High Compression Audio Transmission over Zigbee

Mian Tang
Northeastern University, Department of ECE
14 Mark st, Boston, US
tang.mi@husky.neu.edu

Tiancong Gu
Northeastern University, Department of ECE
14 Mark st, Boston, US
gu.t@husky.neu

Ke Wang
Northeastern University, Department of ECE
14 Mark st, Boston, US
ke1.wang@husky.neu

***Abstract* — Nowadays, the wireless sensor networks (WSN) is being applied on several fields, and wearable device is one of the most popular and typical application of WSN. Zigbee communication is taking an important role in wearable devices, such as headphone, due to the advantage of low power consumption. Although the throughput of Zigbee network is limited, audio streaming over Zigbee is highly desirable because of some application scenarios, such as rescue, emergency. And with the appearance of high compression audio codec, the voice communication of multi-user over Zigbee becomes possible. In this paper, we investigates the feasibility of voice transmission for multi-users over Zigbee network. We present an implementation of a simulation mapping from real hardware platform which allows to evaluate the capability of Zigbee network for voice streaming and to analyze important metrics, such as packet delivery ratio, delay and power consumption, with different payload size. Finally, we analyze the relation among the audio quality, power consumption and PDR in case of multi-users voice communication, and we draw hard conclusions about the adjustment of audio quality which would be needed for the improvement of voice data transmission.**

Keywords— WSN, Zigbee, 802.15.4, Audio, NS2, Codec2

1 INTRODUCTION

Wireless sensor networks (WSN) have drawn worldwide attention for many years because of the development of wireless technologies, small sensors and their inexpensive. The sensor nodes can measure and collect information from the environment, and does limited signal processing and computing, then send data back to the user. In recent years, wearable devices becomes more and more popular, and the communication among wearable devices takes an important role, that is a typical application of WSN.

Wearable devices give human a new way to interact with world. And different kind of wearable devices appear in the market, such as iWatch, Google Class, Headset, and so on. These devices can be used to sense the condition of body, collect the data, control the devices, etc. One of the most promising fields of wearable technology is voice transmission among wearable devices, or between wearable device and other facilities. For example, using the Google Glass, the user can monitor the home and control home appliances with a few voice commands [1]. Another exciting application is the voice chat in a long distance among people with wearable devices, such as headset. It's mainly used for outdoor activities. Mountain climbers, skiing amateurs or forest guardians can keep in touch with their team and ask help from others when someone is injured. These applications show the importance of audio transmission over wireless network. Additionally, wearable devices, such as headset, make the voice communication easier. With mic integrated into headset, it can sense and record the voice message anytime. The audio data is sampled in PCM format with different sampling rate, and compressed for transmission. With wireless transceiver integrated, the wireless network can be built among headsets, such as mesh network. This makes feasible the voice communication over wireless network among wearable devices. On the other side, audio transmission is different from common data transferring due to some challenges:

- In general, the common data is in small size. It is used to describe the environment, such as temperature, alert, etc. The size is only 8-bit or 16-bit. However, audio data is streaming which the size is much bigger.
- The sampling frequency is very low for common data. For instance, we may check the temperature every one second. Especially, for event alert, we just transmit the event to user only if the alert is detected. On the contrary, Audio data needs high sampling frequency, at least 8KHz in order to keep the quality of voice.
- Codec Algorithm is used to make the audio streaming in smaller size for transmission. However, the computation of the algorithm also consumes power. More complicated the algorithm of codec is, more power consumed. And the choice of the algorithm also depends on the computational ability of embedded processor.
- Time-constraint is another concern for audio transmission. More delay may be generated due to the multi-hop of wireless network, and can be worse if more congestion appears. It decreases the quality of voice service.

The wearable devices are power sensitive, and the power is supplied by the battery. The capacity of power depends on the size of the battery. Nevertheless, bigger size of battery brings more weight. Thus, there is a trade-off between the capacity of battery and the weight. And since the small size of battery is expected, the power consumption becomes the key concern of wearable devices. Audio transmission consumes power mainly by the wireless transceiver. So far, different kind of wireless standard can be used for audio transmission, such as Bluetooth, Wifi, Zigbee, etc. The Bluetooth can support high-quality audio transmission with low energy, especially the appearance of BLE4.0. But the transmission distance is limited, which should be within 10 meters. It is mainly used for the audio transmission between phone and wearable devices. Wifi is another choice, and can support ad-hoc network. However, it consumes more power in comparison with bluetooth. Another wireless network standard is Zigbee, which is generally used for short range (10-100 meters, can reach up to 1000meters when using Xbee-pro module) and low power data transfer. The network organization is flexible. It can be star, tree or mesh network. Its low data rate with very low power consumption and low cost significantly meet the requirements of audio transmission with wearable devices. And it can obviously prolong the battery life of wearable devices.

Zigbee standard is maintained by Zigbee Alliance and under the IEEE802.15.4 working group [2]. Although in most conditions, Zigbee network is used for the simply sensing and reporting activities, there is the need to support audio/voice streaming in limited network size, and it would be applicable for the communication among wearable devices.

