



Sonar Interface with FCS and Target Detection

Ravindra V Asundi^{1*}, Anusha L Narayanan², Archana B³, Ananya H A⁴,
C Shrunga Divakara⁵

B M S Institute of Technology and Management Avalahalli Bangalore

* Corresponding Author E-mail: ¹rva2001@gmail.com, ²anushaln96@gmail.com, ³archanabijoor@gmail.com, ⁴ananya.agile@gmail.com, ⁵cshrunga@gmail.com

Abstract

This paper presents a method to interface sound navigation and ranging (SONAR) with fire control system (FCS) and detection of targets. A submarine navigator keeps track of surrounding submarines and active targets such as torpedoes or mines and passive targets such as rocks, friend submarine or a ship, in order to have tactical information of its neighborhood. At present this is carried out manually by trained sonar operator. This task of classifying the object as either mines or rocks by training an artificial neural network using tensor flow framework, thereby giving the submarine (ship) system more autonomy.

Keywords: Sonar, Submarine, Active Sonar, Passive Sonar, FCS, Neural Networks.

1. Introduction

SONAR systems have many similarities to radar and electro-optical systems. The operation of sonar is based on the propagation of waves between a target and a receiver. Sonar differs fundamentally from radar and electro-optical systems because the energy observed by sonar is transferred by mechanical vibrations propagating in water, solids, gases, or plasma, as opposed to electromagnetic waves. The term “acoustic” refers to sound waves in any medium. Acoustic waves come in two types: longitudinal or compression and transverse or shear. The fundamental parameter of an acoustic wave is pressure. When water or air molecules are pushed or pulled apart, they exert a restoring force that resists the motion. In general, longitudinal waves are the most important type of waves in acoustics, particularly when discussing the underwater environment.

Due to severe attenuation of RF and optical signals underwater make audio signals a great way to detect targets. The audio signals generated by the sonar is serial in nature, the sonar data is used by a submarine navigator to gather tactical information of the surroundings. The transmission of serial data is achieved by using RS232. The maximum speed of transmission achieved on using RS 232 is 20Kbps. This transmission speed is low for real time use in submarines. To improve the rate of data transmission in real time scenario it is important to boost the speed of sonar signals. This project achieves increased data rates of up to 100 Mbps by using ethernet. Microcontroller TM4C1294NCPDT is coded so as to convert the serial data to ethernet.

In this paper part II describes generation of SONAR data, part III describes the reconstruction of serial data, visual cognizance, part IV describes the classification of targets and conclusion

2. Generation of Sonar Data

The sonar finds application underwater. The practical application of sonar in this project is not viable, given the complexity with which the sonar works.

To realize this project the sonar data is imitated using Qt, a cross platform application development framework.

A. Sonar Data Interface Format

The sonar exchanges data using two types of frames viz. Information and Control frame.

Four types of messages are exchanged from SONAR to Fire Control System (FCS) and they are

- (a) Sonar status message: Message 1 (5 bytes)
- (b) Active target data message: Message 2 (30 bytes)
- (c) Passive target data message: Message 3 (12 bytes)
- (d) Torpedo Data message: Message 4 (3 bytes)

Control Frame

The control frame is transmitted to check the health of the link and availability of the FCS. The structure of the sonar message is as follows:

Byte No	Bits of byte							
	7 - MSB	6	5	4	3	2	1	0 - LSB
1	TA	CL	MODE			ACT	SON	
2	0	0	SH	DTX				
3	SW				TXWF			
4	0	0	0	FR		SL		
5	RG		RS			PW		

Fig. 1: Structure of information on sonar status

Byte	Bits of byte							
№	7 - MSB	6	5	4	3	2	1	0 - LSB.
1	SV	PV	TSU	TPU	TIC			TDT
2	MSB= TRUE BEARING							LSB
3	180 deg							
4	MSB= RANGE							LSB
5	128Kms							
6	MSB= APPROACH SPEED							LSB
7	SIGN 64Kts							
8	SV	PV	TSU	TPU	TIC			TDT
9	MSB= TRUE BEARING							LSB
10	180 deg							
11	MSB= RANGE							LSB
12	128Kms							
13	MSB= APPROACH SPEED							LSB
14	SIGN 64 Kts							
15	SV	PV	TSU	TPU	TIC			TDT
16	MSB= TRUE BEARING							LSB
17	180 deg							
18	MSB= RANGE							LSB
19	128Kms							
20	MSB= APPROACH SPEED							LSB
21	SIGN 64 Kts							
22	SV	PV	TSU	TPU	TIC			TDT
23	MSB= TRUE BEARING							LSB
24	180 deg							
25	MSB= RANGE							LSB
26	128Kms							
27	MSB= APPROACH SPEED							LSB
28	SIGN 64 Kts							
29	TT2				TT1			
30	TT4				TT3			

