

Ship Motion Prediction for Launch and Recovery of Air Vehicles

Ameer Khan and A/Prof Cees Bil
RMIT University,
School of Aerospace, Manufacturing
and Mechanical Engineering
GPO Box 2476V,
Melbourne,
Victoria, 3001 Australia
Email:ameerkhan@optusnet.com.au

Kaye E Marion
RMIT University,
School of Mathematical and Geospatial
Sciences
GPO Box 2476V,
Melbourne,
Victoria, 3001 Australia

Abstract - Due to the random nature of the ship's motion in an open water environment, the deployment and the landing of air vehicles from a ship can often be difficult and even dangerous. The ability to reliably predict the motion will allow improvements in safety on board ships and facilitate more accurate deployment of vehicles off ships. This paper presents an investigation into the application of artificial neural network methods trained using singular value decomposition and genetic algorithms for the prediction of ship motion. It is shown that the artificial neural network produces excellent predictions and is able to predict the ship motion satisfactorily for up to 7 seconds.

I. INTRODUCTION

An algorithm capable of predicting the motion of a ship is required for the successful deployment of air vehicles that are currently used on ships that operate in open sea environments. The predicted motion and attitude of the ship will be transmitted to the air vehicle to ensure that it is successfully deployed and subsequently recovered. Future ship motion will allow the correct flight conditions to be calculated thereby allowing successful and safe deployment of the air vehicle.

The motion of a ship in an open water environment is the result of complex hydrodynamic forces between the ship, the water and unknown random processes. This leads to the necessity to use statistical prediction methods for the prediction of this motion rather than a deterministic analysis, which would lead to a ship specific model that involves highly complex calculations and many assumptions or idealizations [1].

Past attempts at ship motion prediction [2-5] have shown that traditional statistical prediction techniques such as the autoregressive moving average models and Kalman filters are unable to maintain a high degree of accuracy when the prediction interval is increased above 3-4 seconds when predicting ship motion in high sea states of 5 and above. The traditional statistical techniques used for time series prediction have difficulty dealing with noisy data, do not have much parallelism and fail to adapt to circumstances.

This paper explores the use of artificial neural networks which is a form of artificial intelligence to develop an algorithm that is capable of predicting ship

motions. Artificial Neural Networks, in contrast to traditional statistical techniques, promise to produce predictions with high accuracy as well as high efficiency due to their ability to learn and adapt according to the conditions present.

II. NECESSITY OF RESEARCH

The ability to predict the ship motion reliably in any sea state will enable better control of air vehicles operated off ship platforms. For example, the landing and take off of helicopters and aircraft whether manned or unmanned from ship decks in rough sea conditions can be difficult and at times dangerous.

If the motion of a ship can be predicted with reasonable error bounds and communicated to the aircraft or helicopter, touchdown dispersion can be improved on landing and a smoother aircraft trajectory can be achieved on take off. Prediction of ship motion is also important for the deployment of missiles and remote piloted vehicle from ship platforms such as those shown in Fig. 1 and Fig. 2 for the correct trajectory calculation [5]. In some cases there is a launch "lock-out" condition where the missile or remote piloted vehicle cannot be launched safely if the ship's roll angle exceeds a predefined operational limit.

There is a specific need for an algorithm that is

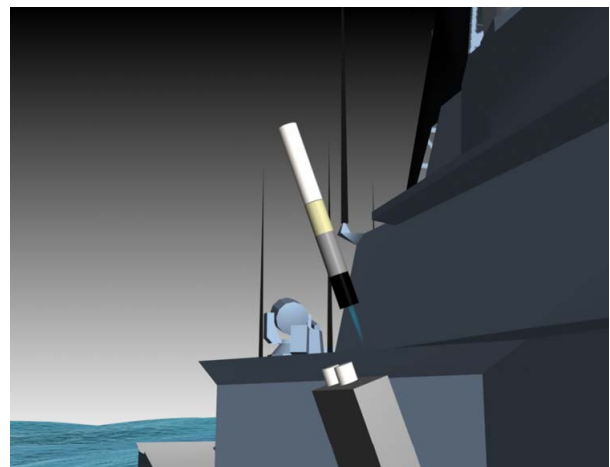


Fig. 1. Deployment of a remotely piloted vehicle



Fig. 2. Harpoon surface to surface missile fired from the USS New Jersey

capable of predicting ship motions beyond 7 seconds. The algorithm developed in this investigation is vital for determining if the air vehicle is in the launch "lock-out" condition as there is a delay between the activation of air vehicle and when they are actually launched. When it is predicted that the angles exceed the launch "lock-out" value the algorithm will deny the activation of air vehicle until it is not in "lock-out".