Overview on IEEE 802.15.4: This section provides a brief introduction of the IEEE 802.15.4 standard focusing on the relevant parameter to our researches. IEEE 802.15.4 is a branch of the IEEE family and it supports the LR-WPAN, which is a simple, low-cost communication network. And the main advantages of the LR-WPAN are ease of installation, reliable data transfer, extremely low cost and a longer battery life. As for the components of the protocol, there are two different kinds of device can participate in an IEEE 802.15.4 network: full-function device (FFD) and reduced-function device (RFD). F-FDs is the device can serve as PAN coordinators or coordinators, while RFDs can only serve as END devices. The IEEE 802.15.4 offers totally 27 channels in physical layer, one in the 868MHz band, ten in the 915MHz band, and the other sixteen in the 2.4GHz band. And the bit rates on these three frequency bands are 20kbps, 40bps and 250 kbps.

As for the topology of the protocol, an IEEE 802.15.4 LR-WPAN operates in either of two topologies, depending on the application requirements: the star topology or the peer-to-peer topology. In the star topology, the communication is established between devices and a single central controller, called the PAN coordinator. Besides, the peer-to-peer topology also has a PAN coordinator; however, it totally differs from the star topology in that any device is able to communicate with any other device as long as they are in range of one another. Peer-to-peer topology allows more complex network formation to be implemented, such as mesh networking topology. A peer-to-peer net-work allows multiple hops to route messages from any device to any other device on the network. In this topology, each device is capable of communicating with any other device within its radio communications range. And this is very suitable for the voice communication among multi-users.

The overall goal of this paper is to describe the audio streaming transmission of a Zigbee-like network over the IEEE 802.15.4 framework for multi-users. In particular we examine several metrics of peer-to-peer network, such as throughput, latency and packet delivery ratio, using different payload size to deliver the information. All the measurements are performed through the mapping of the real hardware.

The remainder of the paper is organized as follows. Related works are reviewed in the next section whereas the motivation is presented in Section 3. We will describe the simulation environment built for the evaluation and analysis of Zigbee network in Section 4, followed by Section 5 which illustrates the results during the analysis of the Zigbee network for audio streaming transmission. And finally Section 6 concludes the paper.

2 RELATED WORK

Audio transmission over Zigbee has been discussed before, and low-rate voice streaming communication is an attractive feature for Zigbee [3]. A system for voice streaming over WSN is fully implemented [4].

In the previous experiment, the hardware platform is based on CC2430 (SoC) provided by TI. Audio information is sampled at 8KHz and data is coded using 8bit A-law conversion. So, the bit rate of PCM data is 64Kbps after sampling. In order to compress the data to smaller size, another processor DS2165Q is used to implement the conversion from PCM to ADPCM. And the data rate is ranged from 16Kbps to 64Kbps. Finally, voice data is transmitted over Zigbee network. In this experiment, some metrics of network performance is discussed. The packet loss increases when the audio bit rate goes up from 8kbps to 16kbps. And packet loss also depends on the distance of transmission. For longer distance, there is a higher packet loss, and vice versa. Throughput is another important metrics. And due to the multi-hop, the throughput decreases exponentially from approximate 110Kbps to 10.6Kbps when number of hop is in range of 2 to 9. Based on these results, we know that voice audio data can be transmitted over Zigbee. However, due to the drawbacks of the standard specifications such as the limited packet size, audio data cannot be transferred efficiently. Another issues is the hardware improvement. Since the computational ability of the processor is limited, audio data can only be compressed in ADPCM format, which is still in big size for Zigbee network.

Additionally, voice streaming transferring is time-constrained and wireless channel can be affected by the end-to-end delays which mainly depends on multi-hope and streaming bitrate, and can cause low quality of service. It has been investigated before [5]. In order to deal with it, the audio rate is adjusted based on ADPCM algorithm. It shows the better performance with 16Kbps bitrate than 32Kbps. Another consideration is the deployment of the nodes. The position of the intermedia node which has cross-traffic between the source and the destination has a big influence on end-to-end delay.

Recently, there are some researches about the implementation and evaluation of the transmission with high compression voice over Zigbee [6]. In this way, the bitrate of voice data can be lower which can improve the performance of the communication. However, the algorithm of voice codec is more complicated than ADPCM. In order to support the computation of the algorithm, either a powerful DSP or a stronger processor is needed, such as ARM926E, TIC64xx. On the other hand, different kind of voice codec has different properties. For voice codec, there are mainly 3 categories:

- Waveform Codecs: less complex, but still higher sampling rate, such as PCM, ADPCM.
- Source Codecs: work by modelling the speech source, and parameters related to this model are transmitted.

- Hybrid Codecs: A compromise between the 2 other categories, which means lower bit rate than waveform codecs and better quality than source codecs.

The different kind of voice codecs are listed in Table 1.

