Leveraging Discrete Fourier Transform to Reduce Power Consumption in Underwater Wireless Sensor Network Communications

Wafa Elmannai, Khaled Elleithy, Ajay Shrestha, Mohamed Alshibli, Reem Alataas

Department of Computer Science and Engineering
University of Bridgeport, CT

Abstract

Wireless Sensor Networks (WSNs) have become an important means of gathering environmental and physical information from a wide range of areas. WSNs could be used in underground, aboveground and underwater applications. In this work, we propose a new solution for underwater Wireless Sensor Networks to overcome the problem caused by the ionized nature of seawater. This work presents a methodology to improve the lifetime of WSNs. The wireless sensors have three main functions: sensing, processing and transmitting. The first two consume very less power compared to the third. Thus, we need to guarantee the successful transmission of signal with nominal and efficient use of power to improve the lifetime of the sensors. Improving the lifetime of these sensors will improve the experience of the end user, as the information-gathering lifetime of the sensors increases. Our validated results showed reduction in the power consumption, thus improving the lifetime and the signal loss rate.

1. Underwater Wireless Sensor Networks deployments

Historically underwater monitoring and data collection were performed by recording devices that did not communicate in real time.

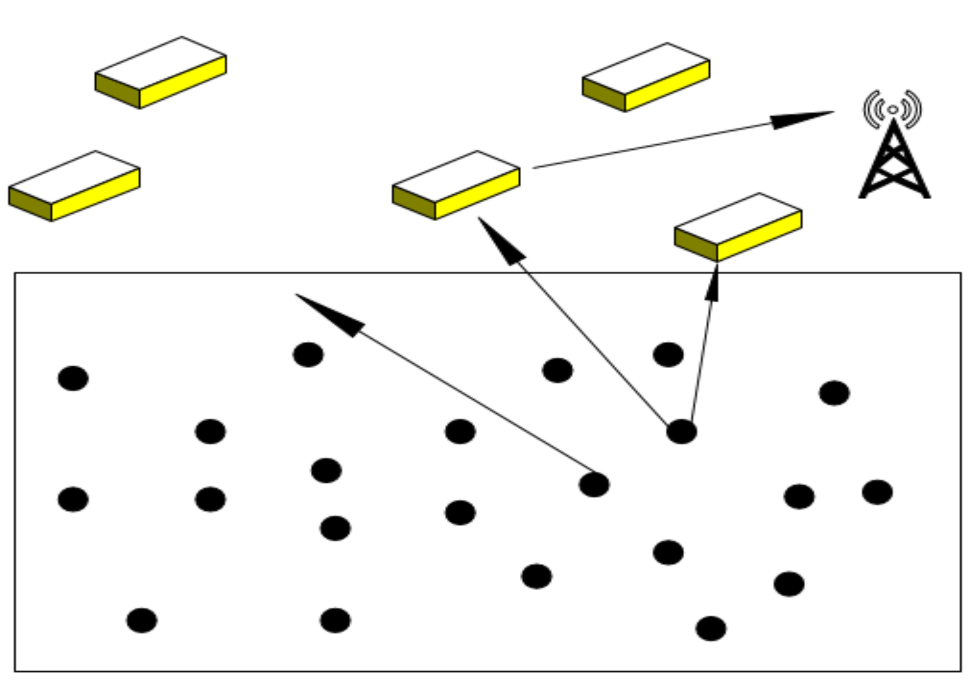

Figure 1.1: An Underwater Acoustic Tested at the University at Buffalo

The figure demonstrates the typical deployment of UWSNs. The sensors at the bottom of the ocean floor collect process and relay the data to the surface substation in the moving ship, which in turn transmit the collected information to the onshore sink.