It is important that the predicted angles are of a high accuracy as the batteries for the system are "one shot" batteries, which means that the process of deployment once activated cannot be reversed.

III. ARTIFICIAL NEURAL NETWORKS

Artificial neural networks (ANN) form a class of systems that are inspired by biological neural networks [6]. A neural network is simply a series of neurons that are interconnected to create a network. They are a class of non-linear systems and there are a wide variety of different approaches that can be used. The use of ANN in time series prediction relate to the application of ANN for the nonlinear system identification. The use of ANN is particularly appealing due to the ability of the ANN to learn and adapt which will be important for this investigation as one of the underlying goals is to create an algorithm that is able to work in all conditions and environments. The ANN architecture that will be used to create the ANN for time series prediction will be the multi-layer feed-forward ANN. This type of architecture has a minimum of two layers consisting of the input layer and the output layer. In this investigation a three layered feed forward neural network consisting of an input layer, a hidden layer and an output layer is used. In a feed-forward ANN the inputs for each layer come from the preceding layer. A single neuron is shown in Fig 3.

It has n inputs including a bias term, which has been

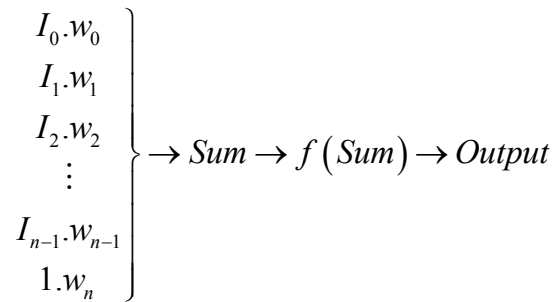


Fig. 3. A representation of a single neuron

set to 1 in this investigation. The inputs are each multiplied by their corresponding weight value, which are summed together and subsequently entered into an activation function. The output of the activation function will correspond to the output of the neuron. Mathematically, the output of a neuron is given as:

$$out = f(net) = f\left(\sum_{i=0}^{n-1} x_i w_i + w_n\right) \quad (1.1)$$

Generally the neuron's operation is not effected significantly by the activation function ($f(net)$) but the training speed is effected somewhat [7]. The activation function is usually a non-linear function that will determine the output of the neuron. Its domain is generally all real numbers. The range of the output for an activation function is usually limited between 0 to 1 and sometimes -1 to 1. The majority of activation functions use a sigmoid (S-shaped) function. In this investigation the activation function shown below was primarily used:

$$f(net) = \tanh(net) \quad (1.2)$$

The use of ANN for time series prediction has a number of distinct advantages. Firstly, any amount of information pertinent to the prediction can be incorporated into the ANN. There is also no need to choose any particular model for the ANN. A validation process is included to ensure that the ANN is working correctly. It is to ensure that the ANN has not over-fitted the data. If the architecture of the ANN is poorly designed, the ANN may be able to learn irrelevant details specific to the training set which will lead to an ANN that is only relevant to the training set. Conversely, the ANN may have a deficient architecture where the ANN is not able to learn the subtleties required for accurate outputs. The validation process should reveal these problems.

To validate the ANN a simple procedure is used. The test set is divided into two. One part of the test set is designated for training purposes only while the second part is designated as the validation set. The ANN is only trained using the training set and no data from the validation set is used while training. Once trained, the validation set is inputted into the ANN and the resulting

predictions based on the validation set are used to measure the effectiveness of the ANN. The validation process is discussed in detail in section III-B.