Table 1. Different kind of voice codecs

Speech Codec	Algorithm	Data rate (Kbit/s)	Sampling rate (Kbit/s)	Sample size (bit)	Ref.
Waveform Codecs					
G.721	ADPCM	32	8	13/14	[7]
G.727	ADPCM	16/24/32	8	13/14	[7]
Hybrid Codecs					
IMBE	MBE	2.4-9.6			[8]
AMR	ACELP	4.75-12.2	8/16	13	[9]
SpeeX	CELP	2-44	8/16/32	8/16	[10]

There are two main concerns for choosing voice codec. One is the bit rate, and the other is royalty-free since there is a notable increasing market trend towards royalty-free and open source voice codecs. Actually, major chip manufacturers have ported these codecs in their products. Thus, SpeeX codec is a good choice from Table 1. It supports wide range of bit rate from 2Kbps to 44Kbps. When the Zigbee network is in light load, high bit rate such as 32Kbps, 44Kbps can be used in order to have high quality of voice communication. And if the network is in heavy load, lower bit rate such as 2Kbps could be selected with low quality of voice. Plus, SpeeX is open source patent-free audio compression codec, and its algorithm has been ported to many embedded platforms. Basically, there are two ways to use SpeeX codec:

- Port the algorithm to a DSP device or a co-processor, which is only used for voice data encoding and decoding.

- Select a powerful processor. Then the processor can handle the general application and voice encoding/decoding at the same time. It is feasible because the source code of SpeeX includes optimized code for a set of different processor architectures, such as X86, ARMv4/v5E only with gcc compiler.

The SpeeX codec used for voice transmission has been discussed before [11]. It shows better performance due to its lower bit rate. Nevertheless, the evaluation is limited since the communication based on multi-hop is not in the consideration. And multi-user can make the performance worse. In fact, the bit rate of SpeeX is still high if the quality of voice is necessary. And multi-user communication can generate congestion, especially when bottle-neck node exists. Even using the lowest bit rate (2Kbps), the problem is still obvious because of the limitation of payload size and the throughput in IEEE802.15.4.

Challenge of voice streaming over Zigbee

1) The throughput: The maximum throughput of 802.15.4 is 250Kbps. In short address scheme, with ACK supported, the useful bandwidth is not more than 55.6% which gives a maximum usable bandwidth of 139 Kbps [12]. Another restriction is packet size. And the payload size in MAC layer is 127Bytes. After removing the header in network and transport layer, the size of voice data is 108 bytes. For SpeeX with 11Kbps, it at least needs 12 packets for each node per second.

2) End-to-end delay: It is only evaluated based on ADPCM algorithm, not high compression codec, such as SpeeX. And end-to-end delay depends on the load of the network, the existance of bottleneck node, and the number of multi-hope. It's hard to be estimated accurately.

3) Packet Drop ratio: In general, the very low packet drop ratio is highly desirable. Even 10 percent of packet drop ratio leads to the misunderstanding during voice communication because the meaning of voice based on multi-packets, not only one packet.

So far, no reference focuses on the voice streaming capabilities of a real Zigbee network with multi-users with high compression algorithm used, analyzing the performance based on the randomly talking from different users. In this paper, we fill this gap using high compression codec under multi-user environment.

3 MOTIVATION

The popular of wearable devices recently brings more attention on the wireless communication and its application. And voice communication over wireless network among wearable devices is becoming the trend. There are some discussion and application based on the different standard wireless network, such as wifi, BLE and Zigbee. For wifi network, it includes 802.11a/b/g/ac/ad/af, which can support different throughput in different range. The wifi ad-hoc network can be implemented in a short range with high throughput, however, it consumes much more power with medium range. And that makes wifi not a good choice for the transmission among wearable devices. Another standard wireless network is Bluetooth, especiall BLE which supports low energy. But it only supports the data transferring in a very short range. Thus, Bluetooth is mainly used for the communication between the wearable and phone or PC. Finally, Zigbee comes into public's eyes due to its low power consumption and the medium transmission range. In general, it support 100meter, and it can be extended to 300meter with stronger transmit power, such as XBee-pro. Additionally, the organization of network is flexible. Multi-users with wearable devices can build their own network for voice communication without additional equipment.

The drawback of Zigbee is the limited packet size and throughput. And this can be fixed by using high compression voice codec. Here we are using Codec 2, which supports higher compression ratio [13]. Codec 2 is mainly used for voice data encoding/decoding. It's an open source speech codec designed for communications quality speech between 700 and 3200bit/s. It fills a gap in open source speech codec beneath 5000bit/s. For the highest bit rate, it only needs to transfer 5 packets per second over zigbee, and with lowest bit rate, less than 2 packets for transmission. That can be the best choice for Zigbee network.

On the other hand, higher compression ratio means the algorithm is more complicated, and lots of floating computation are involved during encoding and decoding. It may become a serious problem before. However, due to the development of embedded processor, this issue can be fixed up. Because of the strong requirement of wearable device from the market, many IC design manufacturers have developed stronger processor with lower power consumption, or support different level of work mode in order to save power. And the related embedded platforms have been designed. For instance, BeagleBone Black platform based on TI AX335x processor, which is ARM Cortex-A8 running up to 1GHz. The platform also support 512MB DDR3 RAM, and it has NEON floating-point accelerator integrated. Another strong embedded platform is Raspberry Pi 1/2/3. The processor could be ARM1176JZF-S with 700MHz, ARM Cortex-A7 with 900MHz and Qual-core ARM Cortex-A53 with 1200MHz. And Raspberry Pi platforms have been selected for the design of wearable devices by some product manufacturers. Thus, Codec 2 can run on these platforms.

