect offers a critical advantage of this work over relative ones which to the best of the authors' knowledge do not base their performance evaluation on real data. In all cases the performance evaluation focuses on the following two metrics

- Compression rate = (compressed_data_size − uncompressed_data_size)/(uncompressed_data_size).
- Compression latency indicating the time interval required to compressed a specific sample of data.

The rest of the paper is structured as follows: Section 2 presents the rationale behind the main characteristics of the compression schemes considered. Section 3 outlines the theoretical background information focusing on selected compression schemes, while Section 4 describes the proposed extensions. Section 5 presents the experimental setup, while Section 6 presents and analyses the most valuable results and measurements. Section 7 offers a comprehensive yet concise comparative analysis. Finally, Section 8 discusses the main conclusions extracted from the aforementioned measurements and provides directions concerning potential future work.

2. Rationale

A critical categorization of the compression algorithms for the intended application domain could be *lossless* or *lossy*. *Lossless* algorithms guarantee the integrity of data during the compression/decompression process. In contrast, *lossy* algorithms may result in information loss to achieve a higher compression ratio. However, when the compressed stream is decompressed the result is not identical to the original data stream as depicted in efforts like [34,33] analyzing compression algorithm functionality. With respect to this characteristic, the criticality and accuracy required by Epilepsy monitoring and study cannot tolerate any datum being corrupted as a result of the compression process. Hence, only lossless compression algorithms suitable for wireless sensors have been considered in this paper [33].

The fact that the digitized EEG and ECG signals are represented as time-series also comprised an important criterion for the selection of the compression schemes. In this context each sample, is, or can be, correlated with the previous and/or following value or values. Moreover, periodic measurements' patterns can be identified, which for example are typically observed in a healthy ECG signal. Finally, a specific and a priori range of values exists enabling the representation of all values through a specific number size (i.e. in our cases typically through 16 bit long numbers). Consequently, the algorithms selected are well known for their effectiveness on time-series datasets which is the cases of datasets encountered in compression algorithm proposal [32,33,35] or data produced in specific realistic cases like industrial scenarios [25] or analyzed in compression reference guide [34].

From the WSN networking point of view, in order to assure on the fly applicability of the compression scheme, a critical trade-off is identified between the unavoidable increased processing latency (due to demanding algorithm execution) and the desired data reduction percentage (i.e. compression ratio). Consequently, the communication efficiency offered by the respective compression approaches pertains to achieving an optimum balance point. Compression algorithms for wireless sensors must be redesigned or adapted so as to reduce the code size footprint and the dynamic memory usage [32]. In that respect, the algorithms selected focus on time series data compression and are characterized by low complexity aiming at minimizing the operation latency overhead.

3. Existing approaches

The goal of this section is to present the main characteristics of the selected compression schemes and details regarding their implementation. The main focus is on trying to extract and analyze the main functionalities and features that affect both the performance and efficiency of each particular algorithm.

3.1. Lossless compression of time-series data based on increasing average of neighboring signals

This algorithm follows a lossless compression approach aiming at predicting the values in time-series data with significant value variations, thus relying on similarities with previous data values [35]. Comparing cumulative distribution features of the current signal with the same features of the past signal, the algorithm identifies their similarities. Next, the algorithm outputs the coded residing signal (the difference between the original signal and the prediction) using a Golomb–Rice encoding and the average of neighboring signal [36]. In order to differentiate the specific algorithm from the other algorithms presented in the rest of the paper, from now on, in the respective measurements, it will be referred to as Prediction".

The method consists of four steps as described in [35]:

- Generation of the differential signal.
- Selection of the reference signal based on the cumulative distribution features.
- Generation of the residual signal.
- Golomb–Rice coding.

The implementation flow chart is depicted in Fig. 1. As shown (and will become more clear when comparing it with the other algorithms) "Prediction" scheme comprises a computationally intensive implementation. In order to achieve accurate prediction, which will assure increased compression rate, multiple demanding calculations must be made: moving average calculation of past samples, calculation distribution features, Euclidian Distance and Golomb–Rice Encoding. As it will be verified in Section 4, the above calculations result into a scheme, which on one hand poses considerable processing latency overhead while, on the other hand, achieves high compression rate efficiency due to its prediction capabilities.

3.2. An efficient lossless compression algorithm for tiny nodes of monitoring wireless sensor networks (LEC)

LEC is a low complexity lossless entropy compression algorithm [33] resulting in a very small code footprint that requires very low computational power. Its operation is effectively based on a very small dictionary and exhibits impressive on-the-fly compression capabilities. Consequently, LEC is quite attractive for being employed in WSNs. The main steps of the algorithm include:

- Calculation of the differential signal.
- Computing the difference d_i between the binary representations r_i and r_{i-1} of the current and previous measurements respectively; encoding is applied upon d_i resulting in the corresponding *bsi* sequence.
- The sequence *bsi* is then concatenated to the aggregate bit sequence.

Fig. 2a shows the implementation flow chart of the algorithm, as it was realized and evaluated in the context of this paper. The main processing effort occurs during the encoding phase of the compression, which aims at transforming d_i to *bsi* bit sequences. During this process, firstly the number n_i bits needed to encode the value of d_i is computed. Secondly, the first part of *bsi*, indicated as s_i, is generated by using the table (using n_i as index) that contains the dictionary adopted by the entropy compressor. In that respect, JPEG algorithm is adopted because the coefficients used in the JPEG have statistical characteristics similar to the measurements acquired by the sensing unit. However, in our implementation the table is extended so as to cover the necessary 16 bit resolution of the Analog to Digital Convertor (ADC) of the sensors. Thirdly, the second part of *bsi* (indicated as a_i) is calculated from the n_i low-order bits of d_i [33].

Focusing on the implementation features of LEC, it can be implemented by maintaining in memory only column s_i of the aforementioned table. Overall, it can be easily extracted that LEC avoids any computationally intensive operation, which is of paramount importance, for this application. As a result, it exhibits very low execution latency. However, since its operation is based on a static lookup table, it cannot dynamically adjust to the characteristics of a specific signal. The latter, is a significant drawback that impacts negatively on compression rate capabilities.

3.3. An adaptive lossless data compression scheme for wireless sensor networks (ALEC)

This algorithm is based on an adaptive lossless entropy compression approach focusing on low computational power. It uses three small dictionaries, the size of which is determined by the resolution of the analog-to-digital converter (ADC). Adaptive

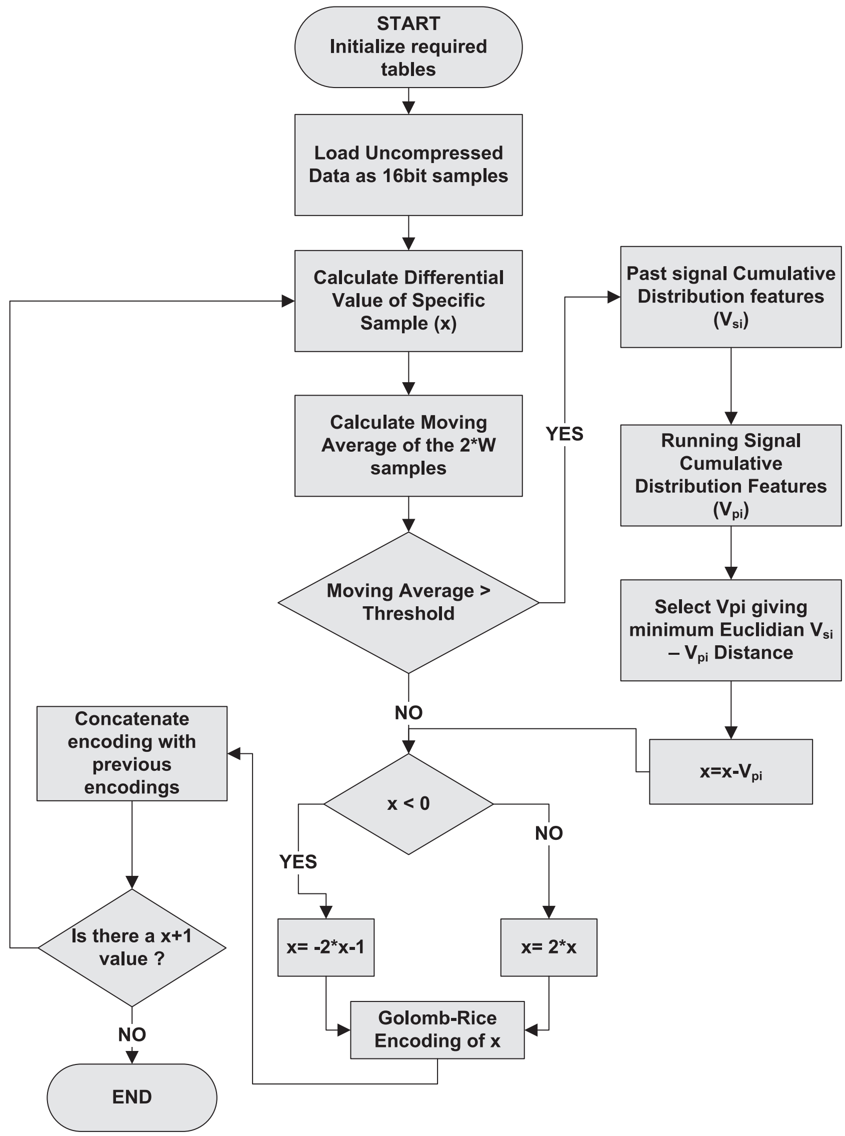
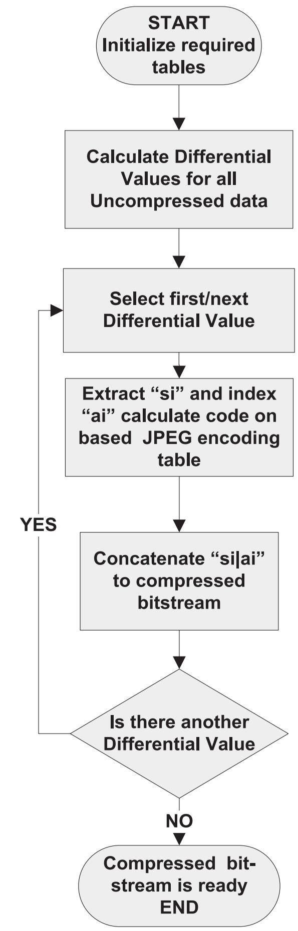