Fig. 2: Structure of the active message

Byte	Bits of byte							
№	7 - MSB	6	5	4	3	2	1	0 - LSB.
1	0	BV	VCS	UOB	TIC			TGS
2	MSB= TRUE BEARING							LSB
3	180deg							
4	0	BV	VCS	UOB	TIC			TGS
5	MSB= TRUE BEARING							LSB
6	180deg							
7	0	BV	VCS	UOB	TIC			TGS
8	MSB= TRUE BEARING							LSB
9	180deg							
10	0	BV	VCS	UOB	TIC			TGS
11	MSB= TRUE BEARING							LSB
12	180deg							

Fig. 3: Structure of the passive message

Byte	Bits of byte							
№	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	TTV	TA
2	TRUE BEARING							LSB
3	MSB							

Fig. 4: Structure of the torpedo message

3. Ethernet – Reconstruction of Serial Data

The PCB for this application is designed using ORCAD software. It is a four-layer PCB. This application uses six out of eight UART modules present in the TM4C1294NCPDT microcontroller. MAX ICs convert RS 232 and RS 422 signal levels to CMOS logic level. It takes in the differential electrical signal and outputs a single ended signal which can be subsequently given to the UART pin of the microcontroller. It can also take in the single ended signal from the microcontroller and output differential electrical signal.

The IDE Code Composer Studio is used to code the microcontroller in embedded C. The conversion from serial to ethernet is achieved using the microcontroller. The serial data is accumulated in an array and put into the UDP frame format. UDP socket is created for this purpose.

To look for the link between the client (submarine-real time application) and the server (PCB), a software by Texas instruments Tiva board finder is used. If the link is successfully displayed the conversion is checked on Wireshark, an open source packet analyzer.