Another consideration is the power consumption of the algorithm. Because of the complexity of Codec 2, it consumer more power. However, in comparison with the power consumed on wireless communication, it doesn't become the main factor any more. And due to the appearance of harvest energy technology and more research on how to collect energy in different ways, this provides extra support to wearable devices.

In summary, Codec 2 is a high compression codec which can support bitrate from 700bps to 3.2Kbps, due to its complicated algorithm. And it can be used to implement voice communication over Zigbee on wearable device. It mainly benefits from the development of embedded processor and power saving technologies. In the following section, we introduce the system architecture and build the simulation to evaluate the performance of voice communication over Zigbee, and discuss the influence to the network with different configuration of Codec 2.

4 EXPERIMENTAL SETUP

4.1 System Architecture

The system architecture of voice communication is shown below before the simulation environment is introduced.

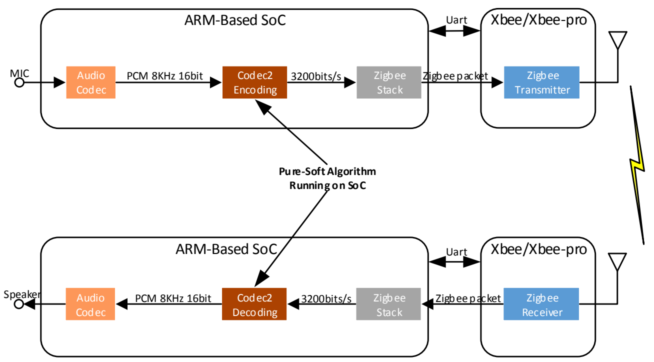

Figure 1. System architecture

For Figure 1, the voice message is sampled by audio codec through MIC, and audio stream is generated in PCM format. The sampling rate is 8KHz, and sample size is 16bits. Thus, here the bitrate of audio raw stream is 128Kbps. And then, the stream is encoded by Codec2 algorithm into frame that each is 20ms. Since the maximum bitrate of Codec2 is 3200bit/s, and 50 frames per second, we know each frames occupies 8 bytes at most. All these encoded data is wrapped into zigbee packet and sent by Xbee module. At the receiver side, message is received by Xbee module and transmitted to ARM-Based SoC. The encoded audio data is extracted from Zigbee packets and decoded by Codec2. After that, the raw audio data can be played by audio codec and output to the speaker.

Additionally, Codec2 algorithm supports different voice quality by different bitrate. This provides an opportunity of adjusting network performance of Zigbee by decreasing the traffic load, which can be implemented by reducing the bitrate for each node. So far, Codec2 consists of 3200, 2400, 1600, 1400, 1300, 1200, and 700 bit/s.

To evaluate the performance of using Codec2 for voice communication, we simulated it in ns-2 environment. Mesh network is used in order to support the communication between any two nodes, and we assume the PAN coordinator is in the center of the network in Figure 2.

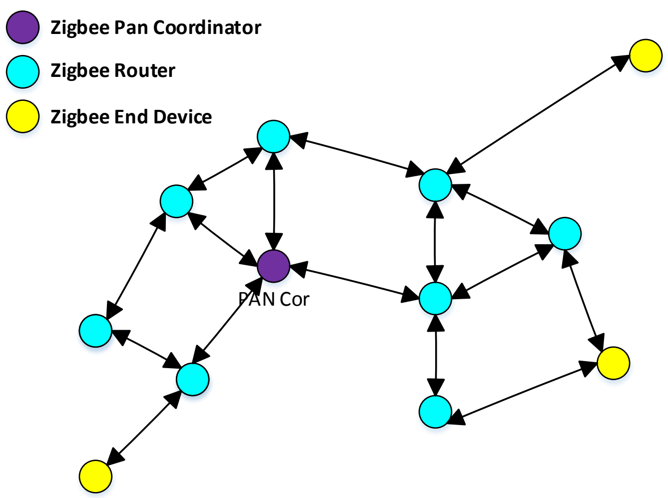

Figure 2. Mesh Network

4.2 Simulation Setup

In our simulation, there are 20 nodes which includes PAN coordinator. All nodes are scattered randomly around PAN coordinator. Since we assume the threshold of carrier sense is within 100m, so each node is less than 100m far from at least one another node. Every node runs in full functional device in Mesh network. We randomly pick up pair of nodes, one is sender to transmit the voice data, and the other is receiver to receive the data. Some metrics of network performance are measured, such as end-to-end delay, power consumption and PDR (Packet Delivery Ratio). Then, we increase the number of pairs from 1 to 5, which means at most 5 voice communication are built over Zigbee network at the same time. With different payload size, the performance is measured and discussed. After that, we increase the number of pairs to 10, and adjust the bitrate of Codec2 in order to evaluate the influence of this adjustment to the network.

The AODV routing protocol is selected for the network layer in our simulation, and UDP is used in transport layer. Above that, our application uses Exponent Traffic model, which can be configured to On/Off traffic model to simulate the unpredictable arrival of voice data.

The simple on/off model is motivated by sources that alternate between active and idle periods. An obvious example is speech where a person is either speaking or mute. The time in both the active state and idle state are exponentially distributed. The state of the source can be represented by the two state continuous-time Markov chain shown in Figure 3. In our scenario, we try to implement the worst case of voice communication. For each user, we assume the average speaking time is 5 seconds, and then take a breath which is 0.5 second of idle time, because user cannot speak all the time. Thus, the voice data traffic should be speaking for 5 seconds in average, and then mute for 0.5 second in average, and back and forth.