Fig. 1. Prediction algorithm [35] implementation flow chart.

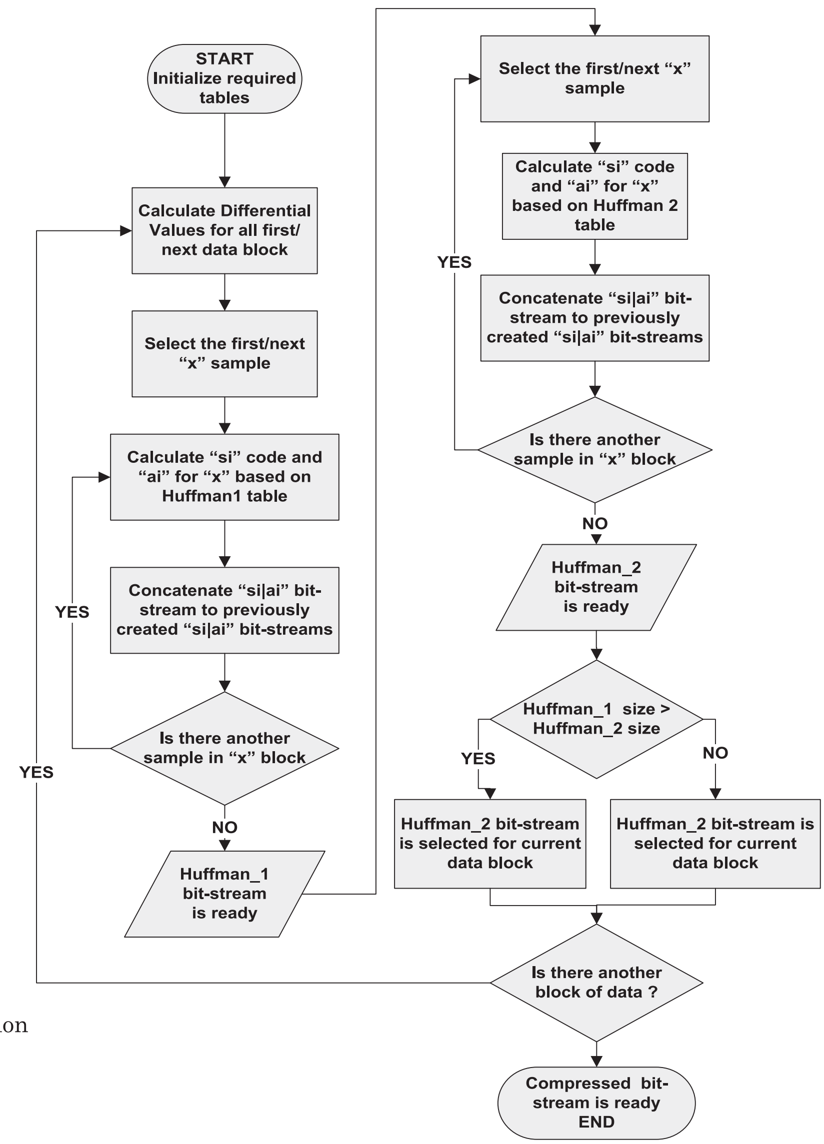
compression schemes allow the compression to dynamically adjust to the data source. The data sequences to be compressed are partitioned into blocks and for each block the optimal compression scheme is applied. The algorithm is similar to the LEC algorithm [33]. The main difference is that this algorithm uses three Huffman coding tables [32] instead of the one table used for the DC coefficients in JPEG algorithm. ALEC uses adaptively two and three Huffman tables, respectively. The implementation flow chart of ALEC algorithm is presented in Fig. 2b. Compared to LEC, ALEC is quite similar apart from the increased number of lookup tables it employs. As a result, its compression rate efficiency is also increased. However, as depicted in Fig. 3, each block is passed through two lookup tables, which eventually results in an increase of the algorithm's processing latency.

4. Proposed compression schemes

Driven by the algorithms described in Section 3, the main objective of this section is to further enhance compression efficiency. In the next paragraphs we present two novel compression schemes based on ideas, which intend to achieve (a) optimum balance between compression efficiency and processing overhead and (b) maximum compression rate efficiency. As far as the former objective is concerned, the proposed algorithms aim at offering optimum balance between the two orthogonal related performance metrics, i.e. *compression rate* and the imposed *compression processing latency*. As depicted in Sections 6 and 7, where the performance of existing approaches is evaluated, the currently available algorithms offering high compression rates tend to be very slow. On the contrary, approaches exhibiting low processing latency usually do not offer a competitive compression rate. In that respect, the idea introduced through the proposed algorithm aims at exhibiting competitive performance in both areas. As it will be analyzed in Sections 6 and 7, the proposed approach manages to offer high compression rates (similar to the maximum performance of the already published ones) without, the penalty of increased compression latency (from which high compression rate algorithms usually suffer). As a result, the proposed approach achieves a highly competitive trade-off between compression rate and compression latency compared to existing algorithms. Such performance is of paramount importance for WSN networks, especially when considering WSN node operation. This is due to the fact that in the software running on the WSN node the compression



(a) LEC Algorithm [33] Implementation Flow Chart



(b) ALEC Algorithm Implementation Flow Chart

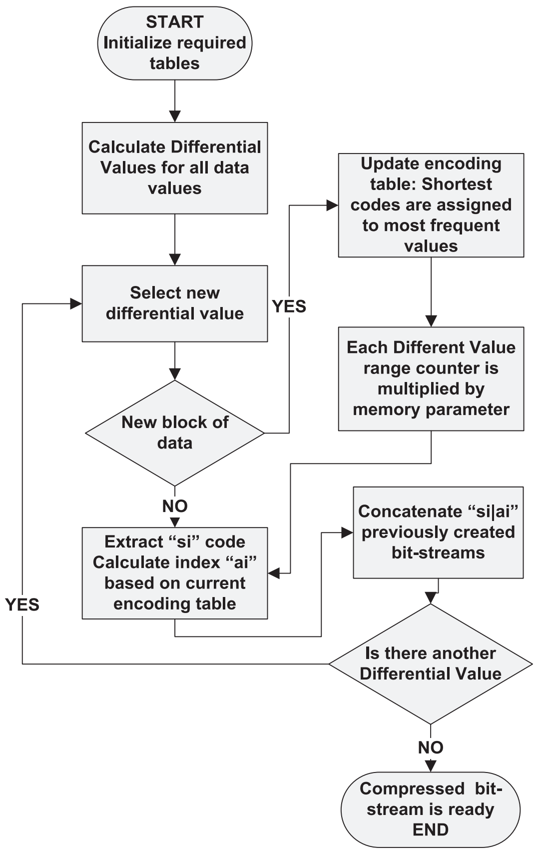
Fig. 2. Low complexity compression algorithms.

algorithms must coexist with a wide range of software related to data acquisition and data processing.