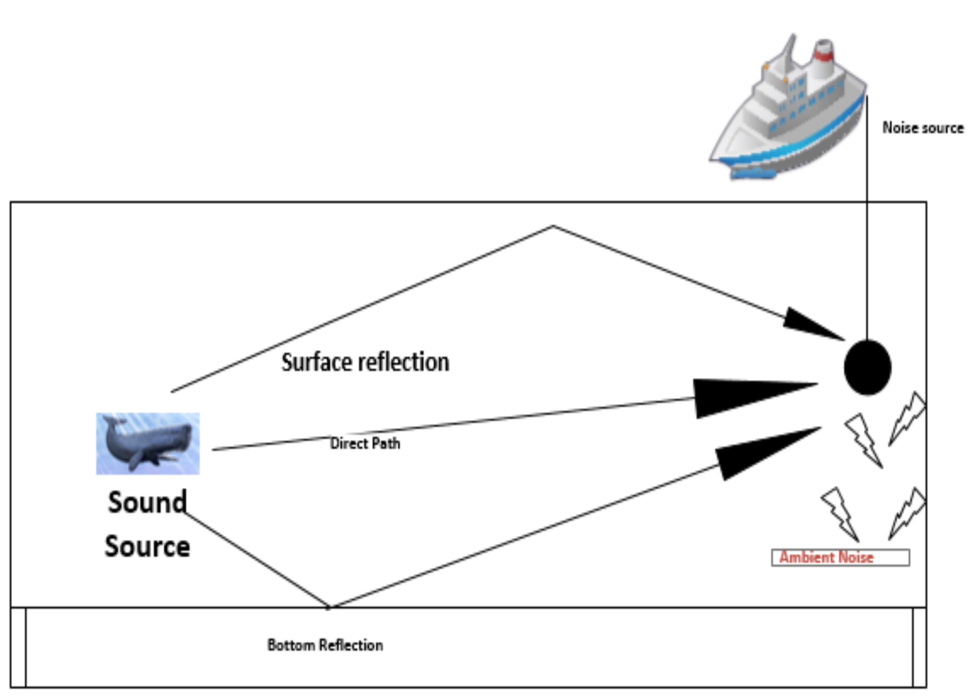

Figure 1.2: Diagram showing underwater multi-path propagation

UWSNs deployment differs from terrestrial WSN as shown in Figure 1.2 deployments in a number of ways. The UWSNs are undoubtedly more complex and vulnerable to environmental conditions. The sensors have limited battery power and are extremely difficult, if not impossible to recharge. Due to the fact that acoustic waves are used instead of RF or microwaves, the data transfer rates are very low, which is further aggravated by propagation delays. The underwater communication also suffers from multi-path propagation as shown in Figure 1.2.

