Journal of Science & Technology 123 (2017) 036-042

Wireless LAN Based Experiment and Evaluation of Effect on the Two-State ECG Compression Algorithm

Duong Trong Luong**, Nguyen Thai Ha, Nguyen Minh Duc, Nguyen Tuan Linh, Nguyen Duc Thuan

Hanoi University of Science and Technology - No. 1, Dai Co Viet, Hai Ba Trung, Hanoi, Viet Nam

Received: November 02, 2016; Accepted: November 03, 2017

Abstract

ECG signal tele-monitoring that uses WLAN has been investigated and developed in recent years in order to enhance the efficiency of cardiovascular disease treatment and monitoring. However, there are several problems that may occur during the ECG signal transmission-receiving process, including bit errors and data packet loss. The causes of such problems include obstructions signal weakening due to far distances, larger number of users accessing the network at the same time, the presence of different WiFi networks operating at the same frequencies as that of ECG-WLAN and especially transmitted data packet sizes, of which significant changes will result in data loss on the transmission line. ECG signal compressing before transmitting is, therefore, critical. A two-state ECG compression algorithm has been proposed by the same group of authors in [1] and [2]. This paper describes the experiment and evaluation of the efficiency of the proposed WLAN-based two-state ECG compression algorithm. ECG data used in this experiment was from an arrhythmia ECG database. The results of experiment were evaluated by comparing some parameters such as packet error rate-PER, signal throughput, percentage of RMS difference (PRD), PRD Normalized (PRDN) and time delay between compressed and uncompressed signals and showed that the two-state compression algorithm helped to improve efficacy in ECG signal- WLAN transmission.

Keywords: Two-state ECG compression, packet error, throughput, Wireless LAN, ECG data

1. Introduction

Cardiovascular disease is among the most fatal diseases, with the sudden death rate up to 50% [1]. Therefore, quick, on-time and efficacious responses from doctors to any suspected symptoms are vital. That leads to the development of a number of ECG tele-monitoring systems using WLAN (wireless local area network) [1-5], a simple, low-cost and common wireless network being currently in operation in most of the hospitals in Vietnam and over the world.

WLAN offers a bandwidth up to 100MHz and a data rate of 600Mbps (IEEE 802.11n) [6,7] that is suitable for monitoring a large number of patients from distance. However, network congestion can still occur when too many devices accessing the network in one time causing delay in data transferring. Furthermore, noise interference from surrounding environment can likely be serious enough to cause errors in data or data packet loss. A common solution for tackling these problems is to compress the ECG at the transmitters which helps reduce the bandwidth usage, the transmission delay, the network congestion and errors in data [8]. Recently, we have developed a compressing method, called two-state ECG compression algorithm, which achieves a good balance between high compression ratio and low error in decompression [9,10]. This algorithm's advantage lies in its flexibility, with its compression ratio being adjustable to suit the WLAN condition. The concept of the algorithm is to divide the ECG into two periods: (i) the simple period including P and T waves (low-frequency fluctuation) compressed with higher compression rate ***hCR***, and (ii) the complex period including QRS complex (high-frequency) compressed with lower compression rate ***lCR*** [9,10]. In publication [10], the algorithm was tested with different pair***h CR-lCRs*** of 2-2, 4-2, 6-2, 8-2, 10-2, 6-3, 9-3, 12-3, 15-3, 8-4, 12-4, 16-4, 20-4, 10-5, 15-5, 20-5 and 25-5 in 48 MIT-BIH ECG recordings and 9 CU ventricular tachyarrhythmia ECG recordings [11]. The most compromised compression ratio in the test was 15-3 with good compression ratio, low PRD (Percentage RMS Difference) and low distortion in P, R and T peaks (low PMAE) [10].

In this paper, the algorithm is tested in the realistic WLAN transmission environment with different compression ratios and different packet lengths. The concept of the experiment will be presented in the next section.

2. WLAN ECG Transmission experiment using Two-state ECG Compression Algorithm.

* Corresponding author: Tel: (+84) 967008876
E-mail: luong.duongtrong@hust.edu.vn

2.1. The experiment scenario

In this experiment, the simple Client-Server topology is used for testing the ECG compression algorithm. The model of a Client-Server WLAN ECG transmission using the compression algorithm is shown in Fig.1. The ECG database is the ECG arrhythmia recording set, which can be found in [11]. In *the transmitter (Client)* side, after the signal processing stage to reduce the power-line noise and baseline drift from the signal, the ECG is compressed using the two-state compression algorithm [10] and then fed to a WiFi module (integrated in Client) and sent to a router. In *the receiver (Server)* side, the ECG signal received from the router will be decompressed for storing and/or displaying.