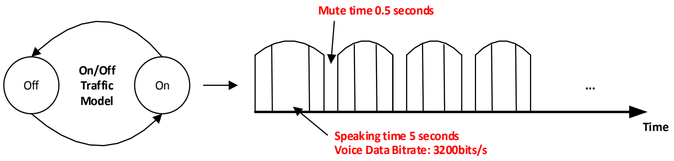

Figure 3. On/Off traffic model

In our experiments, 802.15.4 protocol is configured for PHY and MAC layer. Since this simulation does not focus on the impact of BO or SO, they are both set to be 3, a fixed number. Most of parameters are listed in Table 2.

Table 2. Simulation parameters

parameter	value
simulation time	100 seconds
Propagation Model	Two Ray Ground
Antenna Type	Omnidirectional Antenna
MAC protocol	MAC802.15.4
Physical layer Radio-type	PHY802.15.4
Network protocol	AODV
Transport layer protocol	UDP
Application	Exponential traffic
BO	3
SO	3
Energy Model	Xbee
paket size	8, 40, 80 bytes
bit rate	800, 1600, 2400, 3200 bit/s

After obtaining the simulation results, QoS parameters as packet delivery rate, average end-to-end delay and energy consumption are computed. And the voice message delay is also evaluated. Here are the definition of the metrics:

- **Packet Delivery Ratio (PDR):** It is the fraction of number of packets received to the total number of packets sent averaged by all the nodes in the network. It is the measure of reliability for a particular protocol and network used.

- **End-to-End Delay:** It is the time elapsed when a packet is sent from the source node and is successfully received by the destination node. It includes delays as delay for route discovery, propagation time, data transfer time, and inter-mediate queuing delays.

- **Energy Consumption:** In this scenario, energy consumed is measured in mW for three modes, as transmit, receive and idle. And the total power consumed is calculated for all the nodes.

- **Voice Message Delay:** It is used to evaluate the delay from being encoded at the sender side to the arrival at the receiver side.

5 Analysis

In this section, different kind of performance metrics is shown based on the simulation environment described before, and some simulation results are analyzed.

First of all, we select the different payload size based on the format of audio encoded data. As mentioned above, Codec2 algorithm is used to compress the audio stream and outputs encoded data with 3200bit/s. In fact, the data is organized in frame format. Each frame includes 8 bytes, and can be played for 20ms. Thus, for 1 second audio or voice, it includes 50 frames when 3200bit/s used. On the other hand, Zigbee protocol limits the packet length.

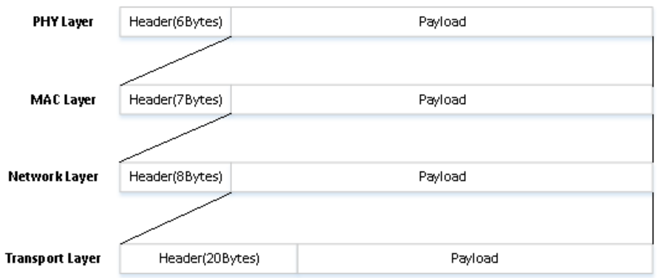

Figure 4. Packet length

In Figure 4, for PHY layer, the payload size is 127 bytes except 6 bytes of header. And In MAC layer, at least 7 bytes are used for MAC header and the payload size decreases to 120 Bytes. After removing 8 bytes of header for network layer and 20 bytes of UDP header for transport layer, the payload size decreases to 92Bytes. Thus, only 92 bytes of audio data can be transmitted in one Zigbee packet in our simulation environment. And we consider that the payload size is 8, 40, 80 bytes, which include 20ms, 100ms, 200ms of audio data respectively.

5.1 End-to-End Delay

In this experiment, we measure and compare the end-to-end delay with different number of pairs. The source node of each pair runs On/Off traffic to generate packets for transmission, and the average bitrate is 3200bit/s.

In Figure5, we show the plot of end-to-end delay. When payload size is 8 bytes, the delay is less than 0.05 second. And as the incremental of number of pairs, the delay increases up to 0.23 second. On the contrary, for payload size of 40, 80 bytes, the end-to-end delay is a little bit higher because bigger packet is transmitted each time, when only one pair exists. And as the number of pairs increases to 5, the delay is still under 0.05 second. So, the transmission with bigger payload shows better performance in end-to-end delay. It is obvious because the bitrate of transmission is almost fixed. Smaller payload size means more packets to be transmitted. When more than one source nodes exist, the traffic load increase. And since the senders and receivers are selected randomly. Each transmission may experience multi-hop. And some of intermediate nodes may need to forward packets for more than one senders. These nodes have chance to become bottleneck node. Therefore, there is high probability of congestion on them, which brings longer delay between the sender and the receiver.

At the same time, the PAN coordinator needs to synchronize all associated nodes periodically. In mesh network, all nodes are full functional devices. So, they will forward the synchronization message to other nodes. This also makes some negative influence to our application.