2. Proposed Work

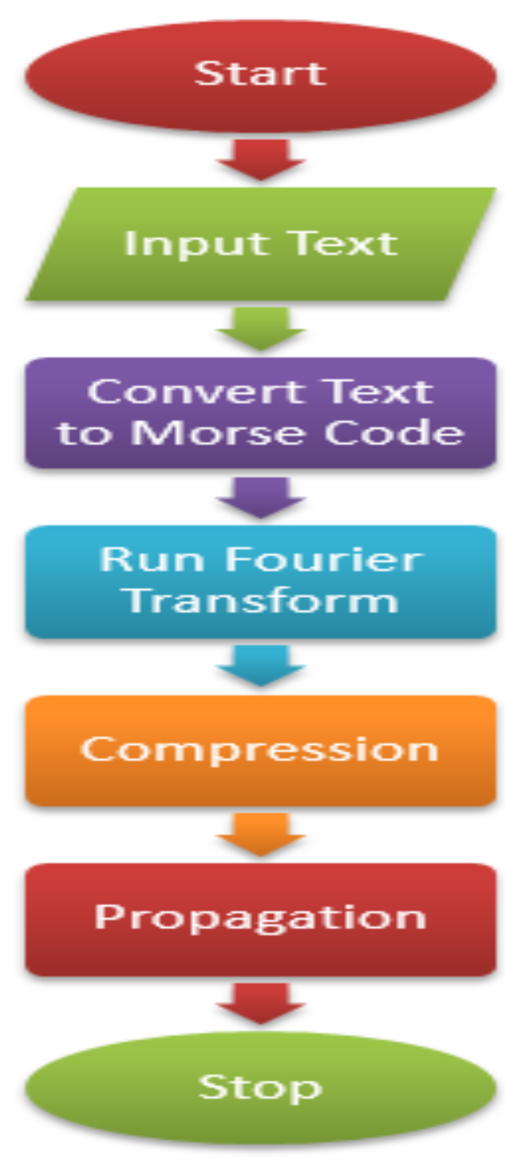

Figure 2: Flowchart of proposed model of wireless underwater sensor communication

In this work, we introduce a compression sound algorithm using Fourier series and Morse code in order to improve the wireless underwater sensor communication. Our main purpose of integrating Fourier series and Morse code is to propagate the sound at various frequencies.

A. Morse Code:

Morse code is used to convert the text to on-off tones series, clicks or lights. This can enable the observers and skilled listener to understand the meaning with no specific equipment.

B. Discrete Fourier Transform (DFT):

DFT is used to convert from time domain to frequency domain. We can work with the data in the frequency domain easier than the data in the Time domain.

C. Compression Algorithm:

Analog audio compression allows the efficient storage and transmission of sound data.

In this implementation, we use the μ-law transformation which is a basic audio compression technique. The logarithmic step spacing represents low-amplitude audio samples with greater accuracy than higher-amplitude values. Thus the signal-to-noise ratio of the transformed output is more uniform over the range of amplitudes of the input signal. The μ -law transformation equation is:

$$y=\begin{cases}255-\frac{127}{\ln(1+\mu)}\times\ln(1+\mu|x|) & for\ x\geq 0\\ 127-\frac{127}{\ln(1+\mu)}\times\ln(1+\mu|x|) & for\ x<0\end{cases}$$

Where μ = 255, and x is the value of the input signal normalized to have a maximum value of 1.

3. Simulation Set Up

Table 1 shows the initial values that can be adjusted using the implemented simulator.

TABLE I
Initializing Morse code and DFT parameters

Parameters	Initial Values
Short beep	0.04 sec
Long beep	0.1 sec
Delay	0.05 sec
Frequency	1000

The simulation is used to tune the parameters in order to improve the distance the signal propagates during transmission.

4. Simulation Results

The application was implemented in C# programming language. We used DFT to compress sound signals to a lower frequency to increase the sound propagation distance. This allowed us to control the propagation distance while keeping the security of the information transferred alive as shown in Figure 4.