The router used in the experiment is the Tenda W311R with the transmit power of 17dBm and gain of 5dBi. The reception sensitivity to different rates is: 54M:-74dBm@10% PER; 11M:-85dBm@8% PER; 6M:-88dBm@10% PER; 1M:-90dBm@8% PER. The Server is a Dell Vostro 2440 (Intel Core i5-3210M @ 2.50GHz, Windows 10 64bit), while the Client is aDell Inspiron 3548 (Intel Core i5-4200@1.6GHz, Windows 7 64bit, card Wifi Intel Wireless-N 7260). The transmit power of the Client is set at 20dBm.

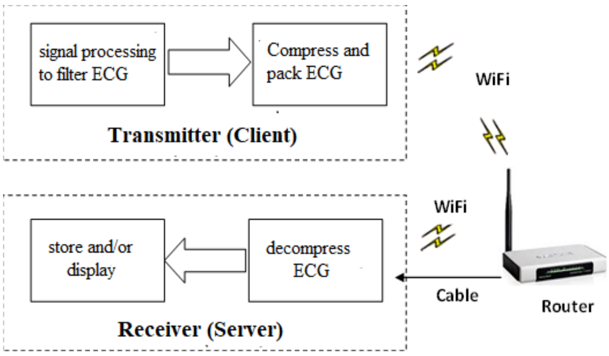
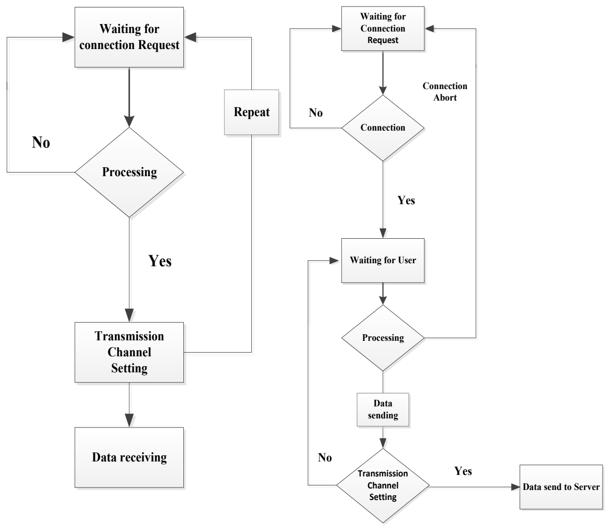


Fig. 1. The model of the Client-Server WLAN ECG transmission experiment with decompression.

The most widely used WLAN standard in Vietnam is IEEE 802.11 due to its flexibility, simplicity and low cost [4]. There are four sub-standards in IEEE 802.11 including 802.11a, 802.11b, 802.11g and 802.11n. 802.11a uses the transmit frequency of 5GHz which is not popular in Vietnam, while 802.11n is complicated and high-cost compared to other three, although it can uses both transmit frequencies of 5Ghz and 2.4GHz [6]. The rest two (802.11b and 802.11g) use only 2.4GHz but they provide good noise immunity (802.11b), good data rate and throughput (802.11g) [12]. Meanwhile, ECG signal with a normal sampling rate of 200-500Hz and the sample resolution of 11-16bit [5] often requires such characteristics of 802.11b and 802.11g to be preserved in the Server side. As a result, the combined topology 802.11b/g is used for the experiment. We only intervened the Application layer (in the OSI model) since testing the compression algorithm is our sole focus. We also chose both TCP and UDP protocol in transmission depending on the kind of package. TCP protocol though possesses a mechanism of controlling congestion, such mechanism increases significantly the transmission delay when used in a noisy environment (high bit error rate - BER) in hospitals [13]. Meanwhile, the allowable delay time for the real-time patient monitoring applications does not exceed 3 seconds [13]. UDP, on the other hand, guarantees the real-time application but provides no package controlling mechanism (no retransmission when an error occurs). In fact, UDP is used in most of the wireless patient monitoring devices. Hence, the package loss and the transmission delay of UDP were also evaluated in the experiment.



a) b)

Fig. 2. Handshaking stage at the Server (a) Client (b)

The process of sending-receiving is described as below:

First stage: handshaking. Server has to finish the handshaking process with Client to start a data communication channel. In this stage, Client will send information related to the compression to Server containing signal sampling rate, compressed or full signal, the two compression rates hCR-lCR if compressed signal. Because of the importance of these information, this handshaking stage is done in TCP protocol. The stage is described in Fig.2.