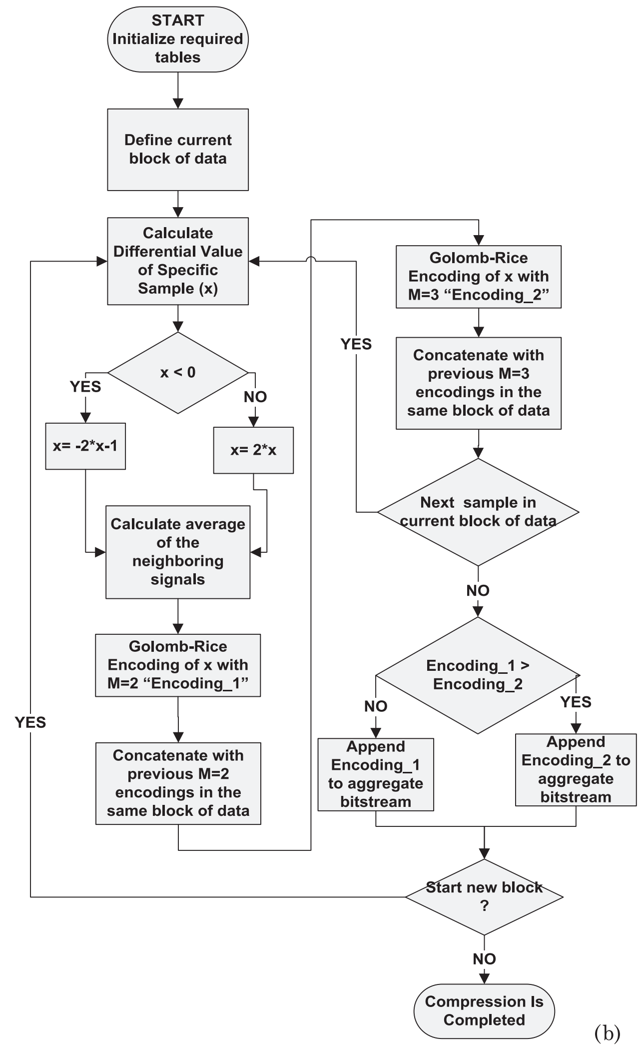
Regarding the latter objective, the novelty of the approach presented is solely related to the maximization of the *compression rate*. The proposed approach is driven by the fact that, in various scenarios, on-the-fly capability is less important compared to maximum compression rate. For example, biomedical data may be stored locally in the WSN provided memory (typically state-of-the-art WSN platforms offer storing capabilities in the order of a few GBytes). Additionally, processing latency effectively depends highly on evolving factors such as the processing capabilities of the nodes. Consequently, in various cases the end user is interested solely in reducing the space (in bytes) that the acquired biomedical data require. In order to address this aspect, a novel algorithm is proposed which achieves the highest compression rate percentage in 50% of the signals used as evaluation datasets. It is noted that even in the cases where the proposed algorithm did not exhibit the best compression rate it offered the second best performance with marginal difference from the first.

4.1. A proposed scheme for the prediction of the most suitable Huffman Code table for lossless data compression in time-series data (Real-Time Huffman)

In this section a lossless entropy compression algorithm is presented focusing on low computational requirements. It also exploits the frequencies of the observed data in the time series in order to offer an efficient dynamically adaptive Huffman table for the encoding. The main novelty (as will be analyzed in the next paragraphs) here is the extension of the approaches derived from LEC and ALEC algorithms (i.e. low complexity, low code footprint, low compression latency). Furthermore, it significantly enhances the achieved compression rate by increasing the adaptability to the data characteristics. As it will be detailed in Section 6, the goal



(a) Real-Time Huffman Scheme Flow Chart



Variable encoding scheme Flow Chart

Fig. 3. Proposed compression algorithms.

is achieved since the compression rate exhibited by the proposed extension is only marginally lower than the best achieved rate while significantly higher than LEC's and ALEC's.

4.1.1. Proposed algorithm rationale

LEC algorithm compresses data in its entropy encoder with the use of a fixed table leading to a very rapid execution. This table is an extension of the table used in the JPEG algorithm to reach the size necessary for the resolution of the ADC in use. Additionally, it is based on the fact that the closer the absolute of a value is to zero the more frequent it is observed in the differential signal. However, it is noticed that although this frequency distribution may be valid for a file or stream of data, it is not always accurate when considering fractions of the file or the data stream. Based on this observation ALEC algorithm uses a small amount of fixed Huffman Codes tables that can be alternatively used to produce smaller code for a packet of data [32]. Furthermore, the specific table is not optimal for the specific data under test at each particular experiment.

Therefore, in the proposed scheme a novel approach is introduced, where the Huffman Codes lookup tables used are continuously adjusted to the specific data used. Moreover, the degree that the tables are adjusted is also configurable, offering fine tuning capabilities that enhance the added value of the respective novel approach.

4.1.2. Utilization of data statistical knowledge

Usually no statistical knowledge is available when time-series data are measured and transmitted through wireless sensors. A method can be used based on earlier observations, but since data are changing over time this knowledge can be of questionable value as far as the compression effectiveness is concerned. Therefore, in the proposed scheme, the previously observed values' frequency, are used to update Huffman Code tables for the values to follow. Initially, the differential signal is produced, as it is highly possible to have values that can be compressed more effectively. Next, the differential signal is separated in fixed size packets. In the first packet, since there is no statistical knowledge of the data, the method is using the table from the LEC algorithm. In the subsequent packets the previous data statistical knowledge is used to create on-the-fly an adaptive Huffman Code table. The alphabet of numbers is divided into groups, the sizes of which increase exponentially. Every new differential signal value observed, results in increasing the frequency of the appropriate group. When the processing of a packet of data ends, the frequencies of each group are used to extract the probabilities of a value that belongs to that group. The blocks are sorted in descending order by their probabilities and a binary tree is created, following the procedure represented in [37].

After creating of the Huffman Code table for the subsequent packets, the current frequency table is multiplied element-wise by with a factor varying between 1 and 0. Therefore, as this parameter is approaching 0 the degree by which history (i.e. frequencies observed in previous packets) is taken into account in the next Huffman Code table diminishes. Therefore, if 0 is selected only the frequencies of the current packet are used; if 1 is selected the frequencies of every previous packet are equally used in the encoding of the next packets.

4.1.3. Analysis of the configuration features of the proposed algorithm

The method just described offers two configuration parameters. The one is the factor that regulates the degree that history frequencies are taken into account. When the respective configuration parameter is set to 1, the frequencies of all packets from the beginning of the stream are equally considered and thus the significance of the latest tendencies is reduced. In contrast, when the coefficient is set to 0 only the latest packet is considered and the older data behavior is ignored. Therefore, in the latter case the algorithm is more responsive to abrupt (or possible random) changes in the signal; the former configuration is more adequate when the overall characteristics of the signals should be considered.

The second variable requiring optimal configuration is the packet size. This parameter can also affect the performance of compression, since the increase of the packet size leads to focus on more general frequency behaviors, thus not adapting to the latest trends. However, by adopting a very small packet size the algorithm may focus too much on specific or incidental patterns of the signal ignoring general behavior tendencies, thus reducing the algorithm's effectiveness. Such parameters can be useful when fine-tuning is required in the context of the datasets (and thus application scenarios considered) or randomly compressing changing data. Regarding, the medical datasets employed in the evaluation presented in this paper, the exhibited effect on compression rate is less than 1%, which is rather negligible as far as processing latency is concerned; so they are not evaluated in more detail.

Probably the most important aspect to analyze is the implementation flow chart depicted in Fig. 3a. Compared to the LEC implementation flow chart, it can be observed that the main difference concerns the segmentation of data into packets (i.e. blocks of data). Moreover, when the process is finalized two more low complexity calculations are added so as to update the lookup table that will be used for the next packet. A counter is also maintained during the processing of each block. Although these additions slightly increase the processing latency, they also increase drastically the compression rate performance.

4.2. A proposed enhancement of the lossless compression of time-series data, based on increasing average of neighboring signals (Variable Length Golomb-Rise)

This novel proposal concerns a lossless compression scheme based on the algorithm presented previously in Section 3.1. This implementation does not predict differential signal values and encodes the residual signal as done in the original algorithm. Instead, it encodes the differential signal with the same encoding but with variable length (m) for the Golomb–Rice encoding. The implementation flow chart this scheme is presented in Fig. 3b.

4.2.1. Proposed modifications

The method commences by producing the differential signal. Then the average value of the neighboring differential signals is calculated. The first enhancement compared to the original design (Section 3.1) is that instead of estimating the differential signal and then encoding the residual (difference between estimation and differential), the differential signal itself is encoded. In this way, the method avoids the highly computationally intensive prediction procedure. Then the encoding is similar to the original method.

The second novelty, compared to the original implementation, is that instead of using a fixed length for the Golomb–Rice encoding, this method uses two different lengths for a packet of fixed length. Then it selects the output code that has the smallest length and adds one bit expressing the selection and the selected length. After experimenting, it was observed that the best-fixed lengths for the Golomb–Rice encoding of each sample are 2 and 3, which are used in our implementation. This difference is critical since Golomb–Rice encoding is executed twice (aiming at assuring the maximum compression rate). This increases the imposed latency, thus counteracting the latency reduction due to the omission of the prediction functionality (although even surpassing it in certain cases). Therefore, as it will be depicted during the comparative analysis, it achieves to surpass (even marginally) the performance of "Prediction" algorithm but with a latency penalty.

In the rest of the paper, the measurements regarding this scheme are entitled "GolombVariableM". As it will be explained in the next sections, it offers useful advantages especially when focusing at maximum compression rate performance.

5. Experimental setup

In order to offer realistic and valid performance measurements, adequate selection of evaluation datasets has been made focusing on highly demanding (in terms of amount of data and thus traffic produced for a WSN network) cases of actual biomedical data modalities related to Epilepsy. The selection of data from various sources, on one hand enhances the objectivity and validity of the evaluation, while on the other hand it allows the repeatability of the evaluation study.