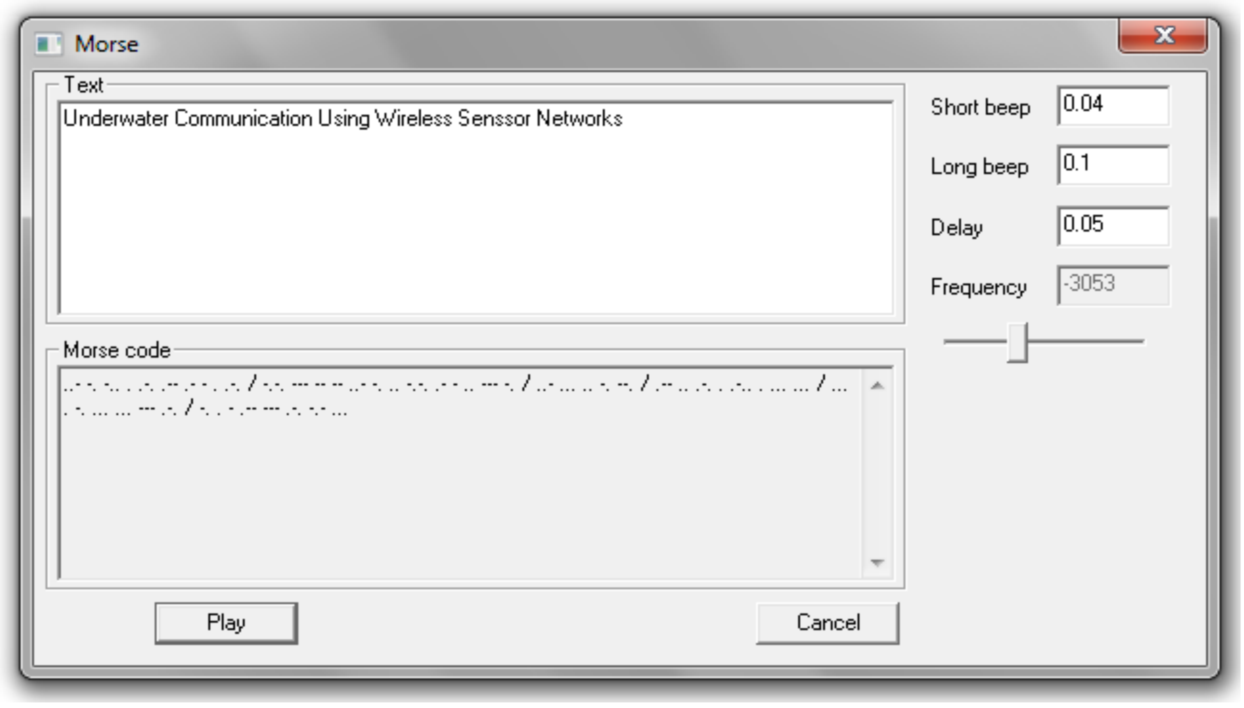

Figure 4: Text to Sound Converter based Morse code

A. Time Average versus Distance Scenario:

Figure 5 was generated for various sound frequencies over a time average versus distance propagation.

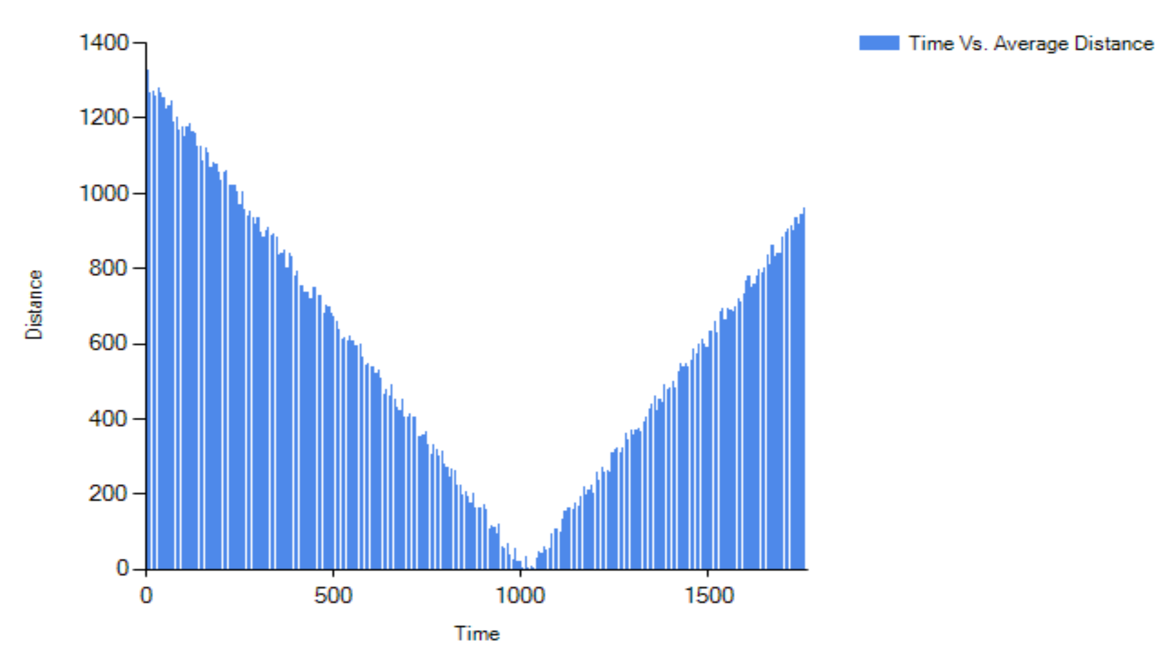

Figure 5: Time Average vs. Distance Propagation Scenario

B. Frequency versus Distance Average Scenario:

Figure 6 shows the optimal frequency required for the best propagation distance based on the frequency values.