Second stage: data sending. In this stage, Client starts sending ECG to Server using UDP protocol. The effect of different compression ratios on the decompressed signal's quality will be evaluated here.

Third stage: end the channel. A notification package is sent from Client to Server using TCP protocol to close the channel

2.2. Experiment setup and parameter selection

2.2.1. Package length and compression ratio selection

It is obvious that there is a delay in getting all the samples before operating the compression. However, this delay time is counted into the overall transmission delay which is not allowed to exceed 3 seconds in the real-time applications [13]. Choosing a small package length each time helps reduce the bit error [8] that leads to the package loss in the Application layer [6]. However, too small packages on the other hand increase the header-data length ratio and hence waste the bandwidth resources. In the experiment, two different package lengths of 90 samples and 180 samples are chosen, which are not too small and whose delay times are 90*1/360=0.25s and 180*1/360 = 0.5s (the sampling rate is 360Hz). The process of communication regarding the package length is shown in Fig. 5.

In terms of compression ratio, a number of pair hCR-lCRs have been validated in [9] to have acceptable errors in decompression including hCR-lCR={4-2, 6-2, 8-2, 10-2, 6-3, 9-3, 12-3, 15-3}. These ratios are also picked for this experiment.

There are 500 packages of ECG in total in one experiment of transmission with one compression ratio and one package length.

2.2.2.Software

Two different software are also written separately for Server and Client with their GUI shown in Fig.3 and 4.

2.2.3.Evaluating parameters

Some common parameters are often used for evaluating compression algorithms including Compression Ratio – CR, Packet Error Rate – PER, throughput, Percentage RMS Difference – PRD, Percentage RMS Difference Normalized – PRDN and amd the total delay.

- *Compression Ratio (CR)*

$$CR = \frac{\text{Bytes before compression}}{\text{Bytes after compression}} \quad (1)$$

- *Packet Error Rate (PER)*

$$PER(\%) = \frac{\text{sent packages} - \text{received packages}}{\text{sent packages}} . 100\% \quad (2)$$

The number of sent packages here is 500 packages, while the number of received packages is recorded in a .csv file generated by the Server software. The packages are numbering from 0-255 and the number is reset back to 0 after the 256th package.

For each compression rate and each package length, we repeat 10 times of the transmission and take the average PER.

According to [6], the maximum allowable package loss is 5% in the ECG real-time application, or it will lead to fault diagnosis. The cause could be some objects staying in between Client and Server, the distance

or the network congestion, and the package will be discarded if the Datalink layer cannot fix the error bits [7].

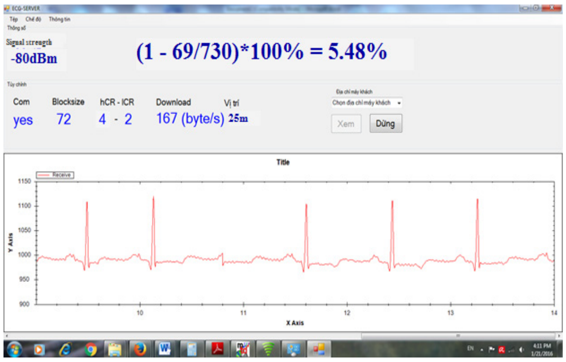


Fig. 3. GUI of the Server's software displaying PER and the pair hCR-lCR

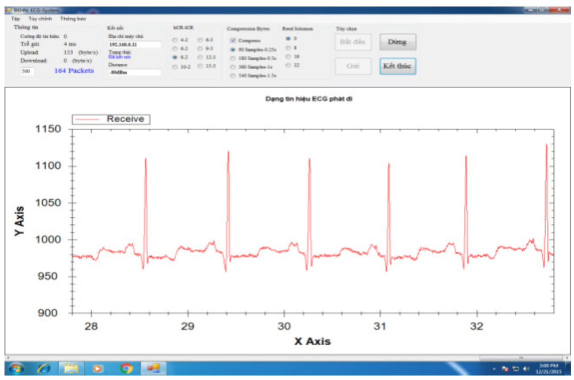


Fig. 4. GUI of the Client's software while sending ECG to Server.

- *Throughput:*

Throughput (byte/s) is calculated at the output of the Client's Application layer.