The basic model for time series prediction is shown in Fig. 4. Any neural network that is capable of accepting real valued vectors as input and producing real valued outputs may be used for time series prediction. In the diagram above it can be seen that there are seven points. Lag 0 represents the current sample while the past six values are represented by lags 1-6. In this investigation lags of up to 60 are used as inputs into the ANN. The output of the ANN is the prediction. It can be noticed that there is only one output shown. For every lead prediction interval it is advisable to use a single ANN. If multiple predictions are required then for every prediction interval a separate ANN is used. The basis for the presumption is that the weights for an optimal prediction will vary according to the prediction interval desired. By having the ANN create multiple predictions, the overall optimal prediction cannot be made. By having separate ANN create separate predictions, the optimal weight configuration can be obtained for each prediction and therefore, higher accuracy can be expected.

A. Training the Network

The training of the network can be viewed as a minimization process where the weights in the ANN are systematically adjusted in a manner that reduces the error between the output of the ANN and the desired output. Therefore the process of training the neural network becomes an optimization problem where the performance of the neural network will be dependent upon the quality of the solution found after the training process has been completed. Therefore the algorithm used to determine the minimum must ensure that global minimum is achieved and has not merely discovered the local minimum. The back-propagation algorithm and the conjugate gradient methods are two techniques which are widely used for training ANN and are very capable of finding the local minimums but there is no guarantee that the global minimum will be found using these techniques. As the global minimum is desired a genetic algorithm has been chosen for this investigation.

The genetic algorithm (GA) is a part of a rapidly growing area of artificial intelligence called evolutionary computing. The term 'evolutionary computing' is based on Darwin's theory of evolution, which states that problems are solved by an evolutionary process resulting in a best solution. It is basically survival of the fittest where the 'fittest' (best) 'survivor' (solution) evolves to create the next population [8]. Solution to a problem solved by the GA uses an evolutionary process based on the principles of genetics and natural selection [9]. A flow chart of the process is given in Fig. 5.

The algorithm begins with a population of solutions. Solutions from one population are taken and used to form a new generation of solutions or the next population of solutions. The expectation is that the new population will be better than the old one. Solutions or individuals are

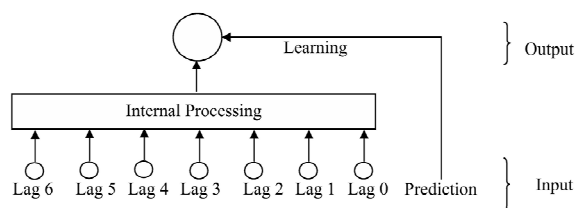


Fig. 4. Basic model for time series prediction using ANN

then selected to form new solutions or 'offspring' according to their fitness. The fitness is a positive value that is used to reflect the degree of 'goodness' of the solution and is directly related to the objective value [10] which is minimization of the mean square error of the difference between the output of the neural network and the target value in this investigation. As mentioned previously, the more suitable they are, the more chances they have to reproduce. This is repeated until a predefined stop criterion is satisfied.

There are a number of advantages in using the genetic algorithm. Firstly, it does not require any derivative information as required by the back-propagation method. It simultaneously searches from a wide sampling of the cost surface which is helpful for finding the general location of the global minimum quickly.

The GA can be implemented on parallel computers which will allow the solution to be found more rapidly than if it were implemented on a single processor which is an important consideration for real time prediction as required for this project. Also, as it provides a list of optimum variables and not just a single solution, if the global minimum is difficult to locate a good alternative

