# A Configurable Quantized Compressed Sensing Architecture for Low-power Tele-Monitoring

Aosen Wang
Computer Science and Engineering
SUNY at Buffalo
Email: aosenwan@buffalo.edu

Chen Song
Computer Science and Engineering
SUNY at Buffalo
Email: csong5@bufflao.edu

Wenyao Xu
Computer Science and Enign[eering]
SUNY at Buffalo
Email: wenyaoxu@buffalo.edu

*Abstract*—**Energy efficiency is one of the most concerns in tele-monitoring. As the rapid development of mobile technology, wireless communication has gradually become the biggest energy sector in most tele-monitoring applications. Recently, Compressed Sensing (CS) has attracted increasingly attention to solve this problem due to its extremely low sampling rate. In this paper, we investigate the quantization effect in the Compressed Sensing architecture. We point out that quantization configuration is a critical factor towards the energy efficiency concern of the entire CS architecture. To this end, we present a configurable Quantized Compressed Sensing (QCS) Architecture, where sampling rate and quantization are jointly explored for better energy-efficiency. Furthermore, to overcome the computational complexity of the configuration procedure, we propose a rapid configuration algorithm, called RapQCS, to promote the configuration speed. Through the experiments with public physiological data, the configurable QCS architecture can gain more than $60\%$ performance-energy trade-off than the constant QCS architecture. Furthermore, our proposed RapQCS algorithm can achieve more than $200\times$ speedup on average, while only decreasing the reconstructed signal fidelity by $1.75\%$.**

## I. INTRODUCTION

Tele-monitoring system has been widely applied in various domains, such as healthcare [1], transportation [2], environmental surveillance [3] and etc. However, energy efficiency becomes the most challenging issue in the tele-monitoring system design. Due to the scaling of data throughput with the development of hardware and computing technology, wireless transmission has contributed to the biggest energy sector in the entire tele-monitoring systems [4]. In recent years, the Compressed Sensing (CS) theory [5] provides a promising solution to tackle the energy-efficiency challenge in wireless communication. The CS theory allows an accurate recovery by sampling the signals at a lower rate which is proportional to their intrinsic salient information rather than their bandwidths. By breaking the lower bound of sampling rate defined by traditional Shannon-Nyquist theorem, Compressed Sensing shows its great potential to lower the power consumption on wireless data transmission by transmitting the data in a more compact manner.

In 2004, D. Donoho, E. Candes and T. Tao first discussed the concept of Compressed Sensing in [5], [6]. Besides energy-efficient sensing, CS theory has been successfully applied to many different fields. M. Duarte *et al.* [7] designed a single pixel camera based on CS, and J. Wright *et al.* [8] employed CS theory into face recognition to reach astonishing results. However, some fundamental technical challenges on the prac-

tice of Compressed Sensing in tele-monitoring system remain unsolved. One of the most concerns is that, in modern computing system, analog signals must be digitized before wireless transmission, and quantization effect in compressed sensing is still under study regarding energy-performance trade-off. For example, different parts in tele-monitoring system emphasize on different design criteria. Specifically, distributed sensors are usually energy-bounded, and more sensitive to energy consumption, while data aggregators are more focusing on data quality.

In this paper, we propose and investigate a configurable Quantized Compressed Sensing (QCS) architecture. Besides the sampling rate, it integrates the quantization operation in practice into the CS framework. With the extensive experiments, we have proved that quantization configuration is a sensitive factor in QCS architecture in term of its energy-efficiency. We build the model for the configurable QCS architecture to quantitatively evaluate its performance-energy trade-off. Moreover, we develop a fast configuration algorithm, called RapQCS, to further reduce the run time while almost compromising no performance deviation of the QCS architecture.

In this work, our main contributions have three-fold:

- We investigate the configurable Quantized Compressed Sensing architecture, which considers both the sampling rate and quantization configuration. We analyze the significance of the quantized bit resolution in the CS framework, and prove that the optimization of quantization configuration along sampling rate can provide a better performance-energy trade-off.

- We develop an RapQCS algorithm to fast locate the optimal configuration on sampling rate and bit resolution, with a bounded energy budget in practice, which can drastically reduce the run time while keeping the excellent efficiency capacity of QCS,

- We evaluate the QCS architecture by the experiments with challenging Electroencephalography (EEG) signals. Results indicate that our configurable Quantized Compressed Sensing architecture can improve the performance-energy trade-off by more than $60\%$. Meanwhile, our fast RapQCS algorithm can offer $200\times$ speed-ups while introducing $1.75\%$ average distortion rate.

The remainder of this paper is organized as follows:

Section II introduces prior work of Compressed Sensing and quantization. Basics of conventional CS theory are described in Section III. Our proposed efficient Quantized Compressed Sensing architecture is elaborated in Section IV, and Section V presents our fast RapQCS algorithm. Related experiments and evaluations are discussed in Section VI. And Section VII is the conclusion of the paper and our future work.

## II. RELATED WORK

Most research works on Compressed Sensing are focusing on reconstruction algorithms to pursue lower sampling rate and better signal quality. Reconstruction algorithms consist of three categories, greedy algorithm, convex optimization and iterative thresholding. Greedy algorithm aims to select the most significant components in sparsity-inducing bases, such as Orthonormal Matching Pursuit (OMP) [9] and Compressive Sampling Matching Pursuit (CoSaMP) [10]. Convex optimization method is based on optimizing $\ell_1$ norm problem and its variants. Representatives include Basic Pursuit (BP) [11], NESTA [12] and Gradient Projection [13]. And iterative thresholding can solve the $\ell_1$ penalized least square problem with fast speed, such as Iterative Splitting and Thresholding (IST) [14] and Iterated Hard Shrinkage (IHT) [15]. Also, Liu *et al.* implemented an energy-efficient reconstruction algorithm on FPGA [16]. Although these reconstruction algorithms can either reduce the measurements dimension or improve the reconstruction accuracy, they ignore that quantization is an inevitable process when applying CS into practical applications.

There are some existing works which take quantization effects into account. Dai *et al.* [17] studied the quantization effects on reconstruction error of CS. [18] and [19] mitigated the quantization effects by adapting the CS reconstruction algorithm. Also, an optimal quantizer was designed in [20] for LASSO reconstruction under high-resolution quantization assumption. Above researches concentrate on reducing the impact from quantization by modeling this process as noise introducing, or designing a better quantizer to minimize the MSE compared with no quantization case. However, the quantization effect on compressed signals is not the information of interest in practice. Fauvel *et al.* gave a comprehensive EEG tele-monitoring architecture in [21], but took the bit resolution of quantization operation as a constant value. In fact, the quantization effort on reconstruction error is indeed related to the over system performance. Moreover, they did not investigate the impact of quantization effect on energy consumption in tele-monitoring applications.

## III. BACKGROUND OF COMPRESSED SENSING

Compressed Sensing is a new emerging low-rate sampling scheme for the signals that are known to be sparse or compressible in certain basis. It has been successfully applied in image processing, pattern recognition and wireless communications.