When payload size is 40, 80 bytes, it decreases the number of packets to be sent, and hence reduces the total traffic load of the whole network. Even with more pairs of transmission at the same time, the intermediate nodes have ability to deal with the forward for multi-sources. Thus, the end-to-end delay can be guaranteed.

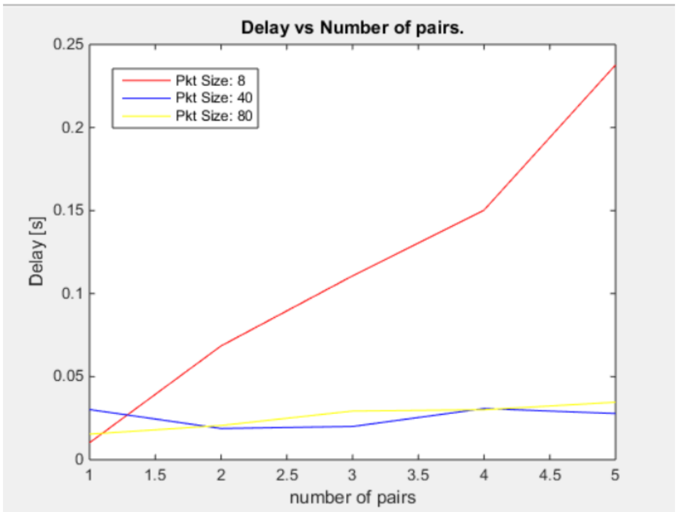

Figure 5. End-to-End delay

5.2 Packet Delivery Ratio (PDR)

Here, we measure and compare PDR with different payload size. PDR is an important metrics for voice transmission. Even a little bit low PDR may cause the misunderstanding of the voice message in the receiver side. For example, when sender transmits voice message which is about 3 seconds. The voice message is encoded into 3 x 50 = 150 frames after Codec2 algorithm with 3200bit/s used. We have known that each frame is 8 bytes, and wrapped into packets for transmission. Imagine that 10 percent of packets fails to arrive at the receiver, which means the receiver only decodes 90 percent of voice message. And that may make the receiver cannot understand the meaning of the whole voice. Thus, high PDR is expected.

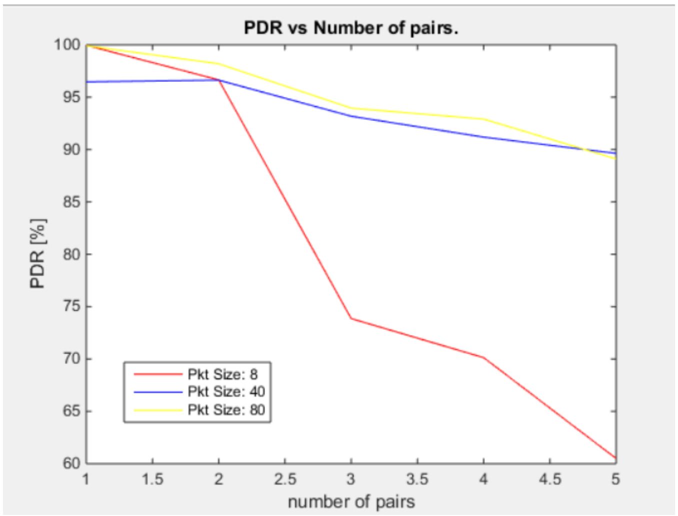

Figure 6. Packet Delivery Ratio

In Figure 6, the plot shows the low PDR when payload size is 8 bytes with more pairs. For more than 3 pairs, the PDR decreases sharply, and finally to about 60 percent. In this condition, the voice cannot be understand by the receiver totally. In Contrast, the PDR maintains above 90 percent when payload size is 40 or 80 bytes. Again, the transmission with bigger payload does better with PDR.

However, bigger payload size also means higher risk, because each packet includes more voice info. For instance, when payload size is 80 bytes, the packet includes voice info of 200ms. If one packet fails for transmission, the whole voice message cannot be understand even the success of the transferring for other packets. Hence, each packet becomes more crucial.

5.3 Energy Consumption

Energy consumption is a key metrics for voice transmission over Zigbee, and we measure and compare it based on the sum of power consumed from all nodes. Figure 7 shows the plot of different payload size. We can see that more power consumed with payload size of 8 bytes even for only one pair. And when packets are transmitted by 5 source nodes at the same time, the total power consumed is 135mW when payload size is 8 bytes. While approximate 116mW consumed with payload size of 80 bytes.

More energy consumption with small payload is not only because of more transmission needed, but also due to more headers' transferring. In our experiments, the size of total header is 41 Bytes from PHY layer to transport layer, which is the sum of PHY header, MAC header, Network header and UDP header. It has been shown in Figure 4, at the beginning of section 5, and is calculated below:

41 Bytes = 6B + 7B + 8B + 20B

When the payload size is 80 bytes, 5 packets are sent per second for voice message when 3200bit/s used, and total bytes transferred for headers is about 205 bytes (41B * 5 packet = 205B). On the other hand, when payload size is 8 bytes, 50 packets is sent per second, and total bytes transmitted for headers is about 2050 bytes (41B * 50 packets = 2050B). So, for smaller payload, more bytes of headers are transferred, which means more energy consumed. Thus, the transmission with small payload is energy inefficient and low bandwidth utilization.