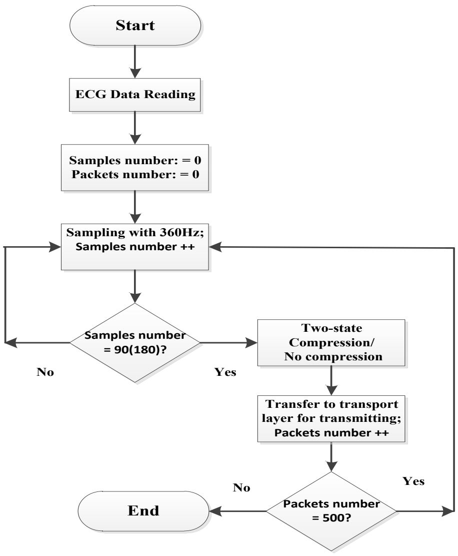


Fig. 5. The communication process regarding different package lengths

After each second, the total sent bytes is stored in .csv file generated by the Client's software. After all 500 packages are sent, an average throughput is calculated.

- *Percentage RMS Difference (PRD):*

PRD compares the original signal before compression and the signal after the decompression. In this experiment, PRD represents for the error at the final output signal at the Server caused by both the package loss and the compression algorithm itself.The lower the PRD, the better the signal is preserved [14]. Similar to PER, PRD here is taken average of the 10 transmissions for each compression ratio and each package length.

$$PRD(\%) = 100\% . \sqrt{\frac{\sum_0^{N-1}(y_i - \hat{y}_i)^2}{\sum_0^{N-1}(y_i)^2}} \quad (3)$$

$y_i, \hat{y}_i$ are respectively the ith sample of the original ECG and the ECG after being decompressed. N is the total samples.

- *Percentage RMS Difference Normalized (PRDN)*

PRDN is another parameter for evaluating the error and is calculated as in (4).

$$PRDN(\%) = 100\%. \sqrt{\frac{\sum_0^{N-1}(y_i - \hat{y}_i)^2}{\sum_0^{N-1}(y_i - \bar{y})^2}} \quad (4)$$

$\bar{y}$is the average value of the original ECG sample.

- *The total delay* (τ_{Delay})

$$\tau_{Delay} = t_{compression+packing} + t_{transmission} + t_{decompression} \quad (5)$$

The compression and packing time is calculated in Client, while the decompression time is recorded in Server. The transmission time is approximated as a half of the Round Trip Time (RTT), also called Round Trip Delay Time (RTD), which is the time finishing sending the package from the transmitter to the receiver plus the time of confirming the package arrival. All the numbers are taken average from 500 packages and stored in a .csv file.

2.2.4. Experiment setup

The system is tested in two conditions: with obstructions and without obstructions staying in between Client and Server. In the case with obstructions, Server and the router are placed inside the room 306 as shown in the Fig.6, while Client stays outside in the corridor with the objects containing the brick wall of 20cm thickness, glass door and windows with iron bars.

In the transmission without obstructions experiment, Server and Client are placed on the same line and in an open space, with the distance of 150m away from each other.

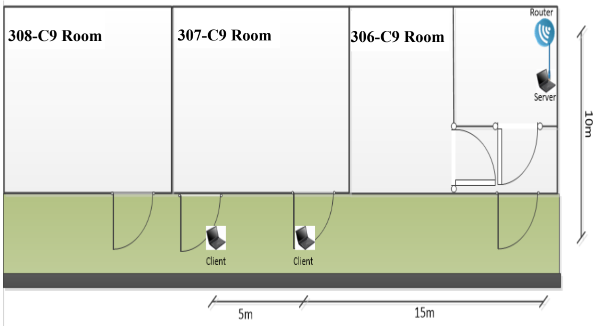


Fig. 6. The scenario of the transmission with obstructions between Client and Server.

3. Rerults

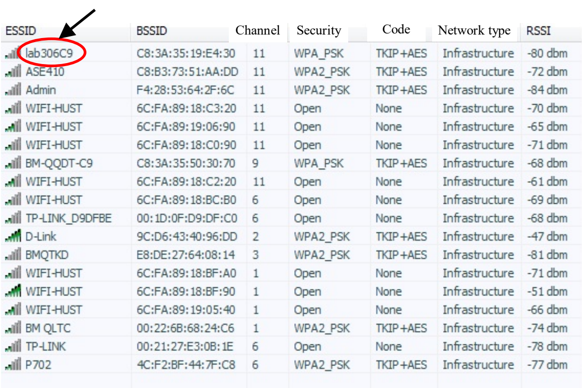
3.1. Transmission with obstructions between Client and Server

During the experiment, the results are interestingly different between the working days and the weekends.