We assume $x$ is an $N$-dimension vector space and sampled using $M$-measurement vector $y$:

$$y = \Phi x, \tag{1}$$

where $\Phi \in R^{M \times N}$ is the sensing array, which models the linear encoding, and $M$ is defined as the sampling rate in $N$-dimensional Compressed Sensing. The elements in $\Phi$ are either Gaussian random variables or Bernoulli random variables [22]. Because of $M << N$, the formulation in Eq. (1) is undetermined, and signal $x$ can not be uniquely retrieved from sensing array $\Phi$ and measurements $y$. However, under certain sparsity-inducing basis $\Psi \in R^{N \times N}$, the signal $x$ can be represented by a set of sparse coefficients $u \in R^N$:

$$x = \Psi u, \tag{2}$$

that is, the coefficient $u$, under the transformation $\Psi$, only has few non-zero elements. Therefore, based on Eq. (1) and (2), the sparse vector, $u$, can be represented as follows:

$$y = \Phi \Psi u = \Theta_{M \times N} u, \tag{3}$$

where $\Theta_{M \times N} = \Phi \Psi$ is an $M \times N$ array, called measuring matrix. Due to the prior knowledge that the unknown vector, $u$, is sparse, it is possible to estimate the value, $u$, using the $\ell_0$ minimization formulation as follows:

$$u = \min \|u\|_0, \quad s.t. \quad \|y - \Theta u\| < \epsilon, \tag{4}$$

where $\epsilon$ is the reconstruction error tolerance. The formulation in Eq. (4) is a determined system with unique solutions. However, $\ell_0$ is a NP-hard problem [23], and one of the methods to solve (4) is to approximate $\ell_0$ minimization formulation to $\ell_1$ minimization formulation:

$$u = \min \|u\|_1, \quad s.t. \quad \|y - \Theta u\| < \epsilon. \tag{5}$$

It has been proved that, under the condition of Restricted Isometry Property (RIP) [18], $\ell_1$ has been theoretically proven to be equivalent to minimizing $\ell_0$. Moreover, $\ell_1$ minimization is convex and can be solved within the polynomial time. In this work, we will use the $\ell_1$-based approach Compressed Sensing. After estimating the sparse coefficient $u$ with the formulation in Eq. (5), the original input signal $x$ can be recovered directly:

$$\dot{x} = \Psi u. \tag{6}$$

We can see that, in the traditional Compressed Sensing formulation, both reconstruction error and wireless communication energy are determined by the sampling rate $M$. People only consider to adapt $M$ with regard to the different design constraints.

## IV. CONFIGURABLE QUANTIZED COMPRESSED SENSING ARCHITECTURE

### A. Architecture Overview

The traditional CS theory did not take the quantization into the formulation (see Eq. (3)). In the practical applications, original signals are analog in nature and need to be quantized before transmitting over the wireless channel. Therefore, the compressed signal, $y$, should be processed by a quantization model formulated as follows:

$$\widehat{y} = Q_b(y), \tag{7}$$

where $Q_b(.)$ is the quantization function, and $\widehat{y}$ is the quantized representation of $y$ with $b$ bits. When considering the quantization process into the Compressed Sensing architecture, the CS formulation is reformulated:

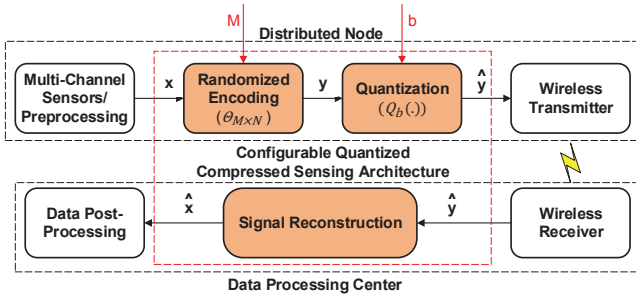$$\widehat{u} = \min \|u\|_1, \quad s.t. \quad \|\widehat{y} - \Theta u\| < \epsilon. \tag{8}$$

Fig. 1.  Configurable Quantized Compressed Sensing Architecture: it includes three modules, Randomized Encoding, Quantization and Signal Reconstruction.

By solving the formulation in Eq. (8), we can obtain the sparse representation $\widehat{u}$ from the quantized measurement $\widehat{y}$. Therefore, the reconstructed signal, $\widehat{x}$, is retrieved by:

$$\widehat{x} = \Psi\widehat{u}. \tag{9}$$

The sensing framework based on the formulation in Eq. (8) is defined as Quantized Compressed Sensing (QCS) architecture, which is illustrated as Figure 1.

We can see that the Quantized Compressed Sensing architecture consists of three parts, for example, randomized encoding, quantization and signal reconstruction module. Original analog signals, which usually denote the raw analog data, $x \in R^N$, coming from sensors, are encoded into an $M$-dimensional vector, $y \in R^M$, by linear encoding $\Theta_{M \times N}$. Through the quantization scheme $Q_b(.)$, every measurement becomes a certain $b$-bit digital representation, $\widehat{y}$. Wireless transmitter streams these digitalized measurements data to the receiver. When wireless receiver gets the data extracted from bit stream, it performs reconstruction algorithms to recover $N$-dimension original input signal $x$ from the quantized $M$-dimension compressed measurements $\widehat{y}$. The reconstructed signal $\widehat{x}$ is sent to the data post-processing model for specific applications, such as classification [24], signal demodulation [25], signal separation [26]. In this architecture, linear random encoding and quantization modules form a distributed node for sensor's flexibility. Reconstruction and post-processing functions are conducted in data processing center. Note that as shown in Figure 1, in this QCS architecture, both sampling rate, $M$, in the random encoding module and bit resolution, $b$, in the quantization module are configurable. These parameters can be setup according to different applications.

### B. Models of Energy and Performance in the QCS Architecture

In this part, we will discuss the models of energy and performance in the QCS architecture. In the above Quantized Compressed Sensing architecture, distributed node is energy-bounded with limited energy budget, and the data center is performance-driven, focusing on the signal quality. In distributed nodes, the power consumption is dominated by wireless communication, and communication energy is proportional to the volume of data stream. Therefore, its energy model can be formulated as follows:

$$E = C \times M \times b, \tag{10}$$

where $M$ is the sampling rate, $b$ is bit resolution in quantization in the QCS architecture, and $C$ is the energy per bit, i.e., the

energy consumption of transmitting 1 bit data[1]. We can see that we can change the power consumption of distributed nodes through the $M$ and $b$ setup. The larger $M$ and $b$, the more energy consumption in the distributed nodes.

The aim of the data center is to reconstruct the original signal from the streaming compressed data for the post-processing. For the simplicity of presentation, we use the reconstruction error as the performance metric in the data center. Therefore, the performance model in the data center can be defined as follows:

$$P(M,b) = \frac{\|x - \widehat{x}\|_2}{\|x\|_2} \times 100\%, \tag{11}$$

where $P(M,b)$ denotes the performance metric, i.e., the reconstruction error under the configuration of sampling rate, $M$, and resolution bit, $b$. $\widehat{x}$ denotes the recovered signal and $x$ is the original input signal. Because $\widehat{x}$ is directly derived from the measure $\widehat{y}$, the performance is affected by the sampling rate, $M$, and the bit resolution, $b$. Specifically, $M$ has an impact on the performance of reconstruction algorithms, and $b$ determines the quantization noise in the measures. From the models in Eq. (10) and (11), we can see the QCS architecture can reach different energy-performance trade-off through the configuration of $M$ and $b$.