Another reason of more power consumed with small payload is the congestion due to heavy traffic load, or the high probability of the existence of bottleneck nodes. That brings more retransmission in both MAC and Network layer.

For MAC layer, Zigbee uses 802.15.4 protocol, which beacon-enabled slotted CSMA/CA mechanism is adopted. Source node sends packet and wait for ACK from receiver if the packet is a MAC command or data, and only broadcast frame doesn't need ACK support. If the sender cannot receive and acknowledgement with ***maxAckWaitDuration*** time, it will retry its transmission at most ***aMaxFrameRetries*** times. When using small payload, high contention level will happen on some bottleneck nodes, where more retransmissions appear. And that consumes more power.

For Network layer, AODV routing protocol is adopted. Since AODV is reactive protocol, each source sends RREQ and waits for RREP message before its data transmission. If a RREP message cannot be received by the time ***RREQ_TIMEOUT*** expires, to find the route, source nodes will broadcast another RREQ message, up to a maximum of ***RREQ_RETRIES*** times. If more retransmissions appears in MAC layer, it also brings more delay to the Network layer. Thus, AODV has more opportunities to do retransmission for routing findings when more source nodes exist. Although AODV has a little support of congestion control, the retransmissions is inevitable sometimes, and it highly depends on the deployment of intermediate nodes, source nodes and destination nodes.

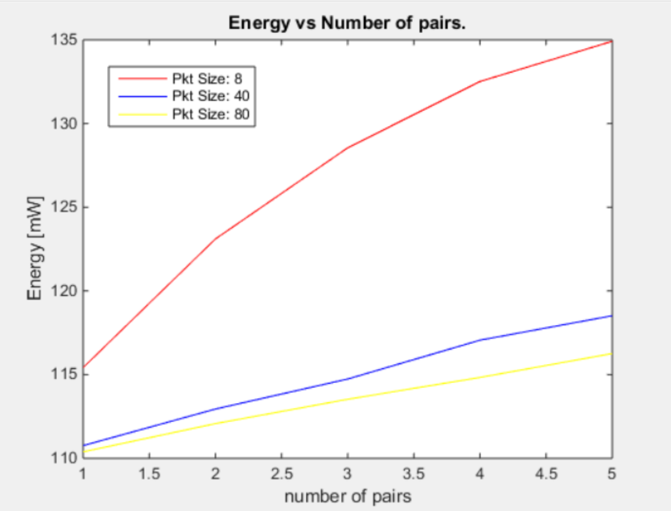

Figure 7. Energy Consumption

5.4 Voice Message Delay

Voice message delay basically is the sum of the time for voice encoding and end-to-end delay. And it highly relies on the payload size. As we mentioned above, 3200bit/s is used for Codec2, and each frame output is 20ms and 8 bytes length. If each frame is wrapped into Zigbee packet (means payload size is 8 bytes) and sent once it is generated by Codec2, it shows a good real-time performance of transmission. However, when multi-users exists, it bring low PDR, more delay and more power consumption due to its small payload size. Based on the previous simulation, we calculate the voice message delay for 5 users with different payload size, and the related end-to-end delays are from Figure 5.

Table 3. Voice Message Delay

Payload Size(Byte)	Voice Message Delay(ms)
8	240ms + 20ms * 1frame = 260
40	28ms + 20ms * 5frame = 126
80	35ms + 20ms * 10frame = 235

In Table 3, we notice the transmission delay is large with payload size of 8, and its total delay is 260ms. Nevertheless, with payload size of 80, the total delay is 235ms, which includes 25ms of transmission delay and 200ms of the time for voice encoding. The transmission with large payload shows much better performance before, however it carries more frames, which needs more encoding time. Actually, voice message delay displays characteristics of real-time. The smaller payload for transmission, the better real-time performance. Again, from Table 3, we see the shortest delay appeared with payload size of 40. Thus, there is a trade-off between the payload size and real-time performance.

5.5 Adaptive Bitrate Performance Evaluation

In this experiment, we try to evaluate the network performance with different bitrate of Codec2 algorithm. Codec2 supports bitrate from 800bit/s to 3200bit/s with low-to-high voice quality. Low bitrate means less throughput over Zigbee network, especially for multi-users. This provides us an opportunity to improve the network traffic in order to support more users' communication at the same time, and lower the power consumption of whole network. Here, we increase the number of pairs from 5 to 10, and keep payload size 80 bytes. Randomly pick source and destination nodes as before, and change the bitrate of On/Off traffic, to see the improvement of energy and packet delivery ratio.