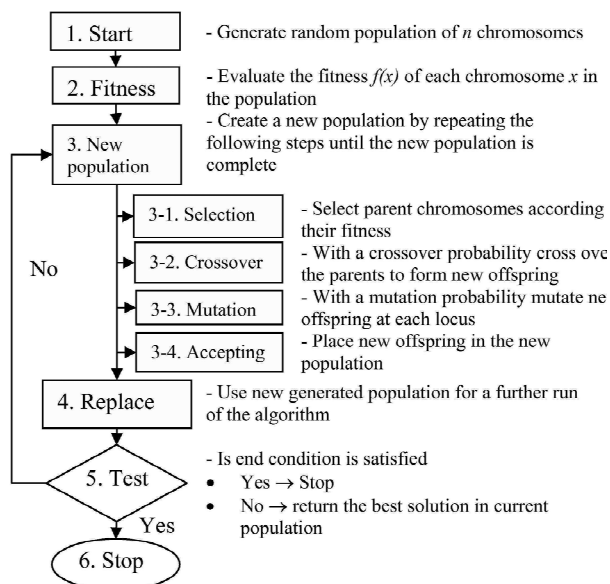


Fig. 5. Flow Chart of the GA process

solution can be returned.

The aim of this investigation is to develop a methodology to predict ship motion in real time. Singular value decomposition (SVD) is a linear regression technique that can quickly obtain an approximate set of optimum weights which is far superior to randomly generating the weights. The values returned from running the SVD can be used as the initial starting point for a selection of the population for the GA algorithm discussed previously and they can sometimes be of such a high standard that this method can be used alone. A detailed description of the SVD technique is beyond the scope of this paper but essentially the matrix X which satisfies the function:

$$A.X = B \quad (1.3)$$

when A and B are known can be calculated efficiently using SVD. When applying it to the ANN process the weights between the input layer and the hidden layer are initially randomly generated. The training samples are then inserted into the ANN and the hidden layer activation functions are calculated creating a matrix equivalent to A . Also, the values for the inverse transfer function of the output are also calculated creating a matrix equivalent to B . Applying SVD and solving equation (1.3), the approximate optimal weights X are found.

B. Validation of ANN Model

To ensure that the weights in the ANN have been correctly set and that the output of the ANN is sufficiently reliable, a validation process is applied after training has been completed. The set of known inputs with their desired output needs to be divided into two distinct sets. The first set is the training set and is used throughout the training period to adjust the weights to the appropriate values.

The second set is referred to as the validation set and is used to test the ANN. Once the values of the training set have been determined, the inputs from the validation set are inserted into the ANN and the output of ANN is compared with the target values in the validation set. The validation process is included to ensure that the ANN is working correctly. It is to ensure that the ANN has not overfitted the data.

The architecture of the ANN refers to the number of neurons that are used in the input and hidden layers. If the architecture of the ANN is poorly designed, the ANN may be able to learn irrelevant details specific to the training set which will lead to an ANN that is only relevant to the training set. Conversely, the ANN may have a deficient architecture where the ANN is not able to learn the subtleties required for accurate outputs. The validation process should reveal these problems. The entire ANN process including the validation process is shown in Fig. 6.

It can be clearly seen that the first stage involves inputting the training set into the ANN. The ANN adjusts its weights in the 'learning' process until the error between the target values and the output of the ANN is

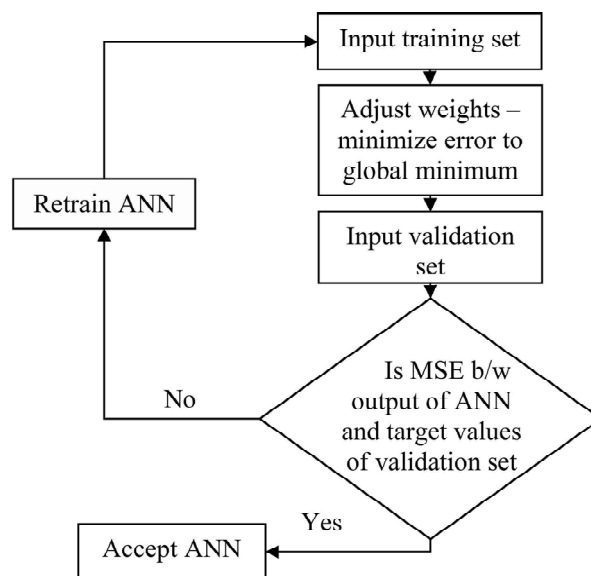


Fig. 6. The artificial neural network process

reduced to a minimum. Next, the validation set is inputted into the ANN. The output of the ANN is compared to the target values of the validation set and the ANN is accepted if the error is of a low enough value or alternatively rejected if the error is too high.