### C. System Design Formulation

In this part, we will present the design formulation towards CS-based distributed systems. The distributed sensor nodes are with a capacity-limited battery, and the battery lifetime is usually the design constraint.

Given an energy bound, $E_0$, for the distributed node, the design criterion in the QCS architecture is to find the optimal configuration of $M$ and $b$ such that the energy $E$ is less than $E_0$, and the reconstruction error is as small as possible. This design formulation can be formulated as follows:

$$(M,b)_{opt} = arg \min_{M,b}(P(M,b)), \quad s.t. \quad C \times M \times b \leq E_0, \tag{12}$$

where $(M,b)_{opt}$ denotes the optimal energy configuration under energy bound $E_0$. We can see that the total design space in the formulation in Eq. (12) is $b \times M$, where $b$ is the bit resolution in the quantization, and $M$ is the sampling rate. Note that the objective function is not in an analytic form, and the constraint function is non-convex. Therefore, it is challenging to efficiently obtain $(M,b)_{opt}$.

### D. Brute Force Algorithm

The most straightforward method to find $(M,b)_{opt}$ is the brute force method. It is also called exhaustive searching, a general technique enumerating all potential solutions to check if it satisfies the problem's statement. In our challenge, when energy bound $E_0$ changes from 0 to $b \times M$, we can get a P-E trade-off point $(M,b)_{opt}$ on every energy bound. If we connect all these P-E trade-off points, it will form an optimal P-E trade-off curve, called the Pareto's curve. Once the Pareto's curve is depicted in the P-E space, we can easily identify the optimal energy configuration $(M,b)_{opt}$ under any given bound

---

[1]The energy per bit is determined by the wireless communication protocol and usually a constant.

$E_0$. We employ the brute force method to search the entire Performance-Energy space for the Pareto's curve and then set the optimal energy configuration under $E_0$ by looking up the Pareto's curve. Actually, after we give the definitions of energy and performance, this problem can be solved by searching for a series of P-E trade-off points, who holds the best trade-off in lower energy interval. It can be formulated as follows:

$$(M,b)_{opt}^{i+1} = arg \min_{M,b} \|E_{opt}^{i+1} - E_{opt}^i\|,$$
$$s.t. \quad \begin{cases} E_{opt}^{i+1} > E_{opt}^i, \\ P(E_{opt}^{i+1}) < P(E_{opt}^i), \end{cases} \tag{13}$$

where $n$ is the $i$-th optimal energy configuration. We execute our searching by the ascending order of energy value. We start from the minimal energy level, and then energy level increases gradually. It is a common case that there could be more than one $(M,b)$ configuration on the same energy level, which results from our energy definition. Under this situation, we exhaustively check every possible $(M,b)$ configuration for the optimal trade-off. That is, we take every energy configuration $(M,b)$ as our candidates, but our searching order is by ascending order of energy values. The entire algorithm is as ALGORITHM 1. Also, an example of the Brute Force algorithm is shown as Figure 2.

---

**Algorithm 1** Brute Force Algorithm

**Input:** $E_0$: given energy bound
   $D_E$:the set of all the energy configuration
   $(M,b)_{min}$:configuration corresponding to the least energy level
   $check13(.)$:function to check if configuration satisfies Eq. (13)
**Output:** $G_E$:set of configurations of P-E trade-off points.
   $(M,b)_{E_0}$:the optimal configuration with energy bound $E_0$
1: $G_E = G_E \bigcup (M,b)_{min}$
2: $(M,b)_{opt} = (M,b)_{min}$
3: **for** each $(M,b)$ in $D_E$ **do**:
4:   **if** $check13((M,b),(M,b)_{opt})$ **then**
5:     $G_E = G_E \bigcup (M,b)$
6:     $(M,b)_{opt} = (M,b)$
7:   **end if**
8: **end for**
9:
10: **for** $(M,b) \in G_E$ **do**
11:   **if** $M \times b \leq E_0$ **then**
12:     $(M,b)_{E_0} = (M,b)$
13:   **else**
14:     $break$
15:   **end if**
16: **end for**

---

From Figure 2, every red point represents a P-E point, located by its energy configuration $(M,b)$ and corresponding performance $P(M,b)$. The green line is the Pareto's P-E curve, connecting all the P-E trade-off points, which are centered in the blue triangular markers. However, run time of Brute Force algorithm increases linearly as the candidate number grows, which means larger searching space in our energy-efficiency challenge.
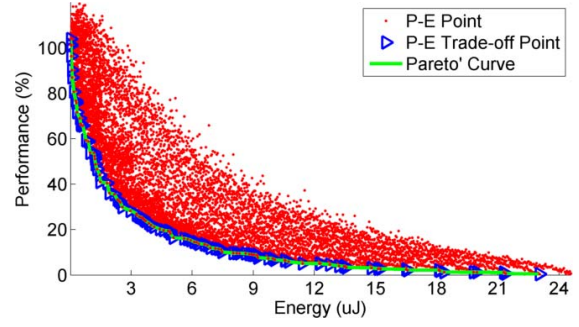


Fig. 2. Brute Force Algorithm: red point indicates a P-E point, blue triangular marks a P-E trade-off point and the green line is the Pareto's curve.

## V. RAPID QCS CONFIGURATION WITH A BOUNDED ENERGY BUDGET

In practical applications, it is normal that an upper-energy bound is set to constraint system's power consumption. For our Quantized CS Architecture, energy bound is usually set on distributed node for its sensitivity to power consumption. It is a big challenge to fast locate the optimal energy configuration under a given energy bound. In the following part, we will tackle this challenge.

### A. Energy-efficiency Control Formulation

When there is an energy budget $E_0$ for the distributed node, it is critical to find an optimal energy configuration corresponding to the best performance under given constraint. This problem has been formulated in Eq. (12), which is an NP-hard problem without any polynomial optimal solutions. An intuitive solution is applying Brute Force algorithm to search the sub-space entirely whose energy is limited in the bound $E_0$. But this method is time-consuming and impractical to large-scale problems. Based on our empirical estimation, the energy level where optimal configuration locates should be just near the energy bound. Thus, it seems not a wise way to search the whole bounded energy space. In next subsection, we will introduce a better heuristic local search algorithm based on the sensitivity analysis to fast locate the optimal configuration.

### B. RapQCS Algorithm

We first define a parameter space $A$ in our Quantized CS architecture:

$$A = \{p_1, ..., p_i, ..., p_n\}, \tag{14}$$

where $p_i$ is the parameter affecting energy or performance in the architecture. For example, due to our energy's definition, $M$ and $b$ are two parameter members of this model. Since parameters are all discrete values, we define a new neighborhood operator " $\rightarrow$ ":

$$s_k \rightarrow l = \begin{cases} s_{k+l} & 1 \leq k+l \leq L, \\ \perp & otherwise, \end{cases} \tag{15}$$

where $s_k$ is the $k$-th configuration of parameter $p_i$. $L$ is the dimension of $p_i$. In our RapQCS, cost function is defined as performance:

$$f_c = P = \frac{\|x - \widehat{x}\|_2}{\|x\|_2} \times 100\%. \tag{16}$$

Assuming $a = (M, b)$ is a configuration vector from parameter set $A$, we can define sensitivity $\sigma$ at configuration $a$:

$$\sigma(a) = f_c(\delta(a)) - f_c(a), \tag{17}$$

where $\delta(a) \in N_d(a)$ is the $d$-neighborhood of configuration $a$. For our QCS architecture, $\delta(a)$ indicates 8-neighborhood, $\delta(a) \in N_8(a)$, with two parameters in $A$. So if the local search chooses the most sensitive direction to proceed, which is with the fastest decreasing of the cost function, optimal (or sub-optimal) configuration can be achieved eventually. For a better convergence speed and a more stable solution, we would like to add a tuning phase to reasonably estimate an initial configuration $a_0$ before the local search based on sensitivity.

$$a_0(EB) = E_{h \in H}(a_{opt}(h, EB)), \tag{18}$$

where $H$ is the application set. $a_{opt}(h, EB)$ is the optimal configuration of application $h$ under energy bound $EB$. In the tuning phase, the Brute Force algorithm will be employed to search the P-E space for optimal configuration $a_{opt}(h, EB)$. So the whole algorithm of RapQCS algorithm is as ALGORITHM 2.