Eight signals (four ECG and four EEG) with various characteristics and duration of 10 min each have been used. Each signal is initially stored in a file considering 16 bits per sample while following a little-endian storing approach. All samples have a length of 16 bits while the sampling rate is varying from 100 samples/sec (low ECG sampling rates), up to 2500 samples/sec (high EEG sampling rate scenarios). Thus a wide range of evaluation cases corresponding to respective demanding WSN traffic scenarios is offered.

5.1. PhysioNet database

PhysioNet [38] was established in 1999 as the outreach component of the Research Resource for Complex Physiologic Signals cooperative project [39]. Signals from the following two subcategories were extracted from PhysioNet database and were used as evaluation testbeds for algorithms implemented.

5.1.1. Apnea-ECG database

This database has been assembled for the PhysioNet/Computers in Cardiology Challenge 2000 [40]. From this database the ecgA04apnea and *ecgB05apnea* datasets have been used in the evaluation process.

5.1.2. CHB-MIT Scalp EEG database

This database [41], collected at the Children's Hospital Boston Massachusetts Institute of Technology (MIT), consists of EEG recordings from pediatric subjects with intractable seizures [42].

5.2. University of Patras, EEG and ECG signals

The dataset is provided by the Neurophysiology Unit, Laboratory of Physiology School of Medicine, University of Patras (UoP) [43].

5.3. EkgMove ECG signals

EkgMove [44] is a psycho-physiological measurement system for research applications. From the various measurement capabilities of the sensor, the ECG signal of a single subject has been used [45].

5.4. Experiment configuration

Measurements were conducted in a 32-bit IBM compatible computer equipped with an Intel Pentium Dual CPU T3200 @ 2.00 GHz and with sufficient memory to load the signal sample in the RAM memory. All measurements were executed in Matlab environment where all algorithms presented were implemented. It is noted that during the implementation phase of the selected algorithms, effort has been devoted to ensure that the used software functions can be implemented in embedded systems. For example, instead of using hash tables and hash sets, only normal lookup tables have been used; hash tables could enhance execution speed, but their implementation is not frequently encountered in most embedded systems.

6. Perfomance evaluation

In this section an in depth performance and behavioral analysis is presented aiming at extracting valuable and important conclusions. To achieve this objective a multi-parametric evaluation is undertaken considering the characteristics presented in Table 1.

Furthermore, in order to objectively reveal all the behavioral characteristics of compression, the respective performance graphs are not only compared to each other but are also studied and analyzed with respect to the specific dataset's sample value and especially the observed variation in them.

6.1. The effect of compression algorithm

As indicated, the evaluation is based on different datasets characterized by diverse attributes. Consequently, the main objective in this subsection is to identify which technique performs better for each particular dataset quantitatively but also reveal the specific characteristics of the datasets affecting the behavior of the compression scheme qualitatively. On one hand it is observed that the absolute values of the signals considered are completely different making it very hard to extract a common way to represent and compare them. On the other hand, all compression schemes considered base their logic on the difference between a particular value and the previous one. Therefore, in order to have a common and objective way to evaluate the effect of the actual signal being compressed, the differential signal is used as the main representation of the actual signal being compressed.

Table 1
Evaluation scenarios' configuration.

Configuration parameter	# of Values	Details–comments
Compression algorithms considered	5	(3 existing and 2 proposed)
Epilepsy modalities considered	2	EEG, ECG
Sampling frequencies	3	100 (Apnea datasets), 245 (MIT, EkgMove datasets), 2500 Hz (UoP datasets)
Sample resolution	2	12, 16 bit
Datasets' signal duration	10 min	–

6.1.1. Electrocardiography (ECG) Apnea A04 dataset

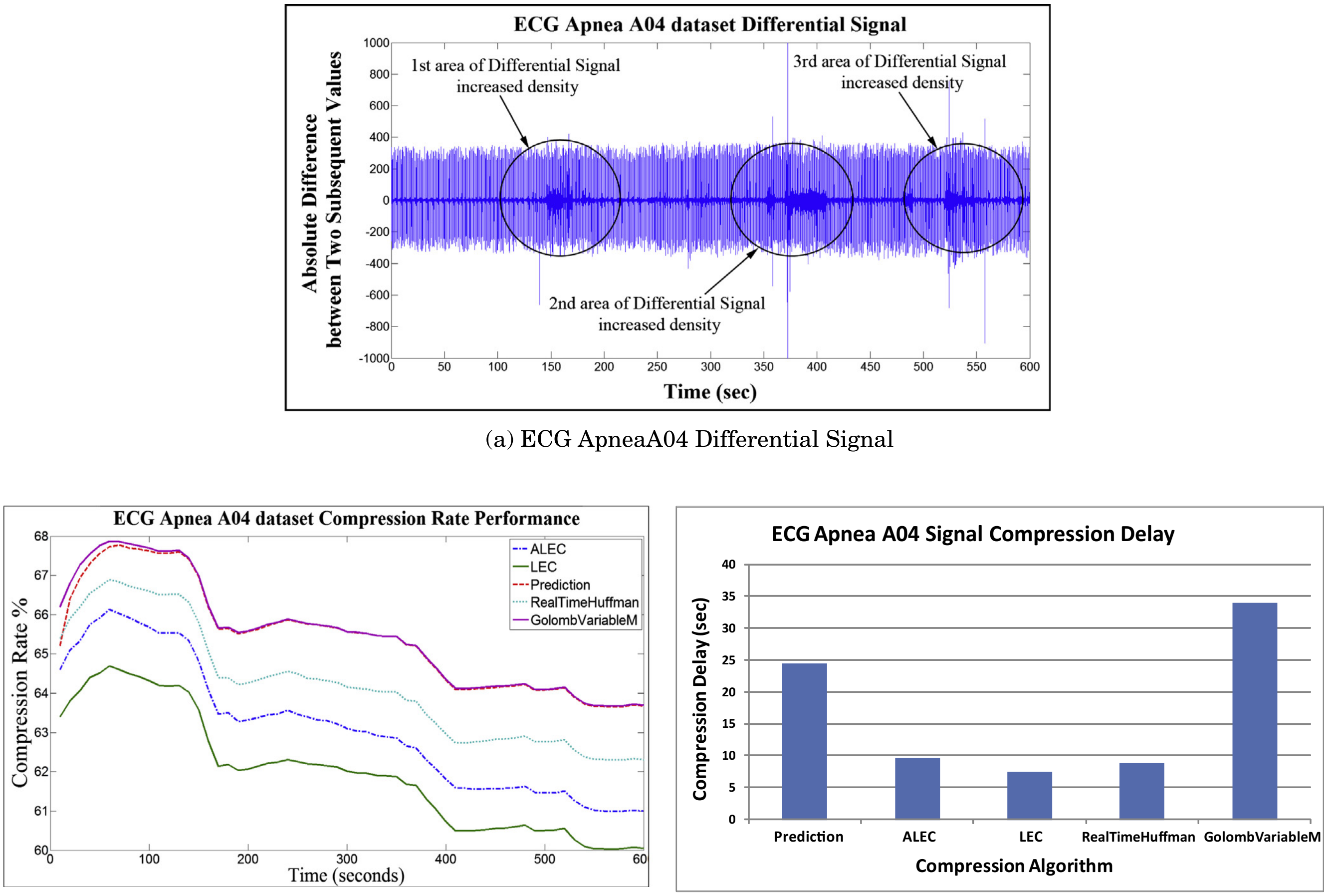
The datasets extracted from the Apnea database concern ECG signals sampled with the lowest frequency among the datasets considered. In Fig. 4a the differential signal is presented indicating the differences between each particular sample and the previous one. As observed, the variation of the absolute values are within the range of −300 up to 300 while the density of the graph is quite homogeneous, except three areas (depicted in Fig. 4a) where the variation density is higher; as it will be shown this has a clear effect on the compression behavior.

Comparing Fig. 4a and b it is evident that each area of increased differential signal density corresponds to a clear drop of compression rate performance. It is noted that this behavior is common to all algorithms considered.

Furthermore, a clear advantage is offered by the "Prediction" and "Variable Length Golomb-Rise". On one hand, the "Pre diction's" advantage is attributed to the fact that heart rate (acquired by the ECG measurements) is a periodic operation thus prediction comes into play. On the other hand, "Variable Length Golomb-Rise" advantages are based on the fact that data are effectively compressed twice and the configuration offering the best compression rate is selected. Both algorithms exhibit an average effective data volume reduction (i.e. compression rate) of ~64% while reaching up to ~68% in particular segments of the dataset. However, both approaches are based on intense mathematical processes, thus significant latency penalty must be paid, as shown in Fig. 4c. The proposed Real-Time Huffman also offers a significant 62.5% compression rate; LEC algorithm presents the lowest performance of 60%, from which the network can still benefit regarding performance and robustness.