The error in the validation set may be higher than the error found at the end of training but should not be significantly larger or there is a problem with the ANN. Also, the validation set should be independent to the training set to ensure that there is no bias added into the validation process. It is not permissible to use any of the training data in the validation stage, as this will not give a good indication of the ANN's validity.

IV. PREDICTIVE CAPABILITIES OF ANN

In the following sections two investigations are presented. The first investigation presented in section IV-A is intended to determine the basic capabilities of ANN in learning a complex equation that is similar to ship motion. The second investigation presented in section IV-B seeks to assess the levels of accuracy that can be attained when using the ANN to predict a ship's roll motion up to 7 seconds.

A. Complex Equation Prediction Using ANN

Neural networks are believed to be able to learn any function or pattern if given sufficient architecture, enough training data and an adequate amount of training is performed. The nature of the neural network makes it difficult to understand exactly how the neural network is capable of learning based upon a pure mathematical premise. In fact there does not seem to be any strict mathematical verification of the neural networks

capabilities.

The neural network is inherently a function itself which is used to approximate another function which is the problem being solved. One can now state that the neural network is effective in learning a function or pattern if it is able to approximate the function or pattern to an arbitrary accuracy. By the term "arbitrary accuracy" it is meant that the function can be approximated so that errors are within acceptable levels. This does not imply that the neural network is expected to produce results that are exact but it does mean that they produce results that are useful for their intended purpose.

For example, in this investigation it is not expected that an algorithm will be developed that predicts ship motion exactly, however it is expected that an algorithm can be developed that is capable of predicting ship motion to an accuracy where the predictions will be of practical importance.

Three of the proven capabilities of a neural network are stated in Masters [7] and are listed below.

1. If the function consists of a finite collection of points a three layer network is capable of learning it.
2. If the function is continuous and defined on a compact domain, that is the inputs have definite bounds rather than having no limits on what they can be, then a three layer network is capable of learning it.
3. Many functions that do not meet the above criteria can also be learned by a three-layer network. Even discontinuities can theoretically be tolerated under all conditions that are expected in real life.

It is obvious that the problem of ship motion lies within the bounds of the cases outlined above and therefore should theoretically be capable of being modelled using a three layered neural network.

The purpose of this section is to confirm that the neural networks are capable of learning any function and to understand some of the likely architectures that will be required for the prediction of ship motion.

This investigation therefore seeks to confirm the stipulations made in the literature regarding neural network. The ANN was applied to data generated from the following equation to assess the capabilities of the ANN.

$$f(t) = 0.1\sin(0.25t) + 0.35\sin(0.4t) + \dots \\ \dots + 0.8\sin(0.8t) + 0.6\sin(1.1t) + \dots \\ \dots + 0.2\sin(0.1t) \quad (1.4)$$

A test set was generated from equation (1.4) with $t = 0, \dots, 1500$ seconds at a frequency of 15Hz. A three layered feed forward ANN was trained using the SVD method alone to predict 7 seconds in advance. As mentioned earlier, an important requirement when applying ANN is to ensure that there is an adequate architecture to represent the equation satisfactorily. The number of neurons in the input layer (NNIL) was varied from 40 to 70 with increments of 10. For each NNIL there was a varying number of neurons in the hidden layer. A graphical representation of the results is shown in Fig. 7 and were generated by applying the trained