---

**Algorithm 2** RapQCS Algorithm

**Input:** EB: given energy budget
    $a_0(EB)$: starting configuration
    $f_c$: cost function
    $\sigma(a)$: sensitivity at configuration a
**Output:** $a_{opt}(EB)$: optimal configuration with $EB$
 1: $a_{opt}(EB) = a_0(EB)$
 2: Search for $(M, b)$:

$$(M, b) = arg\min_{M,b}(\sigma(a_{opt})), \quad s.t. \quad \begin{cases} \sigma(a_{opt}) < 0 \\ M \times b < EB \end{cases}$$

 3: **if** find such $(M, b)$ **then**
 4:    $a_{opt}(EB) = (M, b)$
 5:    $Go\ to\ Step\ (2)$
 6: **else**
 7:    $EXIT$
 8: **end if**

---

## VI. EXPERIMENTS

### A. Experimental Setups and Datasets

In this section, we describe the performance evaluation on our work from two aspects. First, we evaluate the advantage of the configurable QCS architecture, and investigate the performance-energy trade-off gain compared with the traditional CS architecture. Second, we test the effectiveness and efficiency of the RapQCS algorithm, i.e., the reconstruction error, energy-bound accuracy and runtime speedup compared with the brute force method.

In our experiment, we select body sensor networks (B-SN) as the evaluation scenario. BSN is an emerging application of tele-monitoring, which builds the wireless sensor network around human body. Physiological signals, such as Electrocardiography (ECG), Electromyography (EMG), and Electroencephalography (EEG), can be gathered to monitor human health status. Without the loss of generality, we take four typical EEG signal samples from Physionet [27] as our
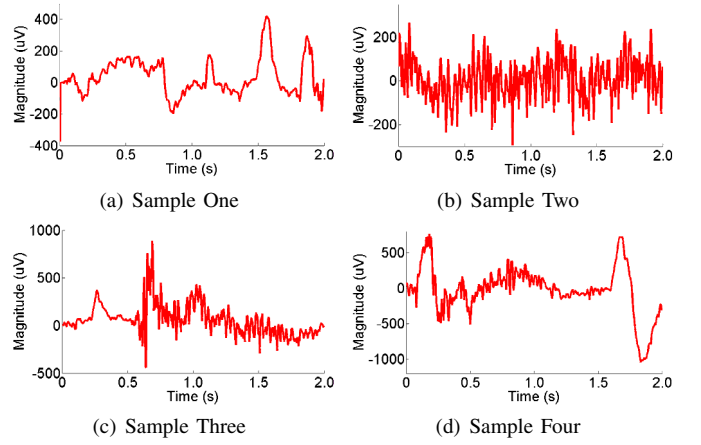


Fig. 3. The waveforms of the four EEG samples from Physionet, and these four patterns are significantly different.

test bench. Each EEG sample consists of 512 points, and their waveforms are shown in Figure 3. We can see that these four EEG samples are with significantly different patterns.

Considering that EEG signals are usually sparse under the Discrete Wavelet Transform (DWT) basis, we use IDWT as the sparsity-inducing transformation basis, $\Psi$, [28]. All our experiments use Gaussian random variables as sensing array and take uniform quantization strategy. For an efficient 350 $\mu$W MSK/FSK transmitter, its TX energy per bit is 3 nJ/bit under MSK modulation and 120 Kbps of bit rate [29]. Under this condition, we setup that the average energy consumption of wireless communication is $C$ = 3 nJ/bit.

### B. Configurable QCS v.s. Traditional CS

*1) Energy-efficiency Comparison:* The significance of the configurable QCS architecture is that it takes into account the flexibility of resolution bit, $b$, in the quantization module. The experiment in this part is designed to compare how much the performance-energy trade-off is gained compared to the traditional CS architecture. In our experiment, we take four EEG samples for the benchmark evaluation. As shown in Figure 3, these four samples are different and have no periodicity. Also, there are no coherence features between any two samples in the time domain. In the traditional CS architecture, bit resolution of quantization is set as a fixed value, such as 8, 12 or 16. Considering that the bit resolution requirement is usually high in the EEG application, we can set $b = 16$ for the traditional CS architecture during the experiments. For the QCS architecture, bit resolution is ranging from 1 to 16. In the experiment, we use the brute force method to find the Pareto's curve of the performance-energy (P-E) space, and the results of four EEG samples are shown in Figure 4 (a)-(d), respectively.

In Figure 4 (a)-(d), the red line is the Pareto's curve of the configurable QCS architecture, and the blue line is the Pareto's curve of the traditional CS architecture. Every marker is an optimal P-E point. Here we can see that the P-E trade-off in the configurable QCS architecture is significantly better than that of the traditional CS architecture. To quantitatively analyze the P-E gain, we define a metric, $A_{PE}$, which is the area between the Pareto's curve and the energy axis. Specifically,
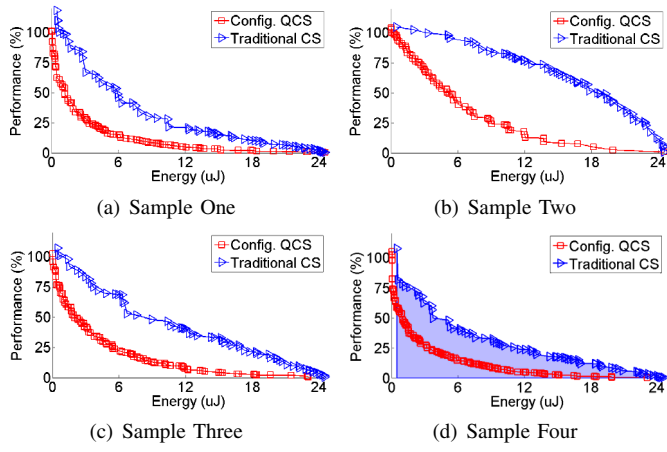
Fig. 4. (a)-(d) are the Pareto's curves of every sample; the blue area in (d) is an example of $A_{PE}$.