One last important observation concerns the fact that the second proposed scheme "Real Time Huffman" proves to be the best next option lacking only ~1–1.5% compression rate compared to the previous ones. It offers significant advantage in terms of latency as shown in Fig. 4c where the compression latency of all algorithms considered is depicted. In conjunction with Fig. 4b and the characteristics of the signal (i.e. the sampling frequency) valuable observations can be made. It is noted that the dataset sample duration is 600 sec. Therefore it can be extracted that for low sampling rate of 100 Hz, the compression for any of the considered algorithms can be performed on-the-fly since the maximum latency measured is only ~35 sec, i.e. 5.8% of the time required to acquire the measurements.

Moving on to a comparative analysis, it is proven that the maximum compression rate comes with a penalty since both "Prediction" and "Variable Length Golomb-Rise" exhibited the highest latency i.e. ~25 and ~35 sec respectively. In contrast, the proposed "Real Time Huffman" scheme is lacking only 1–1.5% of the compression rate performance, while it offers a highly competitive latency performance of ~8 sec representing only 1.3% of the time required for the sensor to acquire the actual data. Quantifying the real-time capabilities of the considered algorithms shows that all algorithms can compress the dataset under evaluation in real-time. This is because the compressed signal (of all datasets considered) represents data acquired over a period of 10 min. Therefore, the data acquired during a process of 600 sec can be compressed in 7 sec (LEC), 8 sec (proposed Real-Time Huffman) or a maximum of 35 sec, which can be easily accommodated using a double buffer logic (one buffer receiving the uncompressed data and a second one compressing data received during the previous period). A



(a) ECG ApneaA04 Differential Signal

(b) Compression Rate of ECG Apnea A04 Dataset

(c) Compression Latency of ECG Apnea A04 Dataset

Fig. 4. Compression performance on ECG Apnea A04 differential signal.

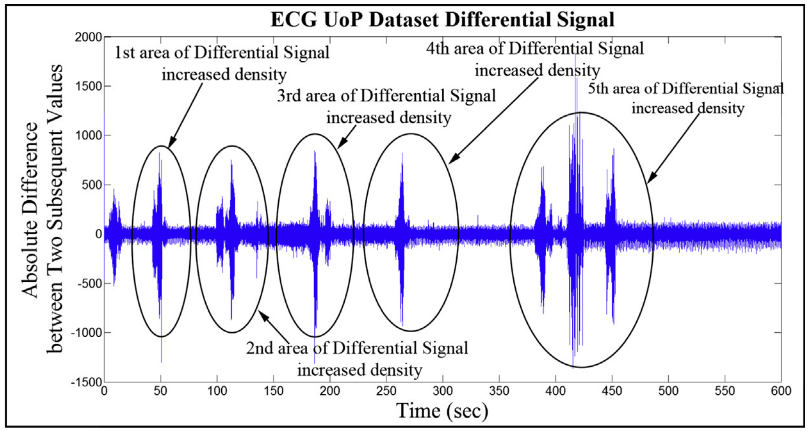
second, yet more important comment, has to do with the fact that a respective component on a real WSN node will have to accommodate multiple similar signals, for example due to the need to concurrently monitoring multiple such ECG sensors. Therefore a sense of multiple sensor capacity is extracted where algorithms like the proposed Real-Time Huffman achieve a metric of 75 (dividing the 600 sec acquisition period by 8 sec compression latency). The performance achieved is more than three times higher compared to Prediction algorithm that is able to compress 24 analogous signals due to the exhibited 25 sec latency.

6.1.2. Electrocardiography (ECG) UoP

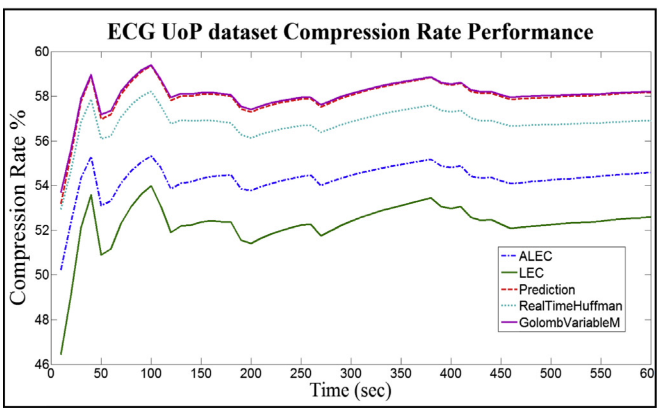
Datasets from University of Patras, Greece (UoP) comprise the most demanding cases considered in the context of this paper. Firstly, the sampling frequency is 2500 Hz representing an emphatic increase compared to 256 Hz and 100 Hz analyzed so far. Secondly, and with respect to Fig. 5a, the differential signal is characterized by high number of areas where the differential signal's range is drastically higher than the previous cases.

These characteristics lead to an overall reduced compression rate performance as indicated in Fig. 5b since they range between ~52% and 58% corresponding to "LEC" and "Prediction"–"Golomb VariableM" exhibited performance respectively. It is interesting to note that between the 4th and the 5th marked area, where the differential signal is homogeneous, all compression rate graphs exhibit increasing tendencies, which are followed by three decreasing slops corresponding to the incidences in the 5th marked area of Fig. 5a. What is very important to point out in favor of the proposed "RealTimeHuffman" scheme is that in such demanding dataset cases the advantages of the highly complicated "Prediction" and "GolombVariableM" are significantly diminished. As observed throughout the dataset duration, the performance of "RealTimeHuffam" is only 1% less than the optimum achieved. Furthermore, and in comparison to the previous cases, the proposed "RealTimeHuffman" clearly outperforms "ALEC" which offered analogous compression rate, in the less challenging datasets. Overall, the proposed GolombVariableM and Prediction offer the highest data volume reduction of ~58%. However the second proposed algorithm exhibits an impressive 57% compression rate the added value of which is even more pronounced when considered in conjunction with compression latency evaluation following. Furthermore, LEC performance lacks considerably marginally exceeding 52% compression rate.

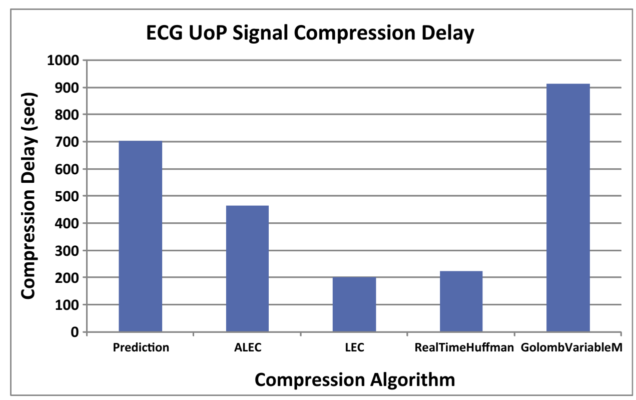
As expected, due to the high sampling frequency, compression latency becomes a critical metric when on-the-fly compression is required e.g. real-time monitoring and epilepsy incident identification. The two schemes that could effectively fulfill the demands of an on-line scenario are "LEC" and "RealTimeHuffman", both imposing a latency compression of ~200 sec. Concerning real-time capabilities critical observations can be made. The first clear observation is that in contrast to the previous dataset, in this case two out of the five implemented algorithms are inadequate for on the fly operation since compression latency requires more time than the acquisition time period of 600 sec. These are the Prediction algorithms exhibiting a processing latency of 700 sec and the GolombVariable M requiring up to 900 sec to compress data aggregated over the 600 sec period. The two algorithms clearly able to accommodate real time operation are LEC and Real-Time



(a) ECG UoP Dataset Differential Signal



(b) Compression Rate of ECG UoP Dataset



(c) Compression Latency of ECG UoP Dataset

Fig. 5. Compression performance on ECG UoP differential signal.

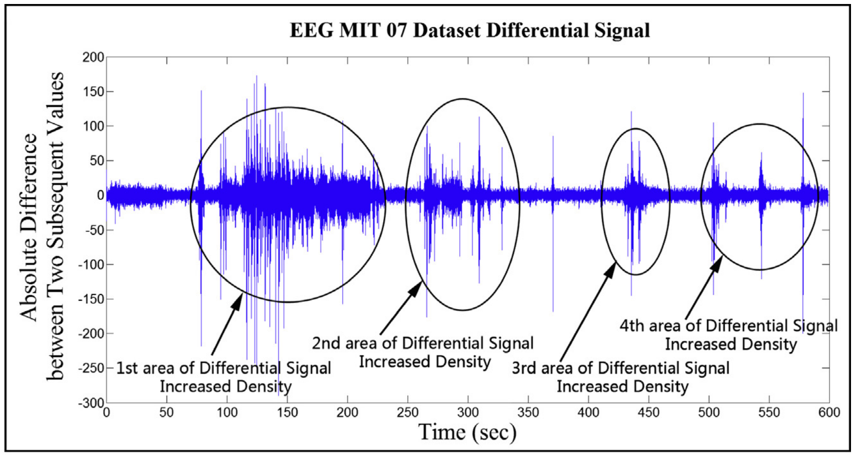
Huffman exhibiting 200 and 224 sec latency respectively. Even more real-time operation based on these algorithms could be supported for concurrently monitoring respective 3 and 2.5 ECG sensors.