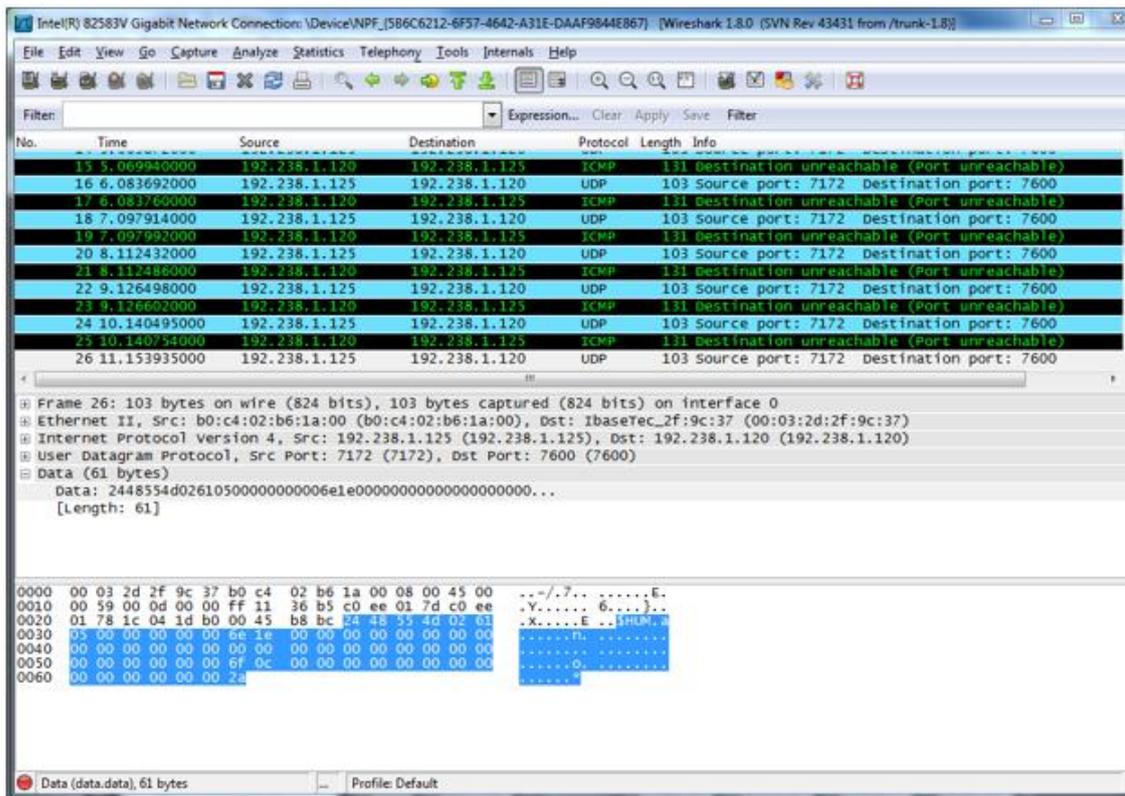


Fig. 5: Simulation result on Wireshark

Once the connection is established between the client and server through a LAN port the simulated data is sent serially to the microcontroller for effective conversion to ethernet frames. The frame analysis is done using Wireshark. The data sent serially through the sonar is now converted to UDP packets(datagrams) as shown in Figure 5.

4. Visual Cognizance

To get the visual perception of the targets in neighborhood of the ship or submarine, the data from the ethernet packet is extracted and displayed on the PPI (Plan Position Indicator). A sonar can detect targets for a maximum radius of 128 km.

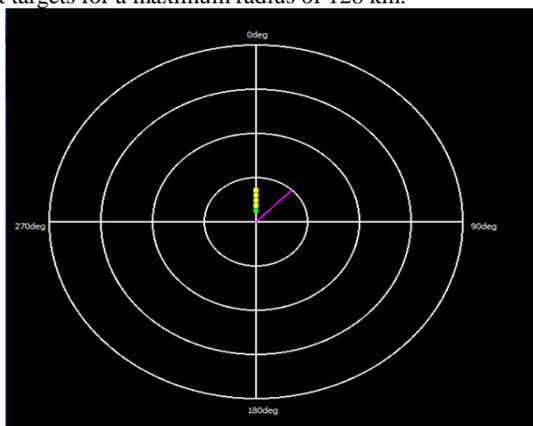


Fig. 6: PPI

As shown in Figure 6. the magenta coloured line indicates the passive target sensed by the sonar, the green dot indicates the initial position of the active target and the subsequent trail of yellow dots indicate the movement of the active target in the given sweep. The position of the target is indicated with respect to the TRUE-NORTH.

5. Classification of Targets into Rocks and Mines

At present the task of keeping track of the targets is done manually. Automating the process of tracking and classifying the targets would make the defense systems more accurate and less time consuming [1]. The data from the SONAR can now be classified as malignant (active targets-mines) or benign (passive targets-rocks) targets. The classification is achieved by the use of artificial neural networks. [3]The data required for training the neural network was taken from UCI repository which maintains the international machine learning database. The dataset consists of two hundred eight samples. Each sample has sixty features and one label.

A. Training the Neural Network Using Linear Regression

For our application, we initially proposed a feed-forward neural network with one hidden layer (Linear Regression model). The hidden layer consists of sixty neurons. This model was trained and tested using a supervised learning algorithm. Coding the neural net was done with the help of TensorFlow library in python. The neural net is trained on a portion of data after shuffling the dataset to prevent overfitting.

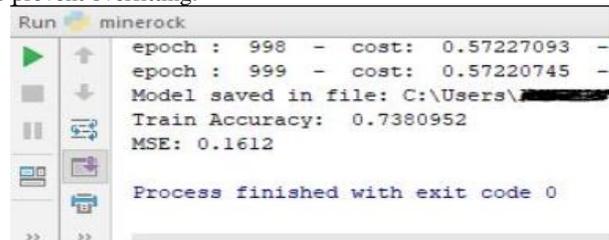


Fig. 7: Training accuracy of linear regression model

```

Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 0 Predicted Values: [1] Accuracy: 0.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [0] Accuracy: 0.0%
Original Class: 0 Predicted Values: [1] Accuracy: 0.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Test Accuracy: 0.76363635

```

Fig. 8: Test accuracy of linear regression model

Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.