TABLE I. TRADE-OFF ENHANCEMENT TABLE

| Sample NO. | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $A_{PE}(CB)$ | 2421.5 | 5591.5 | 3361.2 | 2236.0 |
| $A_{PE}(OB)$ | 824.7 | 2016.9 | 1182.8 | 802.3 |
| $TO_{enhance}$ (%) | **65.94** | **63.93** | **64.81** | **64.12** |

the Pareto's curve consists of all optimal P-E trade-off points, and these points form a set of trapezoidal areas. Therefore, we can add up all these areas and obtain $A_{PE}$ as follows:

$$A_{PE} = \sum_i S_i, \qquad (19)$$

where $i$ is the P-E point's number, $S_i$ is the trapezoidal area between two adjacent P-E points of $i$ and $i-1$, and the energy axis. For example, the $A_{PE}$ of the traditional CS architecture is indicated as the blue area in Figure 4(d). We can further define the trade-off enhancement metric as follows:

$$TO_{enhance} = \frac{A_{PE}(CB) - A_{PE}(OB)}{A_{PE}(CB)}, \qquad (20)$$

where $A_{PE}(CB)$ is the area of Pareto's curve of the traditional CS architecture, and $A_{PE}(OB)$ is the area of Pareto's curve of the configurable QCS architecture. According to Formula (20), we can calculate the enhancement as TABLE I.

From TABLE I, the improvement of P-E trade-off is all more than 63% by considering bit resolution. It is indicated that our configurable QCS architecture can greatly enhance the energy efficiency than the traditional CS architecture. According to our enhancement definition, it is obvious that Sample One gains the most efficiency and Sample Two gets the least improvement, which is contradicted with our observation. This is because our enhancement definition is an $A_{PE}$ ratio corresponding to the Pareto's curve of the traditional CS. The $A_{PE}$ of Sample Two under the traditional CS is much more than that of other samples. Thus, its enhancement is down to the smallest one.

*2) Reconstructed Signal Comparison:* In practical applications, precise reconstruction signals usually need post-processing. Thus, we would like to further check the signal quality of traditional CS and configurable QCS in this experiment. We can directly have a look at the energy configuration and performance when the energy bound is $E_0 = 9\mu J$. Related
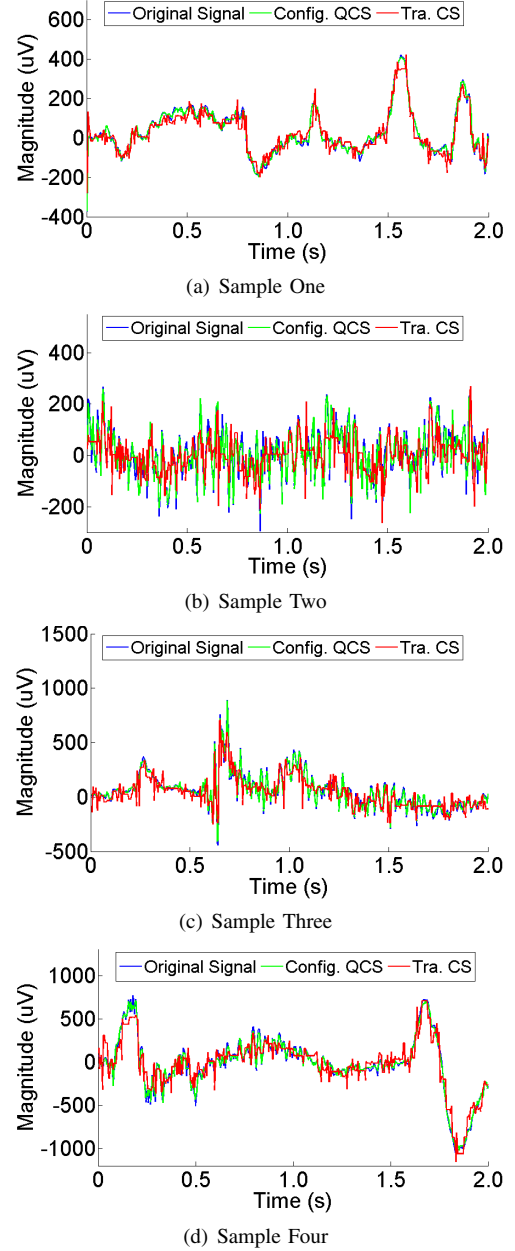


Fig. 5. Reconstructed signals of four samples when the energy bound is set as $9\mu J$.

waveforms are depicted in Figure 5, and detailed energy configurations with corresponding performance information are illustrated in TABLE II.

From the contrasting in Figure 5, we can see that original signal (blue line) is almost covered by our configurable QCS (green line). Reconstruction of the traditional CS (red line) is with larger distortion than the configurable one, especially for Sample Two and Three. The performance of the traditional CS is all larger than the configurable QCS from TABLE II. Specifically, the performance difference of Sample Two and Three between these two cases are both more than 30%, resulting in an easy identification of their waveforms' differences. Then for Sample One and Four, their difference of reconstruction error rate is about 20%, and their distortion is not so significant.

TABLE II.     ENERGY CONFIGURATION AND PERFORMANCE ON
$E_0 = 9\mu$J.

| Sample NO. | | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| Configurable QCS | $M$ | 413 | 498 | 497 | 493 |
| | $b$ | 7 | 6 | 6 | 6 |
| | $E$ ($\mu$J) | 8.67 | 8.96 | 8.95 | 8.87 |
| | $P$ (%) | **8.7** | **26.0** | **13.3** | **9.5** |
| Traditional CS | $M$ | 181 | 184 | 187 | 174 |
| | $b$ | 16 | 16 | 16 | 16 |
| | $E$ ($\mu$J) | 8.69 | 8.83 | 8.98 | 8.35 |
| | $P$ (%) | **30.6** | **84.9** | **48.1** | **33.2** |



(a) Sample One

(b) Sample Two

(c) Sample Three

(d) Sample Four
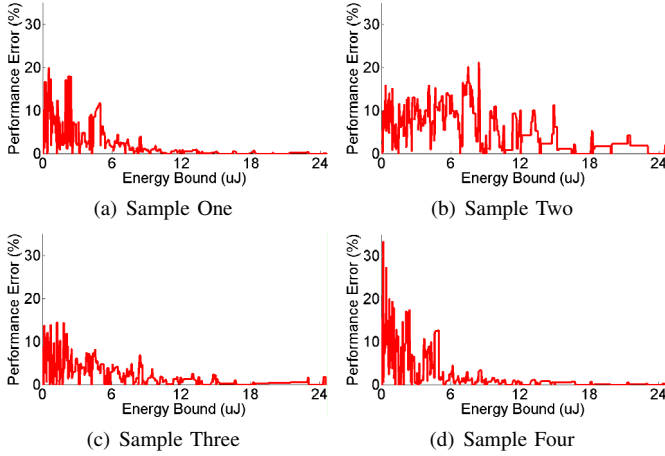
Fig. 6.    Absolute error of performance between RapQCS curve and Brute-force's curve.

Also, Sample One holds the least reconstruction error rate, even less than $10\%$ under the configurable QCS. We can not find any big distortion when performance is below $10\%$. However, for Sample Two, its reconstruction error rate under traditional CS is up to $84.9\%$. Reconstructed signal is far away from the original. Its error rate in the configurable QCS is $26.9\%$, a very high error rate in all the QCS's reconstructions. The large error rate results from the weak sparsity [30] of Sample Two, seeming like a random sequence, which DWT can not reconstruct with high quality.