6.1.3. *Electroencephalography (EEG) from MIT 07 dataset*

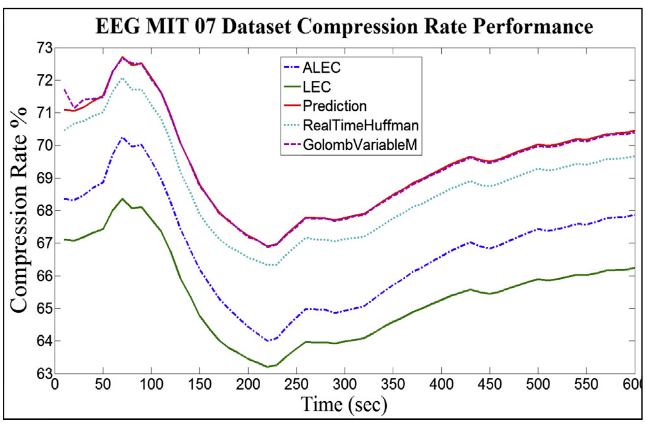
This dataset comprises of EEG signal sampled with a frequency of 256 Hz. The signal outside the marked areas is quite smooth, as indicated by the very small range of differentiated values (that fall within the range between −50 and 50), which can lead to high compression rate. However, throughout the sample considered, multiple areas of relatively high differential values are identified in Fig. 6a, which affect negatively the compression efficiency.

The above characteristics have a clear effect on the compression behavior as presented in the respective graphs in Fig. 6b. As depicted at the beginning of the process, the relatively small differential signal leads to a rapid increase of the compression rate. The latter, is interrupted by the first marked area which concerns a quite extended time period of increased differential signal; this has a drastic negative effect on all algorithms since the compression rate drops by 5–6% in the interval between the 75th sec and 225th sec. Following this significant performance degradation, the compression rate increases, although temporarily obstructed by subsequent areas of increased differential signal. Overall, the compression rate increase (time duration 225th up to 600th sec) compensates for most of the performance degradation attributed to the first area depicted in Fig. 6a. This is depicted by the fact that, in all compression algorithm cases, the final compression rate (achieved at the 600th sec) is less than 1% lower than the initial one (indicated at the first 10 sec of the compression).

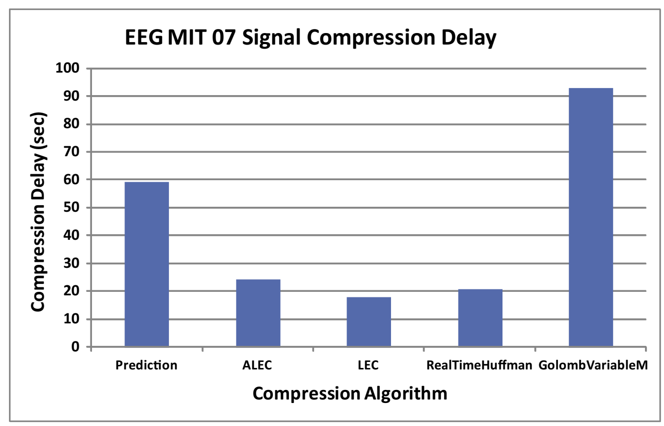
Furthermore, the EEG signal is less periodic compared to ECG and does not favor “Prediction” and “GolombVariableM” approaches. Therefore, from a quantitative aspect although these two algorithms still offer the highest absolute performance (70.3% compression rate), it is less than 1% higher than the proposed “RealTimeHuffman” (69.8%). At the same time, the gap between the latter and LEC increases since the latter manages a relatively mediocre 66.2% data volume reduction. Combining the previous observations with the latency overheads depicted in Fig. 6c, it can be observed that the “RealTimeHuffan” approach, offers less than 1% reduced compression efficiency compared to “Prediction” and “GolombVariableM”, while posing only 1/3 and 1/4.5 respective latency. Such an advantage can be of critical importance when on-the-fly compression is required. Following an approach similar to the previous cases for quantifying real-time capabilities as in previous cases it has been observed that all algorithms are able to compress the received data much faster than actually acquiring them, thus adequate for on-the-fly scenarios. However, once more the proposed “Real-Time Huffman” and “LEC” algorithms exhibit advanced real-time capabilities which allow compressing aggregated data acquired from approximately 30 EEG sensors in the period of 600 sec. Algorithms like “Prediction” and “GolombVariableM” can respectively accommodate ∼10 and ∼6.5 similar sensors which is a significant disadvantage regarding the specific performance metric.



(a) EEG MIT 07 Dataset Differential Signal



(b) Compression Rate of EEG MIT 07 Dataset



(c) Compression Latency of EEG MIT 07 Dataset

Fig. 6. Compression performance on EEG MIT 07 differential signal.

6.1.4. Electroencephalography (EEG) from the F4 sensor UoP dataset

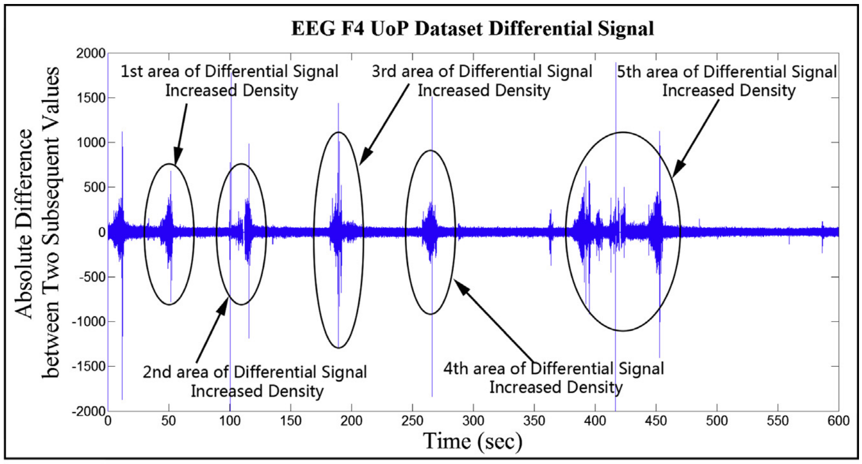
As observed with ECG signals, UoP datasets offered challenging EEG cases as well. Specifically, as depicted in Fig. 7a, there are multiple areas of abrupt and emphatic variation increase of the differential signal posing significant obstacles upon compression schemes. It is interesting to note that these differences reach and surpass 500 of absolute differential signal and in several cases go beyond 1000.

Additionally, between these areas the differential signal varies within a very limited range of values, which, as observed in the previous cases, leads to compression graphs exhibiting an increasing gradient. A critical observation is that the advantage offered by highly complex algorithms like "Prediction" and "GolombVariableM" is, in such cases, negligible compared to significantly less processing demanding algorithms like "RealTimeHuffman" (Fig. 7b); thus, all three algorithms exhibiting a ~59% compression rate. Furthermore, as indicated in Fig. 7c, "RealTimehuffman" posses a processing latency less than 1/3 of the respective performance exhibited by "Prediction" algorithm. Similarly, the latency of "RealTimehuffman" is less than 1/4 of the performance of "GolombVariableM". Even more, "Prediction" and "GolombVariableM" cannot be considered for real time scenarios requiring on-the-fly compression since the processing latency is significantly higher compared to the period required to acquire the signal value. Finally focusing on less efficient data reduction algorithms, LEC manages to significantly reduce the size of data by ~55% while requiring only a very low processing latency of less than 200 sec.

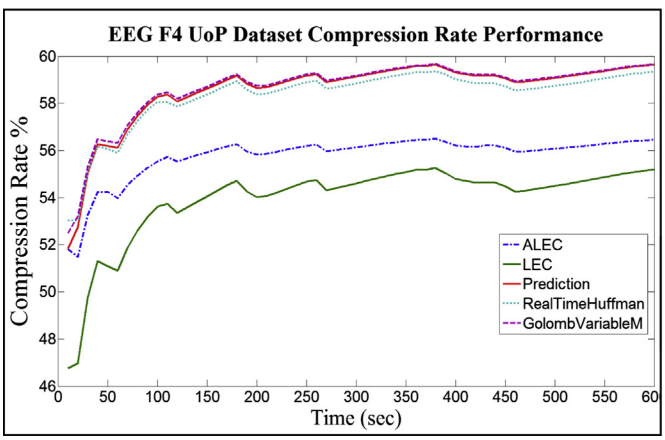
6.2. The effect of dataset on compression scheme behavior

As observed in the context of this evaluation, the relative effect of the datasets considered on specific algorithm performance and behavior is similar for all considered algorithms evaluated. Therefore only the case of the proposed "RealTimeHuffman" algorithm will be analyzed.