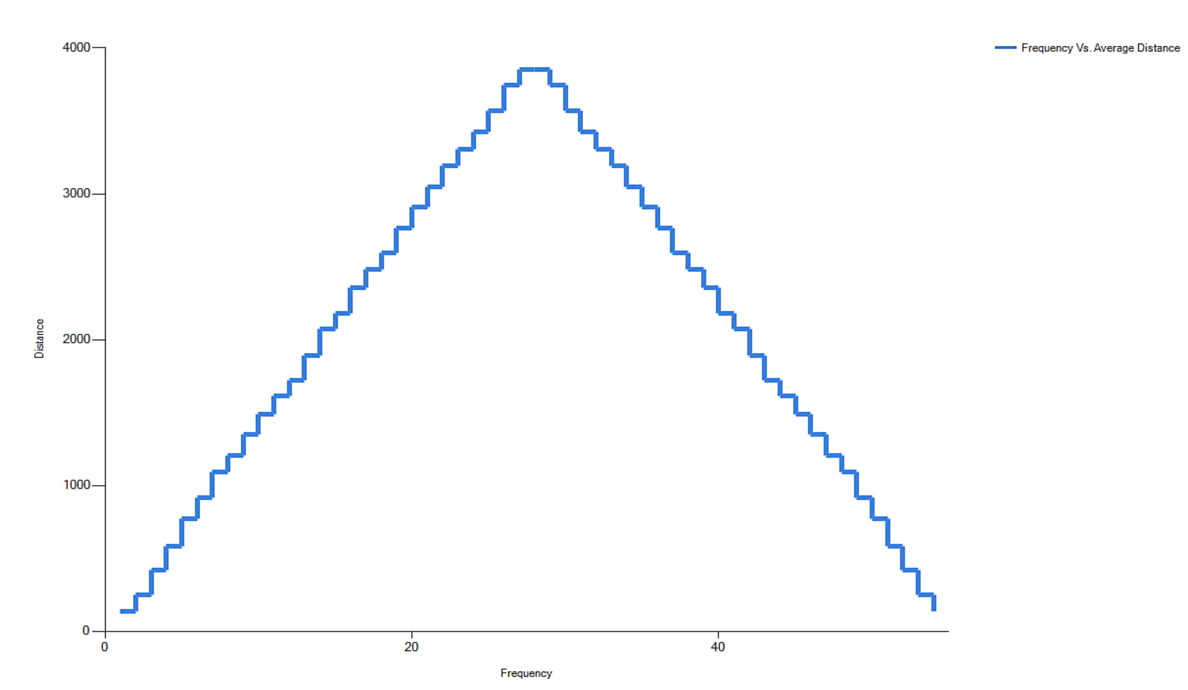

Figure6: Frequency (X-axis) vs. Distance (Y-axis) Average Scenario

5. Conclusion

In this poster we implemented an algorithm to integrate both Morse code to convert the text to series of tones and DFT to compress the sound signal. This algorithm addresses important problems regarding real-time wireless underwater communication, propagation and ionization.

Propagation distance and ionization of the seawater are some of the most challenging issues of underwater communications. Sound in seawater propagates up to a certain distance and then dissipates.

We presented an efficient algorithm using Morse code and DFT to modify the sound signal based on our requirements. The algorithm tunes the frequency to an optimal value that increases the propagation distance without increasing power consumption.