### C. Rapid Optimal Configuration with Energy Bounded Budget in QCS

*1) Performance Accuracy Analysis:*  In this experiment, we test the performance of our RapQCS algorithm compared to the brute-force method. We use four samples of EEG signals. Their patterns are shown in Figure 3. As our RapQCS algorithm's requirement in tuning phase, we consider all cases of three EEG samples' combination as tuning applications, and the remaining one under corresponding case is the test waveform. For a comprehensive viewing, we set energy bound on every discrete energy level. After the searching of RapQCS algorithm, we can obtain an optimal RapQCS performance curve (we just call it RapQCS curve). The absolute difference of performance between RapQCS curve and brute-force curve is illustrated in Figure 6.

In Figure 6, there are $4$ sub-graphs indicating the absolute error between RapQCS curve and Brute-force curve. The common trend of them is that the absolute error between these two types of curves decreases rapidly as energy bound increases. For all samples except No. 2, the relatively larger

errors always occur on the interval whose energy bound is less than 3 $\mu$J. When energy bound increases more than 6 $\mu$J, their absolute error can fluctuate in a relatively small range, less than $5\%$. On the low energy bound, measurements number $M$ and bit resolution $b$ are both small. For $\ell_1$ convex optimization problem, smaller $M$ will result in large reconstruction error rate. It's reported that $M$ should meet the following condition for a successful reconstruction:

$$M \geq Klog(\frac{N}{K}), \tag{21}$$

where $K$ is the sparsity of input signal. Thus small $M$ even may cause reconstruction failure. Smaller $b$ introduces more quantization error into the reconstruction, which is always modeled by noise. Therefore, performance in small energy interval takes on strong randomness, making our RapQCS method trapped in local minima easily. As energy increases, $M$ and $b$ both grow gradually. Less quantization error and more accurate convex optimization make reconstruction error rate decreasing rapidly and more stable. Less local minima provides a larger chance for our algorithm to approximate to the optimal solution. For Sample Two, its absolute error is relatively larger than other cases. In Figure 3, the waveform of Sample Two is more like a random sequence. And DWT is not good at dealing with such signals. Thus, the performance may engender more local minima to confuse our RapQCS method to reach its right destination.

Specifically, we use average error rate $AvgError$ and standard deviation $StdError$ to quantitatively evaluate the performance accuracy of our RapQCS algorithm. Because our performance has already been defined as reconstruction error rate, we can calculate the average absolute error rate and standard deviation on all the energy bounds to indicate the overall error level:

$$AvgError = \frac{1}{|EB|} \sum_{eb \in EB} abs(P_{Rap}(eb) - P_{opt}(eb)), \tag{22}$$

$$StdError = (\frac{1}{|EB|} \sum_{eb \in EB} (abs(P_{Rap}(eb) - P_{opt}(eb)) \\ - AvgError)^2)^{\frac{1}{2}}, \tag{23}$$

where $EB$ is the set of all energy bounds. $P_{Rap}(eb)$ indicates the performance of RapQCS method under energy bound $eb$. And $P_{opt}(eb)$ is the optimal performance querying from the brute-force curve. The $AvgError$ and $StdError$ of all $4$ samples are shown in Figure 7.

From Figure 7, our RapQCS method holds about $1.75\%$ average error rate in general case except Sample Two. Their standard deviations are about $3\%$, which is a trivial fluctuation. As our observation in Figure 6, these three samples only suffer from larger error rate in a small low energy bound interval, while they hold smaller error rate on high energy bound. For Sample Two, it has the largest average error rate and standard deviation, for its not good reconstruction. Even under this situation, the level of average performance error rate of all the samples is still around $4\%$, with about $3.5\%$ standard deviation. We can see that the impact of reconstruction distortion affected by such subtle error rate can be nearly neglected. Therefore, when given a bounded energy budget, our RapQCS algorithm can reach an accurate performance-energy trade-off.
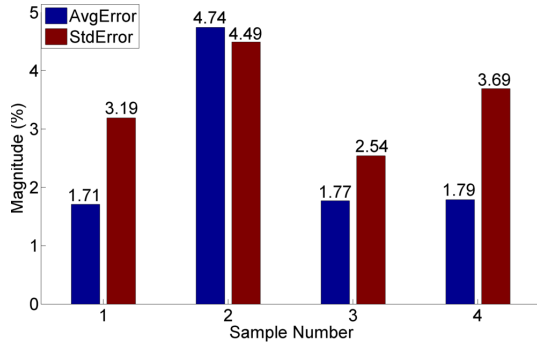
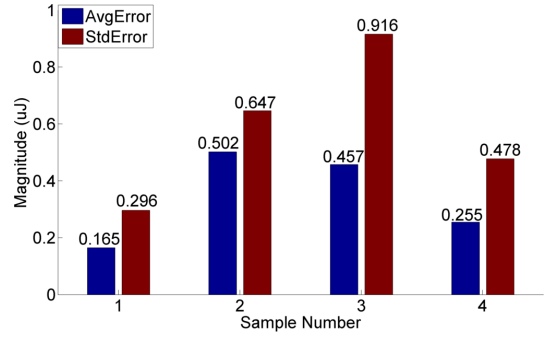Fig. 7. $AvgError$ and $StdError$ of absolute performance error of all EEG samples.



(a) Sample One

(b) Sample Two



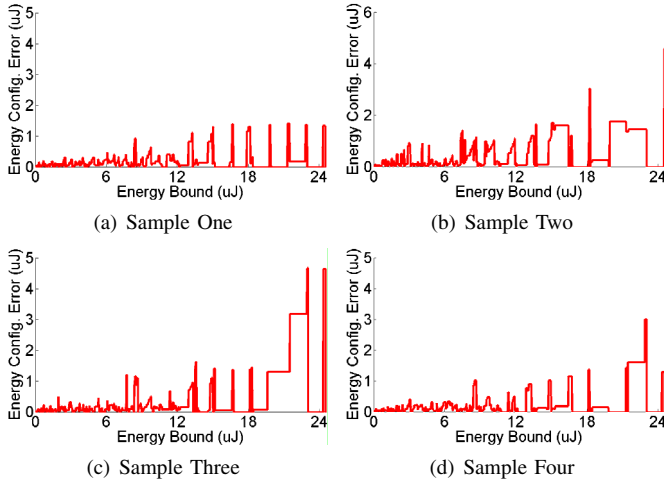(c) Sample Three

(d) Sample Four

Fig. 8. Absolute energy configuration error between RapQCS algorithm and Brute Force algorithm.

*2) Energy Accuracy Analysis:* In this part, we will check the energy accuracy between our RapQCS algorithm and the Brute Force algorithm. The absolute energy configuration error of these two algorithms is illustrated in Figure 8.

In Figure 8, combining the trends of 4 samples, the absolute error is very small on low energy bound. As energy bound increases, this error shows a growing trend. We can see that on high energy bound, the absolute error is either very small or relatively big. The reason is that our P-E space consists of many discrete energy levels. As the energy definition in Eq. (10), the step length between two adjacent energy levels will increase as energy is bigger. Thus, any inaccurate searching will result in a big absolute energy error. In all these 4 samples, Sample One and Four can keep accurate configurations on high energy bound, but other two samples suffer from different degrees of deviation. On all energy bounds, Sample Two shows relatively large fluctuation.