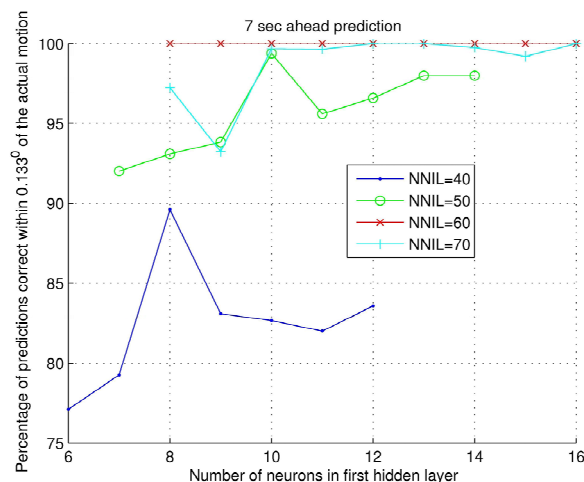


Fig. 7. Percentage of predictions correct within 0.133° of the actual value

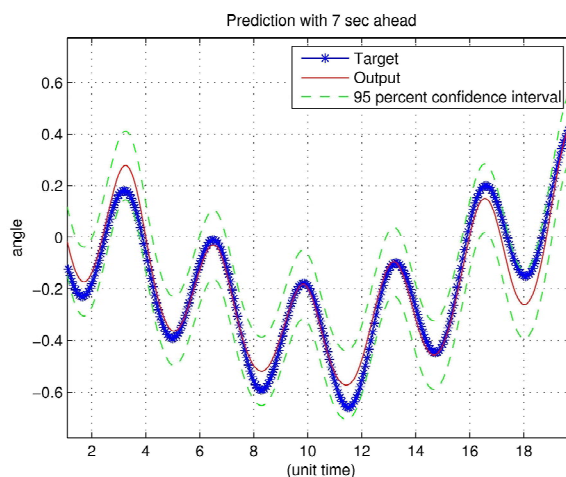


Fig. 8. Sample of the prediction generated using the ANN algorithm with 60 neurons in the input layer and 15 neurons in the hidden layer

ANN to the validation set which was the last third of the test set. Fig. 8 shows a sample prediction created with the trained ANN.

The results seen in Fig. 7 clearly show that the architecture of the neural network has a profound effect on the overall performance of the ANN. Fig. 7 shows that when there were only 40 neurons in the input layer the architecture was inadequate and the ANN was not able to predict future values of the validation set. It also shows that having too many neurons in the input layer leads to a degradation of the ANN's performance possibly due to overfitting where the ANN learnt insignificant aspects of the training set which were irrelevant to the general population.

A general trend that can be seen in Fig. 7 is that there is an optimum number of neurons in the hidden layer required to produce the best results. The trends exhibited in Fig. 7 are in conformity with the previous stipulation that the given architecture of the ANN is an important factor in the ultimate success of the neural network. Another outcome of this initial investigation is that a ANN trained using SVD methods can be very accurate and that using SVD methods to find an initial estimate for the ANN can be of great benefit.

B. Application of ANN to Ship Motion Prediction

The algorithm developed were subsequently applied to measured ship roll angle data taken from a cruiser size vessel operating in sea states 5-6. The term sea state is a description of the properties of sea surface waves at a given time and place [11]. The greater the sea state the rougher the conditions. There was 666 seconds of roll angle data available sampled at 15Hz. The training data was set to two thirds of the data sets and the validation set was designated as the final third of the data sets. All results shown are the predictions made using the validation set only. A graphical representation of the results are shown in Fig. 9 while an example prediction is shown in Fig.10.

In this investigation NNIL was varied from 40 to 60 neurons and the number of neuron in the hidden layer were also varied. The SVD technique was first used to generate 10 approximately optimum sets of weights which were then inserted into the GA along with other randomly generated sets of weights creating the initial population. One hundred generations were completed with the best individual solution chosen.

The results show that the architecture of the neural network is important for high accuracy. Fig. 10 clearly shows that the ANN learnt to predict the ship motion and