On the working days:

The distance between Client and Server was 25m (Fig.6). The signal power at this position was -80dBm with a chosen Wifi network shown in Fig.7.

With package length of 90 samples: the average delay time ($\tau_{trễ}$) with all compression ratios was 37.3ms. It proves that the compression and decompression were simple enough for the application. With the sampling time of 90*1/360=0.25 seconds, the total delay was 287,3ms which is also accepted for a real-time monitoring. The results of the experiment with different compression ratios is presented in table 1. As can be seen, in the case of sending data without compression, the PER is 8.6%, PRD=0.62423% and PRDN = 34.3185% indicating it the worst case among others. With compression and with the increase of CR (hCR-lCR=4-2, 6-2, 8-2, 10-2, 6-3, 9-3, 12-3, 15-3), better PERs, PRDs and PRDNs were achieved. In the least CR of hCR-lCR = 4-2, PER was 7.2%, Throughput was 5 times lower (only 146 byte/s) compared the uncompressed case. While in the highest CR of hCR-lCR =15-3, PER was down to merely 1% and Throughput was nearly 1/10 of the uncompressed case. The delay time $\tau_{tr\tilde{e}}$ although fluctuated among cases, the numbers were generally low (below 52ms) which is fast enough for the real-time applications. Fig. 8 displays the ECG received and decompressed on working days, in case of the highest CR of hCR-lCR =15-3 and the signal power of -80dBm.



ESSID	BSSID	Channel	Security	Code	Network type	RSSI
lab306C9	C8:3A:35:19:E4:30	11	WPA_PSK	TKIP+AES	Infrastructure	-80 dbm
ASE410	C8:B3:73:51:AA:DD	11	WPA2_PSK	TKIP+AES	Infrastructure	-72 dbm
Admin	F4:28:53:64:2F:6C	11	WPA2_PSK	TKIP+AES	Infrastructure	-84 dbm
WIFI-HUST	6C:FA:89:18:C3:20	11	Open	None	Infrastructure	-70 dbm
WIFI-HUST	6C:FA:89:19:06:90	11	Open	None	Infrastructure	-65 dbm
WIFI-HUST	6C:FA:89:18:C0:90	11	Open	None	Infrastructure	-71 dbm
BM-QQDT-C9	C8:3A:35:50:30:70	9	WPA_PSK	TKIP+AES	Infrastructure	-68 dbm
WIFI-HUST	6C:FA:89:18:C2:20	11	Open	None	Infrastructure	-61 dbm
WIFI-HUST	6C:FA:89:18:BC:B0	6	Open	None	Infrastructure	-69 dbm
TP-LINK_D9DFBE	00:1D:0F:D9:DF:C0	6	Open	None	Infrastructure	-68 dbm
D-Link	9C:D6:43:40:96:DD	2	WPA2_PSK	TKIP+AES	Infrastructure	-47 dbm
BMQTKD	E8:DE:27:64:08:14	3	WPA2_PSK	TKIP+AES	Infrastructure	-81 dbm
WIFI-HUST	6C:FA:89:18:BF:A0	1	Open	None	Infrastructure	-71 dbm
WIFI-HUST	6C:FA:89:18:BF:90	1	Open	None	Infrastructure	-51 dbm
WIFI-HUST	6C:FA:89:19:05:40	1	Open	None	Infrastructure	-66 dbm
BM QLTC	00:22:6B:68:24:C6	1	WPA2_PSK	TKIP+AES	Infrastructure	-74 dbm
TP-LINK	00:21:27:E3:0B:1E	6	Open	None	Infrastructure	-78 dbm
P702	4C:F2:BF:44:7F:C8	6	WPA2_PSK	TKIP+AES	Infrastructure	-77 dbm

Fig. 7. Available Wifi networks at the place of the experiment.