Because our reconstruction algorithm cannot recover Sample Two well, its performance may take on strong randomness, which will result in more confusing local minima. Thus the energy configuration of our RapQCS under Sample Two has larger fluctuations than other samples. Also, when we take a closer look at the performance on high energy bound in Figure 6, their performances are all in a small range. This persuasively demonstrates the superiority of our configurable



Fig. 9. $AvgError$ and $StdError$ of absolute energy configuration error of all EEG samples.

QCS architecture to exploit the energy efficiency. Specifically, we will still use the $AvgError$ and $StdError$ to evaluate the energy configuration accuracy of our RapQCS method. Since we already get the absolute energy error on every energy bound, these two indicators of all 4 samples can directly calculated, as shown in Figure 9.

Figure 9 confirms our previous observation. The average absolute errors of Sample Two and Three are relatively larger than other samples, about $0.2$ $\mu$J more. Their standard deviations are also much more than others. Sample Two is mainly affected by the reconstruction algorithm while Sample Three is caused by the initial energy configuration setup. Sample Three is with the biggest standard deviation, because the large gap of absolute error between its lower bound and higher bound. Although there are some impacts caused by these factors, the average error rate of our results is only about $0.4$ $\mu$J, which is a strong support for the excellent energy accuracy of our RapQCS algorithm.

*3) Run Time Analysis:* Although Brute Force method can find the optimal solution in Performance-Energy space, its huge computational burden is a main obstacle for the practical application. In this experiment, we will continue to explore the run time between RapQCS method and Brute Force method. Also, we set energy bound on every energy level to do a comprehensive comparison. Related run time graphs are illustrated in Figure 10. Because the run time difference between these two methods is too large, we employ log axis of run time to give a more clear comparison.

From Figure 10, the run time of Brute Force algorithm is rising all the time. Its searching space first extends with a dramatically fast speed when energy bound is less than 3 $\mu$J. Then run time increases by a small speed after energy bound 3 $\mu$J. Because we have no information on the energy bound value ahead of time, when we get the energy bound, the Brute Force method must search all the sub-space whose energy is less than the given energy bound. However, our RapQCS method starts from its optimal initial configuration. So its solution can usually be found in several searching steps. This is a critical factor for time-reducing. Like $A_{PE}$ and $TO_{enhance}$ in Experiment *B-1)*, we will introduce $A_{run}$ and $Speedup_{run}$ to calculate the speedup quantitatively. $A_{run}$ refers to the area between run time curve and energy bound axis:

$$A_{run} = \sum_i S_{run}(i), \qquad (24)$$

(a) Sample One

(b) Sample Two

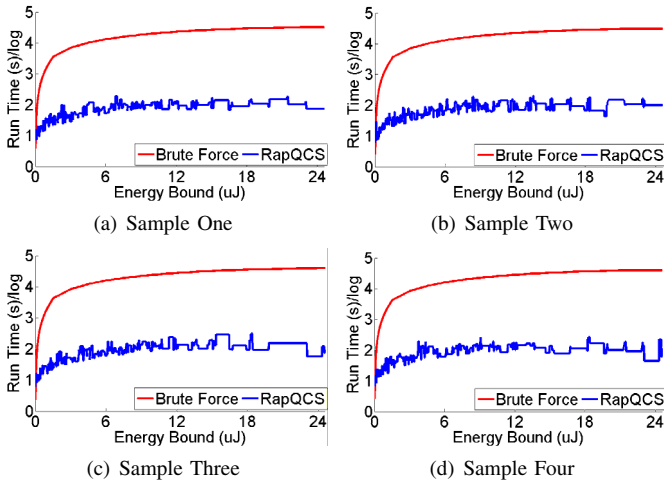(c) Sample Three

(d) Sample Four

Fig. 10. Run-time comparison between the RapQCS algorithm and the Brute Force algorithm.

TABLE III. RUN TIME SPEEDUP BETWEEN BRUTE FORCE AND RAPQCS ALGORITHM.

| Sample NO. | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $A_{run}(BruteForce)$ ($10^8$) | 1.74 | 1.68 | 2.08 | 2.11 |
| $A_{run}(RapQCS)$ ($10^5$) | 7.53 | 7.33 | 9.50 | 8.47 |
| *Speedup* | **230.3×** | **229.0×** | **218.9×** | **249.6×** |

where $S_{run}(i)$ is the trapezoidal area between two adjacent points in run time curve and energy bound axis in run-time-energy-bound axis, not the log run time axis as Figure 10. Therefore, we can define the time speedup as follows:

$$Speedup_{run} = \frac{A_{run}(BruteForce)}{A_{run}(RapQCS)}. \qquad (25)$$

TABLE III shows the speedup of 4 samples. The speedup in all cases are more than 200 times without significant difference. This is because the run time of 4 samples under Brute Force algorithm is similar. Our RapQCS method's run time fluctuates with a rising trend due to the initial energy configuration. Thus the speedups have small waves. It is firmly demonstrated that our RapQCS algorithm gives a very significant improvement of run time.

*4) Case Study:* In experiment C-1), we see the relationship between performance of RapQCS's curve and Pareto's curve. In this experiment, we will take a closer look at the reconstruction results of our RapQCS algorithm, comparing them with optimal solution and ground truth. Optimal solution indicates the optimal result from brute-forcing the performance-energy space, while ground truth is the original input signal. For a better comparison, we set an energy bound, $E_0 = 6\mu J$. Their reconstruction results are as Figure 11.

In Figure 11, the blue line is the original input EEG signal, the red signal is our RapQCS's result and the green signal is the reconstruction from brute-force algorithm. For all the sample cases except Sample Two, the patterns from RapQCS algorithm and Brute-force curve are very similar to each other. We cannot directly identify which is with a better performance between the two methods. They all approximate to the ground truth closely. Therefore, we investigate their
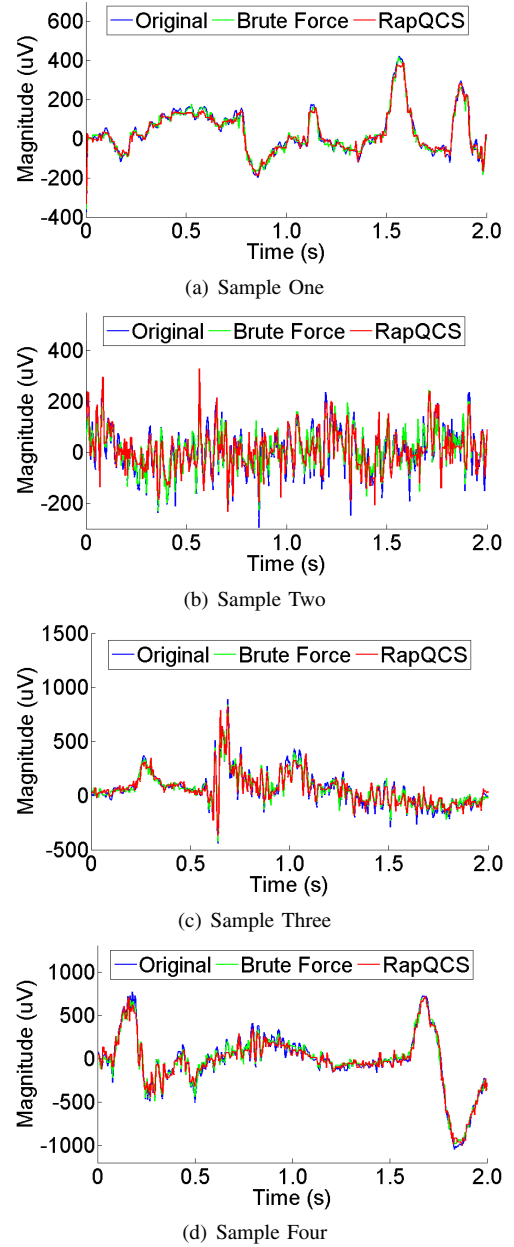


(a) Sample One



(b) Sample Two



(c) Sample Three



(d) Sample Four

Fig. 11. Reconstruction results of all four EEG samples on $E_0 = 6\mu J$.

detailed information, energy configuration and performance, listed as TABLE IV.