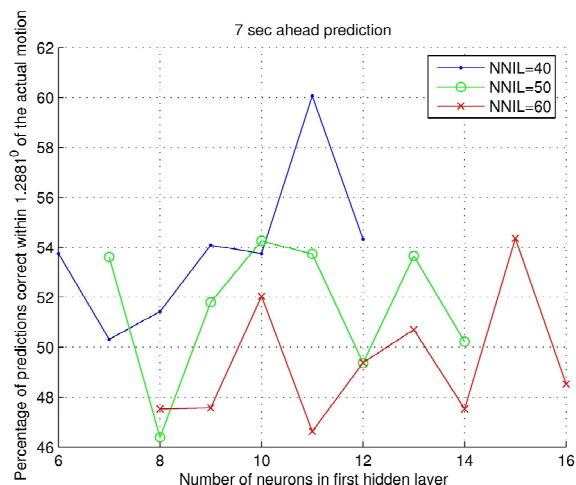


Fig. 9. Percentage of predictions correct within 1.288° of the actual value

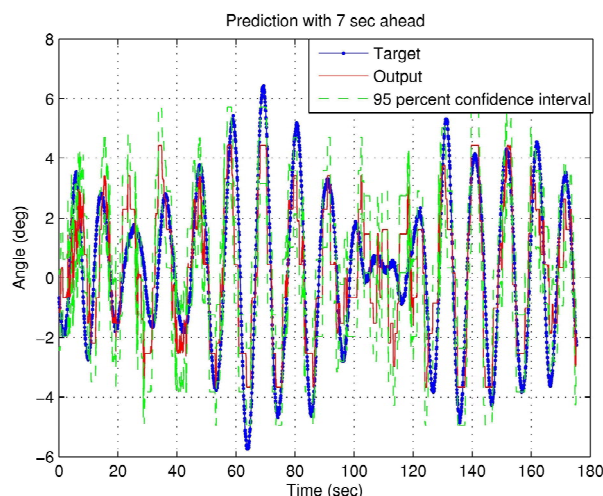


Fig. 10. Sample of the prediction generated using the ANN algorithm with 60 neurons in the input layer and 15 neurons in the hidden layer

remained valid for up to 160 seconds. It can however be seen that between 100 seconds and 130 seconds of the prediction shown in Fig. 10 the ANN performed poorly. In this region the amplitude of the motion is small and as the aim of the investigation is to predict the large amplitude motion where the 'lock-out' condition is likely to exist, this is not a major concern.

This investigation clearly demonstrates that artificial neural networks can be used to predict ship motion effectively for large amplitude motion and is therefore suitable for prediction of the 'lock-out' conditions.

V. CONCLUSION

In this paper an artificial neural network based method utilizing a combination of the singular value decomposition and genetic algorithm for the prediction of the ship motion was presented. It was shown that the singular value decomposition technique alone was able to learn a complex equation and effectively predict future values. It was shown that the three layered artificial neural networks were capable of learning the ship motion and producing highly accurate predictions for intervals up to 7 seconds.

VI. ACKNOWLEDGMENTS

The author would like to thank BAE Systems for providing funding and support for the project.

REFERENCES

- [1] W. G. Price and R. E. D. Bishop, *Probabilistic theory of ship dynamics*. Salisbury: Chapman and Hall Ltd, 1974.
- [2] I. Yumori, "Real time prediction of ship response to ocean waves using time series analysis," *OCEANS*, vol. 13, pp. 1082 -1089, 1981.
- [3] N. B. Cortes, "Predicting ahead on ship motions using Kalman filter implementation," in *Department of Aerospace Engineering*. Melbourne: RMIT University, 1999, pp. 68.
- [4] M. R. Crump, "The Dynamic and Control of Catapult Launching Unmanned Air Vehicles from Moving Platforms," in *Department of Aerospace Engineering*. Melbourne: RMIT University, 2002, pp. 264.
- [5] M. M. Sidar and B. F. Doolin, "On the feasibility of real-time prediction of aircraft carrier motion at sea," *IEEE Transactions on Automatic Control*, vol. 28, pp. 350-356, 1983.
- [6] J. A. K. Suykens, J. P. L. Vandewalle, and B. D. Moor, *Artificial neural networks for modeling and control of non-linear systems*. Dordrecht: Kluwer Academic Publishers, 1996.
- [7] T. Masters, *Practical neural network recipes in C++*. Boston: Academic Press, 1993.
- [8] M. Obitko, "Genetic algorithms," vol. 2004. Prague, 1998.
- [9] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*: John Wiley and Sons, 1998.
- [10] K. F. Man, K. S. Tang, and S. Kwong, *Genetic Algorithms*. London: Springer Verlag, 1999.
- [11] T. Keefer, "Glossary of Meteorology," vol. 2005. Boston: Allen Press., 2005.