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	8.6	724	0.62423	0	34.3185	0	47.1
Compression Ratio (hCR-lCR)	4-2	4.96	7.2	146	0.56150	0.11238	30.8699	6.18129	51.2
	6-2	6.41	4.8	113	0.37147	0.13677	20.4224	7.50603	46.1
	8-2	6.58	3.5	110	0.36435	0.14242	20.0314	7.83362	25.3
	6-3	7.39	3.1	98	0.32055	0.14271	17.6233	7.84980	30.5
	10-2	8.23	2.6	88	0.32458	0.14811	17.8448	8.12873	43.2
	9-3	8.83	1.7	82	0.29822	0.14965	16.3953	8.23143	24.9
	12-3	8.94	1.3	81	0.27165	0.17152	14.9347	9.43433	51.3
	15-3	9.92	1.0	73	0.23130	0.18248	12.7164	10.01458	15.7

Table 1. Results with the packet length of 90 samples on working days and with obstructions between Client and Server.

With package length of 180 samples: the average delay time ($\tau_{tr\tilde{e}}$) with all compression ratioswas 35ms, similar to the case of 90-sample package. With the sampling time of 180*1/360=0.5 seconds, the total delay was 535ms, still low compared the maximum allowable delay time of 3 seconds. The results in this package length case can be found in table 2 with a same trend in 90-sample package case. It is, along with the increase of the CR, the lower PER, Throughput, PRD and PRDN we get. Throughput results were lower than in 90-sample case that is understandable because of the lower number of headers throughout the whole transmission.

In order to observe a heavy distorted signal, we increased the objects by placing Client in the Room 306 (Fig.6) and repeated the transmission with the package length of 90 samples and hCR-lCR = 15-3. The distorted ECG in this situation is shown in Fig.9, with PER of 16.28% and the signal power of -90dBm only. Some periods went missing which was the result of the transmission only, not of the compression algorithm.

Table 2. Results with the packet length of 180 samples on working days and with obstructions between Client and Server.

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	5.8	722	0.51969	0	28.5714	0	42.8
Compression Ratio (hCR-lCR)	4-2	6.02	5.2	120	0.48263	0.11608	26.5337	6.38204	58.1
	6-2	8.20	4.1	88	0.34324	0.12591	18.8706	6.92260	34.1
	8-2	8.49	3.1	85	0.31830	0.13657	17.4996	7.50865	24.2
	6-3	8.91	2.3	81	0.29123	0.14611	16.0112	8.03295	39.5
	10-2	10.03	2.5	72	0.32094	0.14698	17.6448	8.08057	28.4
	9-3	11.11	1.7	65	0.30951	0.15346	17.0160	8.43721	51.6
	12-3	11.65	1.5	62	0.28184	0.16679	15.4949	9.17010	17.3
	15-3	12.45	0.8	58	0.22335	0.17598	12.2796	9.67531	21.8

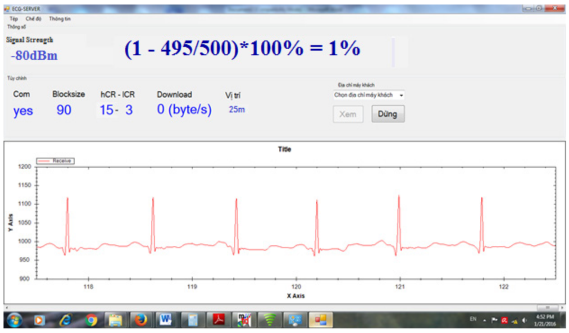


Fig. 8. ECG after being decompressed in the experiment with obstructions between Client and Server on working days. The package length was 90 samples, the compression ratio was hCR-lCR = 15-3 and the signal power was -80dBm.

On weekends:

In the same position with the signal power of -80dBm (the distance between Client and Server is 25m), PERs were found to be small (below 5%) even in the uncompressed case, so it is hard to recognize the effect of the compression algorithm. Hence, Client was moved further to a distance of 30m (Fig.6) where the signal power was -85dBm. The results corresponding with the packet length of 90 and 180 samples are respectively presented in the table 3 and 4.

Fig. 9. Distorted ECG when increasing obstructions between Client and Server.

Table 3. Results with the packet length of 90 samples on weekends and with obstructions between Client and Server.

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	8.0	724	0.60662	*0*	33.3503	*0*	22.9
Compression Ratio (hCR-ICR)	4-2	4.96	7.4	146	0.57021	*0.11238*	31.3489	*6.18129*	50.9
	6-2	6.41	5.8	113	0.43920	*0.13677*	24.1463	*7.50603*	57.2
	8-2	6.58	2.3	110	0.29707	*0.14242*	16.3322	*7.83362*	25.6
	6-3	7.39	2.3	98	0.30841	*0.14271*	16.9557	*7.84980*	52.73
	10-2	8.23	1.9	88	0.25989	*0.14812*	14.2882	*8.12873*	47.32
	9-3	8.83	1.7	82	0.30177	*0.14965*	16.5906	*8.23143*	52.63
	12-3	8.94	1.1	81	0.24063	*0.17152*	13.2296	*9.43433*	10.06
	15-3	9.92	0.6	73	0.21838	***0.18248***	12.0059	***10.01458***	14.47

Table 4. Results with the packet length of 180 samples on weekends and with obstructions between Client and Server.