We can observe that except Sample Two, other absolute error rates of performance are all around 2%. This demonstrates the similar reconstruction signals between our RapQCS case and optimal case. For Sample Two, its optimal solution is closer to the ground truth than RapQCS. However, the two reconstructions of Sample Two are both not similar to the ground truth. Their error rate differences are both more than 40% according to the TABLE IV.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a configurable Quantized Compressed Sensing architecture. It jointly considers the configuration of sampling rate and quantization in its framework to make

the Compressed Sensing theory more practical. Through a case study on the application of body sensor networks, our experimental results indicated that the configurable QCS architecture can provide more than $60\%$ energy-performance trade-off gain than the traditional Compressed Sensing architecture. Also, we proposed a fast configuration algorithm, RapQCS, for the configurable QCS architecture. In experiments, the proposed RapQCS algorithm can reach more than $200\times$ speedup, while only decreasing the signal fidelity by $1.75\%$.

In our future, we will explore larger design space under the configurable QCS architecture. For example, besides the bit resolution, we will also investigate the non-uniform quantization strategies. Also, we will investigate better construction algorithms and transformation basis for complex signal, such as Sample Two in the experiment.

## ACKNOWLEDGMENT

## REFERENCES

[1] R. Paradiso, "Wearable health care system for vital signs monitoring," in *Information Technology Applications in Biomedicine, 2003. 4th International IEEE EMBS Special Topic Conference on*. IEEE, 2003, pp. 283–286.

[2] M. Al-Rousan, A. Al-Ali, and K. Darwish, "Gsm-based mobile telemonitoring and management system for inter-cities public transportations," in *Industrial Technology, 2004. IEEE ICIT'04. 2004 IEEE International Conference on*, vol. 2. IEEE, 2004, pp. 859–862.

[3] A. M. Tabar, A. Keshavarz, and H. Aghajan, "Smart home care network using sensor fusion and distributed vision-based reasoning," in *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*. ACM, 2006, pp. 145–154.

[4] S. Meystre, "The current state of telemonitoring: a comment on the literature," *Telemedicine Journal & e-Health*, vol. 11, no. 1, pp. 63–69, 2005.

[5] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[6] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *Information Theory, IEEE Transactions on*, vol. 52, no. 12, pp. 5406–5425, 2006.

[7] R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, 2008.

[8] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[9] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, 2007.

[10] D. Needell and J. A. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples," *Communications of the ACM*, vol. 53, no. 12, pp. 93–100, 2010.

[11] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM journal on scientific computing*, vol. 20, no. 1, pp. 33–61, 1998.

[12] S. Becker, J. Bobin, and E. J. Candès, "Nesta: a fast and accurate first-order method for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.

[13] M. A. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, no. 4, pp. 586–597, 2007.

[14] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on pure and applied mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[15] K. Bredies and D. A. Lorenz, "Iterated hard shrinkage for minimization problems with sparsity constraints," *SIAM Journal on Scientific Computing*, vol. 30, no. 2, pp. 657–683, 2008.

[16] B. Liu, Z. Zhang, G. Xu, H. Fan, and Q. Fu, "Energy efficient telemonitoring of physiological signals via compressed sensing: A fast algorithm and power consumption evaluation," *Biomedical Signal Processing and Control*, vol. 11, pp. 80–88, 2014.

[17] W. Dai, H. V. Pham, and O. Milenkovic, "A comparative study of quantized compressive sensing schemes," in *Proceedings of the 2009 IEEE international conference on Symposium on Information Theory-Volume 1*. IEEE Press, 2009, pp. 11–15.

[18] A. Zymnis, S. Boyd, and E. Candes, "Compressed sensing with quantized measurements," *Signal Processing Letters, IEEE*, vol. 17, no. 2, pp. 149–152, 2010.

[19] J. N. Laska, P. T. Boufounos, M. A. Davenport, and R. G. Baraniuk, "Democracy in action: Quantization, saturation, and compressive sensing," *Applied and Computational Harmonic Analysis*, vol. 31, no. 3, pp. 429–443, 2011.

[20] J. Z. Sun and V. K. Goyal, "Optimal quantization of random measurements in compressed sensing," in *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*. IEEE, 2009, pp. 6–10.

[21] S. Fauvel and R. K. Ward, "An energy efficient compressed sensing framework for the compression of electroencephalogram signals," *Sensors*, vol. 14, no. 1, pp. 1474–1496, 2014.

[22] N. R. Chaganty and H. Joe, "Range of correlation matrices for dependent bernoulli random variables," *Biometrika*, vol. 93, no. 1, pp. 197–206, 2006.

[23] M. Hyder and K. Mahata, "An approximate l0 norm minimization algorithm for compressed sensing," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 3365–3368.

[24] E. Haselsteiner and G. Pfurtscheller, "Using time-dependent neural networks for eeg classification," *Rehabilitation Engineering, IEEE Transactions on*, vol. 8, no. 4, pp. 457–463, 2000.

[25] D. Zibar, R. Sambaraju, A. Caballero, J. Herrera, U. Westergren, A. Walber, J. B. Jensen, J. Marti, and I. T. Monroy, "High-capacity wireless signal generation and demodulation in 75-to 110-ghz band employing all-optical ofdm," *Photonics Technology Letters, IEEE*, vol. 23, no. 12, pp. 810–812, 2011.

[26] E. Weinstein, M. Feder, and A. V. Oppenheim, "Multi-channel signal separation by decorrelation," *Speech and Audio Processing, IEEE Transactions on*, vol. 1, no. 4, pp. 405–413, 1993.

[27] G. B. Moody, R. G. Mark, and A. L. Goldberger, "Physionet: a web-based resource for the study of physiologic signals," *IEEE Eng Med Biol Mag*, vol. 20, no. 3, pp. 70–75, 2001.

[28] H. Ocak, "Automatic detection of epileptic seizures in eeg using discrete wavelet transform and approximate entropy," *Expert Systems with Applications, ELSEVIER*, vol. 36, no. 2, pp. 2027–2036, 2009.

[29] J. L. Bohorquez, A. P. Chandrakasan, and J. L. Dawson, "A 350 w cmos msk transmitter and 400 w ook super-regenerative receiver for medical implant communications," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 4, pp. 1248–1259, 2009.

[30] Z. Zhang, T.-P. Jung, S. Makeig, and B. D. Rao, "Compressed sensing of eeg for wireless telemonitoring with low energy consumption and inexpensive hardware," *Biomedical Engineering, IEEE Transactions on*, vol. 60, no. 1, pp. 221–224, 2013.