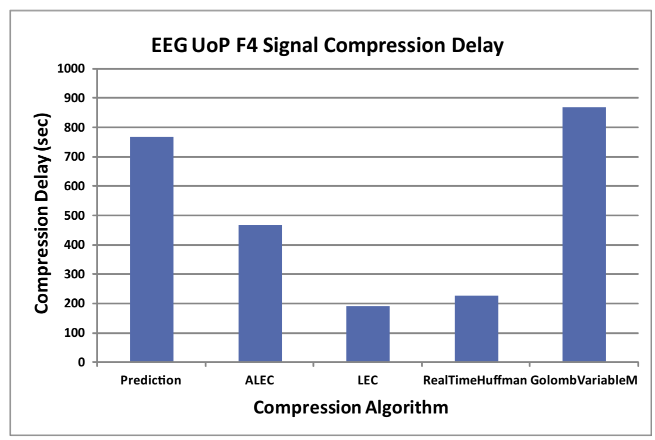
Based on Fig. 8 and taking into account the characteristics of the datasets employed, the variation of the differential signal's values is by far the most critical parameter as far compression rate is concerned. Therefore, in all cases EkgMove based dataset offers the highest compression efficiency (surpassing 70%) while UoP's datasets (regardless of the signal type) proved to be the less compression-prone cases occupying the last three places (Fig. 8) with compression rate ranging between 55% and 60%. Additionally, Apnea ECG and MIT EEG based datasets exhibited a relatively minor performance variation for different compression schemes. Furthermore, the modality of the signals (i.e. EEG and ECG) and the source of the signal (EkgMove sensor, Apnea database, MIT databased and UoP databased) also did not prove to be significant factors that favor the use of a specific scheme when a specific dataset is used.



(a) EEG UoP F4 Dataset Differential Signal



(b) Compression Rate of EEG UoP F4 Dataset



(c) Compression Latency of EEG UoP F4 Dataset

Fig. 7. Compression performance on EEG UoP F4 differential signal.

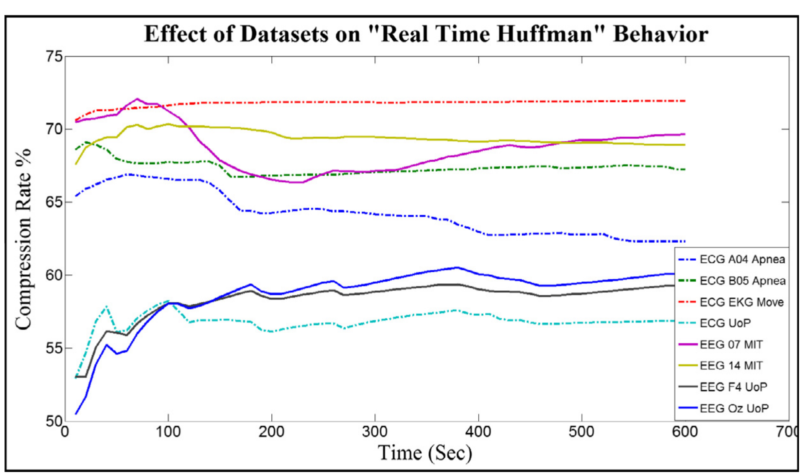


Fig. 8. Signal effect on "Real Time Huffman" performance.

Regarding the processing latency, the clearly dominating factor is the sampling frequency selected. In all compression cases, moving from Apnea based datasets (characterized by the lowest sampling frequency, i.e. 100 Hz) to EkgMove and MIT based datasets with 256 Hz sampling frequency, and to UoP datasets having the highest sampling frequency of 2500 Hz, the respective compression latency increases proportionally to the sampling frequency. Once again real-time quantification offers valuable insight, indicating that only proposed "Real-Time Huffman" and "LEC" algorithms are able to compress data rapidly enough so that multiple sensors can be monitored (2.5 and 3 respectively). "ALEC" is able to host data acquired only from one sensor, "Prediction" and "GolombVariableM" are inadequate for real-time operation since their compression latency clearly surpasses the upper limit of 600 sec.

7. Comparative analysis

In this section, a comparison among the evaluated algorithms is presented in order to extract useful conclusions and practical guidelines regarding the efficiency and the degree of on-the-fly applicability of each algorithm. Respective comparative analysis commences with Table 2 where the algorithm offering the smallest compression latency is highlighted and the respective cell indicates the absolute latency value. The rest of the values indicate the percentile processing latency overhead with respect to the optimal one. In this way, Table 2 not only presents the algorithm ranking in terms of speed in a qualitative manner, but provides also quantitative analysis on how much slower one algorithm is from another in a particular dataset.

The first observation is that "LEC" poses the minimum processing demands in all cases, comprising the most suitable solution for on-the-fly scenarios. A critical observation is that the second "faster" algorithm is the proposed "RealTimeHuffman". On one hand, this overhead advocates the use of "RealTimeHuffman" solution in real time demanding scenarios as quantified in previous section. On the other hand, the difference between the second (i.e. RealTimeHuffman) and the third best performance (i.e. "ALEC") is significantly higher compared to the difference between "LEC" and the proposed scheme. Specifically, in Apnea, EkgMove and MIT datasets the overhead imposed by "ALEC" compared to the proposed scheme vary between 7.5% and 20%. When focusing on UoP datasets is increases in the area of 120%, which as indicated in previous analysis, invalidates the minimum requirements for on-the-fly scenarios. This shows that the proposed scheme is clearly the second best solution that is very close to the algorithms with the best performance ("LEC"). The other two algorithms exhibit very high latency overheads.

However, the added value of novel "RealTimeHuffman" algorithm is further enhanced through Table 3, where the comparative performance of the compression rate is presented. The first interesting observation concerns the fact that the two algorithms posing the most significant latency overheads (and the most complex implementations) are the ones offering the highest compression rates. Specifically, in the less compression prone datasets of UoP, the proposed variation "GolombVariable" exhibited the best performance as well as in one dataset of Apnea, while in all the other scenarios, "Prediction" yielded the optimum performance. However, both algorithms exhibited excessive latency overheads, thus not being recommended for on the fly scenarios, which are critical when WSN deployment is considered. The most important observation is that the following best solution, in all cases, is the proposed "RealTimeHuffman" scheme exhibiting minimum compression rate penalty (between 0.5% and 3.5%).

Overall this performance, in conjunction with the highly competitive latency performance, raises the proposed "RealTimeHuffman" scheme as the most adequate solution when the optimal trade-off is the main concern.

As a last step, the compression rate of "RealTimeHuffman" algorithm has been compared against three of the most widely used, general purpose compression algorithms. ZIP [46] and RAR [47] as part of WinRAR software, and LZO [48]. Throughout the evaluation, EEG 07 MIT, ECG EkgMove and ECG UoP dataset were used. In order to allow a fair comparison between the four candidates, the performance evaluation focuses on the compression rate achieved rather the processing latency. This is because the proposed algorithm is executed in Matlab environment (as applied for all algorithms implemented in the context of this effort) whereas ZIP, RAR and LZO are commercial applications, thus a compression latency comparison would have be meaningless due to the overheads imposed by Matlab environment.

As indicated in Table 4 in almost all cases the proposed algorithm offers higher compression rate compared to the state-of-the-art compression algorithms. It is noted that RAR compression managed to offer higher compression rate in two cases. However, the complexity and memory requirements of RAR application comprise negative factors when considering utilization in the context of WSN networks. LZO, on the other hand, although offered very fast execution and low complexity, leads to a significantly reduced compression rate performance with respect to the specific type of data. Finally, ZIP also exhibits reduced

Table 4
Compression rate performance comparison.

	EEG 07 MIT (%)	Ecg EkgMove (%)	Ecg UoP (%)
RealTime Huffman	69.7	71.8	56.9
ZIP	56.2	57.8	28.3
RAR	64.6	72.0	61.3
LZO	34.7	35.9	8.9

Table 2
Comparative latency performance of compression algorithms.

	ecgA04apnea	ecgB05apnea	ecgEkgMove	ecgUoP	EEG07 MIT	EEG14 MIT	EEG F4 UoP	EEG Oz UoP
Prediction	+227.51%	+235.99%	+258.36%	+248.6%	+228.00%	+239.27%	+300.53%	+254.2%
ALEC	+30.05%	+35.83%	+35.1%	+131.75%	+34.19%	+38.04%	+144.49%	+124.24%
LEC	7.50	6.78	16.82	201.69	18.14	17.85	191.92	198.17
RealTimeHuffman	+18.69%	+28.38%	+26.09%	+11.42%	+14.6%	+23.63%	+18.06%	+10.56%
GolombVariableM	+354.32%	+373.27%	+398.01%	+354.1%	+412.58%	+402.37%	+353.44%	+352.09%

The gray shaded values are the best values recorded for the specific dataset and also serves as reference point for the rest of the measurements produced by different algorithms and implementations.

Table 3
Comparative compression rate performance of compression algorithm.

	ecgA04apnea	ecgB05apnea	ecgEkgMove	ecgUoP	EEG07 MIT	EEG14 MIT	EEG F4 UoP	EEG Oz UoP
Prediction	−0.01%	69.68	73.58	−0.05%	70.45	70.33	−0.03%	−0.11%
ALEC	−4.22%	−3.6%	−2.32%	−6.2%	−3.65%	−5.15%	−5.36%	−6.64%
LEC	−5.69%	−5.78%	−6.25%	−9.65%	−5.96%	−6.57%	−7.48%	−7.92%
RealTimeHuffman	−2.17%	−3.5%	−2.23%	−2.23%	−1.09%	−2.02%	−0.52%	−0.78
GolombVariableM	63.69	−0.11%	−0.1%	58.2	−0.07%	−0.09%	59.66	60.64

The gray shaded values are the best values recorded for the specific dataset and also serves as reference point for the rest of the measurements produced by different algorithms and implementations.

compression rate performance, against the proposed algorithms, while it is also characterized by a relatively high complexity making it unsuitable choice of WSN networks.