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	7.6	722	0.56628	*0*	31.1328	*0*	62.0
Compression Ratio (hCR-ICR)	4-2	6.02	5.4	120	0.49426	*0.11608*	27.1734	*6.38204*	24.3
	6-2	8.20	4.5	88	0.36126	*0.12591*	19.8610	*6.92260*	58.2
	8-2	8.49	3.3	85	0.32143	*0.13657*	17.6717	*7.50865*	55.5
	6-3	8.91	2.4	81	0.29496	*0.14611*	16.2163	*8.03295*	21.2
	10-2	10.03	2.3	72	0.31895	*0.14698*	17.5354	*8.08057*	52.5
	9-3	11.11	1.5	65	0.29511	*0.15346*	16.2244	*8.43721*	27.5
	12-3	11.65	1.3	62	0.26411	*0.16679*	14.5201	*9.17010*	29.77
	15-3	12.45	1.1	58	0.24073	***0.17598***	13.2348	***9.67531***	26.0

As clearly seen in table 1, 2, 3 and 4, with both options of package lengths in the transmission with objects between, the compression ratio of 15-3 produced the best outcome of CR, PER, Throughput, PRD and PRDN.

3.2. Transmission without obstructions

In this experiment, the distance between Client and Server is 150m where the measured power signal is -78dBm in average but there was a big fluctuation in the power signal. Table 5 and 6 exhibits the results with the package length of 90 and 180 samples on working days.

Table 5. Results with the packet length of 90 samples on working days and without obstruction*s* between Client and Server

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	6.0	724	0.53468	*0*	29.3951	*0*	32.6
Compression Ratio (hCR-ICR)	4-2	4.96	5.2	146	0.48632	*0.11238*	26.7364	*6.18129*	34.7
	6-2	6.41	3.8	113	0.33204	*0.13677*	18.2546	*7.50603*	41.4
	8-2	6.58	3.0	110	0.31278	*0.14242*	17.1957	*7.83362*	51.3
	6-3	7.39	3.2	98	0.33015	*0.14271*	18.1507	*7.84980*	17.3
	10-2	8.23	2.6	88	0.32784	*0.14812*	18.0237	*8.12873*	19.5
	9-3	8.83	2.4	82	0.33023	*0.14965*	18.1551	*8.23143*	36.8
	12-3	8.94	1.8	81	0.29854	*0.17152*	16.4128	*9.43433*	25.6
	15-3	9.92	1.2	73	0.24840	***0.18248***	13.6563	***10.0145***	20.5

Table 6. Results with the packet length of 180 samples on working days and without obstructions between Client and Server

		CR	PER (%)	ThroughPut (Bytes/s)	PRD		PRDN		τ_{Delay} (ms)
					Compression + Transmission	Only Compression	Compression + Transmission	Only Compression	
No Compression	-	1.00	5.4	722	0.50132	*0*	27.5611	*0*	25.6
Compression Ratio (hCR-ICR)	4-2	6.02	5.1	120	0.48023	*0.116086*	26.4016	*6.38204*	39.3
	6-2	8.20	4.2	88	0.35026	*0.125918*	19.2562	*6.92260*	51.7
	8-2	8.49	3.8	85	0.37127	*0.136578*	20.4113	*7.50865*	44.7
	6-3	8.91	2.9	81	0.31961	*0.146114*	17.5712	*8.03295*	52.1
	10-2	10.03	2.1	72	0.29312	*0.146981*	16.1149	*8.08057*	54.6
	9-3	11.11	1.4	65	0.27691	*0.153468*	15.2237	*8.43721*	20.8
	12-3	11.65	1.2	62	0.25038	*0.166799*	13.7651	*9.17010*	13.4
	15-3	12.45	0.8	58	0.22653	***0.175988***	12.4539	***9.67531***	15.1

From table 5 and 6, a same trend with the transmission with obstructions can be observed, where PER, Throughput, PRD and PRDN all decreased when increasing the compression rate from 4-2 to 15-3.