B. Training the Neural Network Using Support Vector Machine

The training and testing accuracy of the linear regression model for this application provides very low accuracy varying between 70-80 percent. This model if used in real life situations would lead to wrong classification of the targets there by misusing the resources available and also causing damage in the neighbourhood.

For better classification accuracy, a nonlinear model of classification is preferred. The nonlinear classifier used for our application is SVM (Support Vector Machine). It is a supervised learning model that is used for classification and regression analysis. In Python, the SVM module is available under the Scikit-learn package. On trying various kernels, the polynomial kernel with degree 5 provided better accuracy for the current application. [2][4]

We observed that the testing accuracy has improved when compared to the linear regression model but is still not very efficient. The chances of wrong classification are still greater than 10 percent.

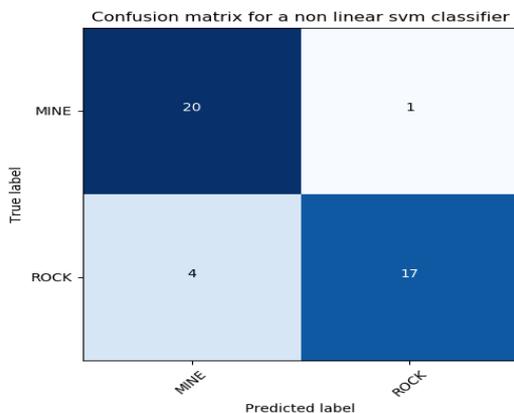


Fig. 9: Confusion matrix for SVM with 5th degree polynomial kernel [5]

The confusion matrix shown in Figure 9. Helps visualize the data graphically. From the figure 9.it can be understood that twenty samples are correctly classified as mines, one sample is incorrectly classified as a rock, four samples are incorrectly classified as mines and the remaining seventeen samples are correctly classified as rocks. This matrix gives a clear perception of the accuracy with which the model works (in this case the SVM).

C. Training the Neural Network Using Multilayer Perceptron

To improve the overall test accuracy, we implemented a multilayer perceptron model. We used a feed-forward neural network with four hidden layers to realize our MLP algorithm. The hidden layers consist of sixty neurons each. MLP utilizes supervised learning technique.

```

epoch : 993 - cost: 0.011706206 -
epoch : 994 - cost: 0.011686009 -
epoch : 995 - cost: 0.011670673 -
epoch : 996 - cost: 0.011647675 -
epoch : 997 - cost: 0.011638807 -
epoch : 998 - cost: 0.011604025 -
epoch : 999 - cost: 0.011580233 -
Model saved in file: C:\Users\...
Train Accuracy: 0.85714287

```

Fig. 10: Training accuracy of the MLP

```

Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Original Class: 0 Predicted Values: [0] Accuracy: 100.0%
Original Class: 1 Predicted Values: [1] Accuracy: 100.0%
Test Accuracy: 1.0
Process finished with exit code 0

```

Fig. 11: Testing accuracy of the MLP

6. Conclusion

It can be observed that the conversion of serial data to Ethernet greatly increases the speed of transmission of sonar data to the Fire Control system (FCS) of the submarine or the ship. Training the FCS using MLP algorithm ensures close to 100 percent accuracy in classifying the targets, thereby providing a boost to the automation in defense systems. The machine learning code is written in python and has been implemented on PyCharm IDE.

References

- [1] R. P. Lippmann. "An introduction to computing with neural nets," IEEE ASSP Magazine, pp. 4-22, April 1987.
- [2] Suresh S. Salankar and Balasaheb M. Patre "SVM Based Model as an Optimal Classifier for the Classification of Sonar Signals", World Academy of Science, Engineering and Technology International Journal of Electronics and Communication Engineering Vol:1, No:2, 2007
- [3] D. Li, K. Wong, Y. Hu, and A. Sayeed, "Detection, classification, and tracking of targets," IEEE Signal Proc. Mag., pp. 17-29, Mar. 2002.
- [4] O.Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee. Choosing Multiple Parameters for Support Vector Machines. Machine Learning, 46(1):131 - 159, 2002.
- [5] Nadav David Marom, Nadav David Marom, Armin Shmilovici, "Using the Confusion Matrix for Improving Ensemble Classifiers", 2010 IEEE 26-th Convention of Electrical and Electronics Engineers in Israel
- [6] Dua, D. and KarraTaniskidou, E. (2017). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.