8. Conclusions

The advent of the era for Internet-of-Things, where devices of any kind and with various resources, usually scarce, are interconnected and exchange critical data, has created new challenges for WSNs, which appear to be the dominant technology for device interconnection. Despite the fact that most of the devices are rather limited CPU power, memory and storage capabilities, the applications that take advantage of them pose significant pressure as far their efficiency is concerned. This has led the designers to develop novel algorithms that will allow efficient use of the limited device resources in an attempt to deal with critical requirements such as realtimeness and data transmission efficiency.

In order to deal with the aforementioned challenges, the work presented here introduced a set of novel compression algorithms for real time transmission of medical data. The scenarios considered have been borrowed from one of the most demanding areas of medical research, the real time monitoring of epileptic patients. The algorithms presented have been developed and tested using real world, demanding EEG and ECG datasets acquired from different sources. The proposed schemes exhibited significant advantages over the competition including general purpose compression applications. Specifically "Real-Time Huffman" presented algorithm exhibited the best trade-off performance offering 50–70% data volume reduction in combination with minimum latency overhead. Furthermore, measurements' analysis highlighted sampling rate as the main factors affecting compression latency and the differential signal variations as the most important compression rate parameter.

The research work presented intends to pave the way for a new type of devices that will be part of a world of Internet-of-Things and that will be able to use their limited resources in a more efficient way. As a next step in this direction, the proposed algorithms will be implemented, using state-of-the-art hardware platforms, as embedded devices into the next generation of smart sensors/devices that will be able to adapt their transmission characteristics to their communication needs and their available resources.

Conflict of interest

The authors declare that there are no conflicts of interest.

Acknowledgment

This study is part of the collaborative project ARMOR which is partially funded by the European Commission under the Seventh Framework Programme (FP7/2007-2013) with Grant Agreement: 287720.

References

[1] H.M. de Boer et al., ILAE/IBE/WHO Global Campaign Against Epilepsy: a Partnership that Works, CurrOpin Neurol.
[2] M.P. Kerr, The impact of epilepsy on patients' lives, Acta Neurol. Scand. Suppl. 194 (2012) 1–9, http://dx.doi.org/10.1111/ane.12014.
[3] C.R. Newton, H.H. Garcia, Epilepsy in poor regions of the world, Lancet 380 (9848) (2012) 1193–1201.
[4] J. Olesen, The economic cost of brain disorders in Europe, Eur. J. Neurol. 19 (2012) 155–162.
[5] P. Panayiotopoulos, A Clinical Guide to Epileptic Syndromes and Their Treatment, Book by Springer 2010.
[6] A. Strzelczyk, Cost of epilepsy: a systematic review, Pharmacoeconomics 26 (6) (2008) 463–476.
[7] ARMOR FP7 Project: <http://armor.tesyd.teimes.gr/>.
[8] Chiron EU Project: <http://www.chiron-project.eu/>.
[9] Hugo Furtado et al., Applications of wireless sensors in medicine, 34th MIPRO, 23–27 May 2011, Vienna, Austria.
[10] N. Dessart et. al., Simulation of large scale WSN for medical care, IEEE ISCC 2010, 22–25 June 2010, Riccione, Italy.
[11] Yeongjoon Gil, A wireless EEG monitor system based on BSN node, CBMS 2011, 27–30 June 2011, Bristol.
[12] E. Lawrence et al., Data Collection, Correlation and Dissemination of Medical Sensor Information in a WSN, ICNS '09, 20–25 April 2009, Valencia Spain.
[13] AANT&TAC: American Academy of Neurology Therapeutics and Technology Assessment Committee. Assessment Intensive Monitoring for epilepsy. Neurology 39(8), 1101–1102 (1989 reaffirmed 1997).
[14] A.T. Berg, S. Shinnar, The risk of seizure recurrence following a first unprovoked seizure: a quantitative review, Neurology 41 (1991) 965–972.
[15] D.E. Crompton, S.F. Berkovic, The borderland of epilepsy: clinical and molecular features of phenomena that mimic epileptic seizures, Lancet Neurol. 8 (2009) 370–381.
[16] Epilepsia FP7 Project: <http://www.epilepsiae.eu/>.
[17] S. Farshchi, A TinyOS-enabled MICA2-based wireless neural interface, IEEE Trans. Biomed. Eng. 53 (2006).
[18] S. Farshchi, A TinyOS-based wireless neural interface, in: IEMBS '04. 26th Annual International Conference of the IEEE, vol. 2, 1–5 September 2004, San Francisco, CA.
[19] R.N. Henson et al., MEG and EEG data fusion: simultaneous localization of face-evoked responses, NeuroImage 47 (2009) 581–589.
[20] B. Huyghe et al., Design of flexible, low-power and wireless sensor nodes for human posture tracking aiding epileptic seizure detection, IEEE Sensor 2009, 25–28 October 2009, Christchurch, New Zealand.
[21] Ming-Feng Wu, The design of wireless sleep EEG measurement system with asynchronous pervasive sensing, in: IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009. 11–14 October 2009, San Antonio, TXs.
[22] Ch. Antonopoulos et al., Experimental evaluation of a WSN platform power consumption, 23rd IPDPS May, 2009 Rome Italy.
[23] Isabel Dietrich et al., On the lifetime of wireless sensor networks, ACM TOSN 5 (1) (2009). February.
[24] Petr Jurcik et al., Dimensioning and worst-case analysis of cluster-tree sensor networks, ACM TOSN 7 (2) (2010). August.
[25] K.S. Low et al., Wireless sensor networks for industrial environments, in: CIMCA, 2005, vol. 2, 28–30 November 2005, pp. 271–276.
[26] Luca Mottola, Not all wireless sensor networks are created equal: a comparative study on tunnels, ACM (TOSN) 7 (2) (2010). August.
[27] J. Polastre et al., The mote revolution: low power wireless sensor network devices, in: Hot Chips, August 2004.
[28] Telos 2004, Telos: Ultra low power IEEE 802.15.4 compliant wireless sensor module, Moteiv Corporation, 2004, revision B. [Online].
[29] Shimmer WSN Platform: <http://www.shimmersensing.com/>.
[30] e-Health WSN Platform: <http://www.cooking-hacks.com/documentation/tutorials/ehealth-biometric-sensor-platform-arduino-raspberry-pi-medical>.
[31] openBCI WSN platform: <http://www.openbci.com/>.
[32] J.G. Kolo et al., An adaptive lossless data compression scheme for wireless sensor networks, J. Sens. 2012 (2012).
[33] F. Marcelloni et al., An efficient lossless compression algorithm for tiny nodes of monitoring wireless sensor networks, Comput. J. (Advance Access published April 30, 2009).
[34] D. Salomon, Data Compression: The Complete Reference, fourth ed., Springer, London, UK, 2006.
[35] T. Tetsuya et al., Lossless compression of time-series data based on increasing average of neighboring signals, Electron. Commun. Jpn. 93 (8) (2010).
[36] S.W. Golomb et al., Run-length encodings, IEEE Trans. Inf. Theory 12 (2003) 339–401.
[37] Huffman Tables, <http://en.wikipedia.org/wiki/File:Huffman_coding_example.svg>.
[38] A.L. Goldberger et al., Components of a new research resource for complex physiologic signals, Circulation 101 (23) (2000) e215–e220.
[39] Physionet, <http://www.physionet.org/>.
[40] T. Penzel et al., The Apnea-ECG database, Comput. Cardiol. 27 (2000) 255–258.
[41] Ali Shoeb, Application of Machine Learning to Epileptic Seizure Onset Detection and Treatment, PhD Thesis, Massachusetts Institute of Technology, September 2009.
[42] Physiobank, <http://www.physionet.org/physiobank/database/chbmit/>.
[43] V. Kokkinos, G.K. Kostopoulos, Human non-rapid eye movement stage II sleep spindles are blocked upon spontaneous K-complex coincidence and resume as higher frequency spindles afterwards, J. Sleep Res. 20 (1pt1) (2011).
[44] EkgMove, <http://www.movisens.com/images/stories/download/doc/EKGMove_Bedienungsanleitung.pdf>.
[45] Movisens, <http://www.movisens.com/en/products/ecg-and-activitysensor>.
[46] PKWARE Inc. 2012, APPNOTE – .ZIP File Format Specification, Version 6.3.3, September 2012.
[47] RARLAB, RAR 5.0 Archive Format, <http://www.rarlab.com/technote.htm>.
[48] Markus Franz 2011, LZO – A Real-time Data Compression Library, Version 2.06, 12 August 2011.