4. Conclusion

In this paper, an experiment for testing the efficacy of the two-state ECG compression algorithm in a WLAN transmission is presented. The Client-Server topology was used using TCP protocol for handshaking process and UDP protocol for sending data from Client to Server, which is also used widely in many available wireless monitoring devices. In the experiment, PER, Throughput, PRD, PRDN and delay times were used to evaluate the effect of the compression on the transmission in two situations: with obstructions between Client and Server and without obstructions. The ECG arrhythmia recordings were used as database for the experiment. From the table 1 to 6, the results confirm that the compression algorithm helps both reduce the errors and the bandwidth usage with lower PER, PRD, PRDN and Throughput. The higher the compression ratio the better the results were. Besides that, the package loss which could be observed somewhere in the results was solely due to the transmission, not the compression algorithm. Also, the algorithm was proved its flexibility when adapted to different package lengths (90 and 180 samples). In conclusion, the two-state compression algorithm is proved to be ready for the practice.

References

[1]. Cai Ken, Liang Xiaoying "Development of WI-FI Based Telecardiology Monitoring System". Intelligent Systems and Applications (ISA), May (2010), pp.1-4.

[2]. Duong Trong Luong, Nguyen Duc Thuan, "Evaluation of packet errors in wireless electrocardiogram (ECG) signal transmission system using wifi technology". Journal of Science and Technology. Technical Universities, Vietnam, No.96, (2013), pp.40-46.

[3]. Upkar Varshney, "Patient monitoring using infrastructure-oriented wireless LANs". Int.J. Electronic Healthcare, Vol. 2, No. 2, (2006), pp.149-164.

[4]. Kyungtae Kang, Kyung-Joon Park, Jae-Jin Song, Chang-Hwan Yoon, and Lui Sha "A Medical-Grade Wireless Architecture for Remote Electrocardiography". IEEE transactions on information technology in biomedicine, vol. 15, No.2, (2011), pp.260-267.

[5]. Kyungtae Kang, Junhee Ryu, Junbeom Hur, and Lui Sha, "Design and QoS of a Wireless System for Real-Time Remote Electrocardiography". IEEE journal of biomedical and health informatics, vol. 17, no.3, (2013), pp.745-755.

[6]. Rachana Khanduri, S.S. Rattan, "Performance Comparison Analysis between IEEE 802.11a/b/g/n Standards". International Journal of Computer Applications, Vol. 78, No.1, (2013), pp.13-20.

[7]. Kenguka M. Kenguka, Atindimile S. Kumchaya, "Improving WLAN performance with enhanced mac, node cooperation and two-stage FEC scheme". Journal of Theoretical and Applied Information Technology, (2006), pp.14-20.

[8]. Jun Yin, Xiaodong Wang and Dharma P. Agrawal "Optimal Packet Size in Error-prone Channel for IEEE 802.11 Distributed Coordination Function". WCNC / IEEE Communications Society, (2004), pp. 1654-1659.

[9]. Duong Trong Luong, Nguyen Minh Duc, Nguyen Tuan Linh, Nguyen Duc Thuan, "A novel two-state ECG compression Algorithm used in telemedicine". Tạp chí khoa học và công nghệ các trường Đại học 109 (2015), pp.022-027.

[10]. Duong Trong Luong, Nguyen Minh Duc, Nguyen Tuan Linh, Nguyen Thai Ha, Nguyen Duc Thuan, "Advanced Two-State Compressing Algorithm: A Versatile, Reliable and Low-Cost Computational Method for ECG Wireless Applications". International Journal of Soft Computing and Engineering (IJSCE), Volume-5 Issue-6, (2016), pp. 56-70,

[11]. http://physionet.org/cgi-bin/atm/ATM

[12]. Mr. Sankusu Sharma, Prof.Rinku Shah, "Comparitive Study of IEEE 802.11a, b, g & n Standards". International Journal of Engineering Research & Technology (IJERT), Vol. 3 Issue 4, (2014), pp. 1846-1851.

[13]. Álvaro Alesanco and JoséGarcía, "Clinical Assessment of Wireless ECG Transmission in Real-Time Cardiac Telemonitoring". IEEE transactions on information technology in biomedicine, vol. 14, no. 5, (2010), pp. 1144-1152.

[14]. Mohammadreza Balouchestani, Kaamran Raahemifar and Sridhar Krishnan "High - Resolution QRS Detection Algorithm for Wireless ECG Systems Based on Compressed Sensing Theory". IEEE explore (2013), pp. 1326-1329.