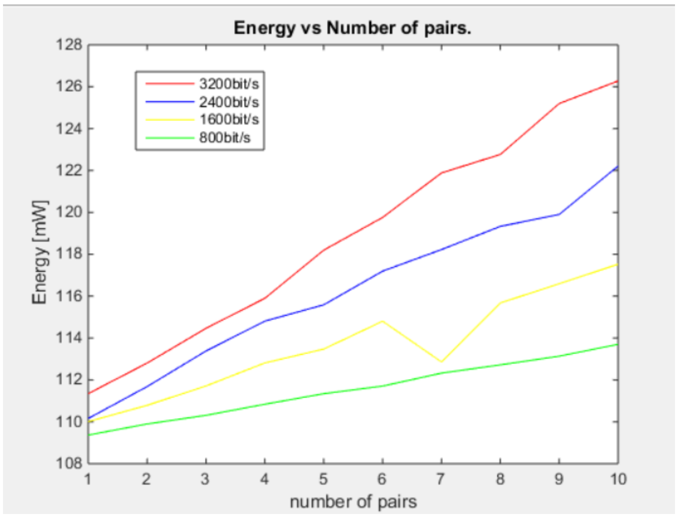

Figure 8. Energy Consumption with Adaptive Bitrate

In Figure 8, the energy consumed by the whole network is shown. From pair1 to 10, power consumed increases, and with higher bitrate, such as 3200bit/s, more power used because each source node transmits more bits per second in average. Therefore, low bitrate can help save power for transmission. When one node's power is in low level, it could use low bitrate of Codec2 to encoded the voice and transmit it. In this way, it prolongs the battery life. And the bitrate can be adjusted dynamically. Only one disadvantage is that it lower quality of voice. But it could be acceptable if the voice message received can be understand correctly by the receiver.

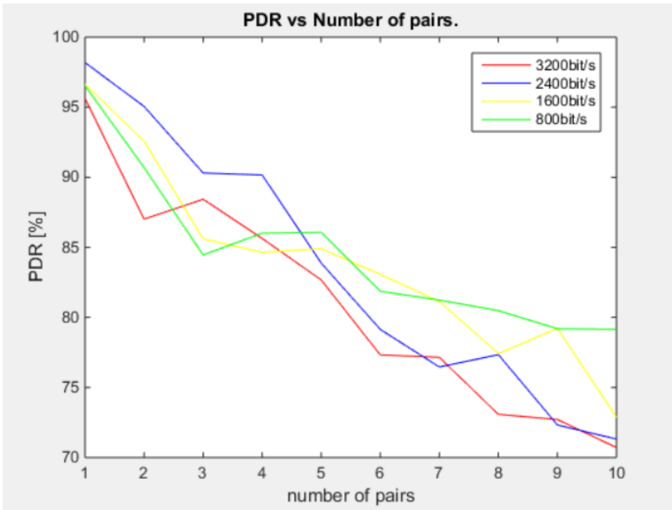

Figure 9. PDR with Adaptive Bitrate

In Figure 9, packet delivery ratios are plotted with different bitrate. As number of pair increases to 10, PDR decreases to about 70 percent with 3200bit/s. Each curve in Figure 9 shows fluctuation due to the limited run times. We selected the lowest bitrate 800bit/s, and observe a little improvement in PDR, since the total traffic load decreases for the whole network. However, it only provides limited improvement.

6 CONCLUSION

This paper presents a new implement of audio transmission over Zigbee network. Codec2 algorithm is used to compress the audio data, and the communication for multi-users is feasible and evaluated. With small payload size, the real-time performance of communication is better, however it is energy inefficient and displays low PDR with multi-users. Big payload size is preferred since it provides high packet delivery ratio, less power consumption, and shorter delay for transmission. However, each packet becomes more crucial because it contains more voice info, so high PDR should be maintained. We also investigated the improvement of network when adaptive bitrate is used. Lower bitrate used in Codec2 algorithm can reduce the energy consumed for the whole Zigbee network. However, the improvement of PDR is limited.

Reference

[1]. Chinta, R., Fong, R., Murdock, I.: A Novel Approach to Home Automation: Application Development with Google Glass (2015).

[2] IEEE Standard 802.15.4-2003, wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs).

[3] Brunelli, D., Maggiorotti, M., Benini, L.: Analysis of Audio Streaming Capability of Zigbee Networks (2008).

[4] Mangharam, R., Rowe, A., Rajkumar, R., Suzuki, R.: Voice over sensor networks. In: RTSS 2006. Proc. of the 27th IEEE International Real-Time Systems Symposium (2006).

[5] Brunelli, D., Teodorani, L.: Improving Audio Streaming over Multi-hop ZigBee Networks (2008).

[6] Meli, M., Gysel, M., Sommerhalder, M.: Using IEEE 802.15.4 / ZigBee in audio applications (2006).

[7] ITU-T. G.x, Transmission Systems and media, and digital system http://www.itu.int/rec/recommendation.asp?type=products&lang=e&parent=T-REC-G, (2005)

[8] Digital Voice Systems, http://www.dvsinc.com/, (2011).

[9] 3rd Generation Partnership Project. 3GPP Specification series. http://www.3gpp.org/ftp/Specs/html-info/26-series.htm (2015).

[10] Xiph.Org Foundation. Speex. http://www.speex.org/, (2005).

[11] Toupoulis, E., Meliones, A., Apostolacos, S.: Implementation and Evaluation of a Voice Codec for Zigbee, (2011)

[12] B. Latré, P. De Mil, I. Moerman, N. Van Dierdonck, B. Dhoedt, and P. Demeester, "Maximum throughput and minimum delay in IEEE 802.15.4," Lecture Notes in Computer Science, vol. 3794, pp. 866–876, (2005)

[13] Rowe, D.: http://www.rowetel.com/blog/?page_id=452, (2015)