

An artificial neural network model for optimization of finished goods inventory**Sanjoy K. Paul^{a*} and Abdullahil Azeem^a**^a*Department of Industrial and Production Engineering, Bangladesh University of Engineering and Technology, Dhaka - 1000, Bangladesh***ARTICLE INFO***Article history:*

Received 1 November 2010
 Received in revised form
 17 January 2011
 Accepted 17 January 2011
 Available online
 17 January 2011

Keywords:

Finished goods inventory
 Artificial neural network
 Optimization
 Inventory model
 Lot sizing

ABSTRACT

In this paper, an artificial neural network (ANN) model is developed to determine the optimum level of finished goods inventory as a function of product demand, setup, holding, and material costs. The model selects a feed-forward back-propagation ANN with four inputs, ten hidden neurons and one output as the optimum network. The model is tested with a manufacturing industry data and the results indicate that the model can be used to forecast finished goods inventory level in response to the model parameters. Overall, the model can be applied for optimization of finished goods inventory for any manufacturing enterprise in a competitive business environment.

© 2011 Growing Science Ltd. All rights reserved

1. Introduction

In the global competitive business environment, manufacturers must maintain optimum quantity of finished goods inventory to reduce cost and to maximize the efficiency of supply chain. Companies aim to supply the required amount of finished goods in right place and at right time with right cost. In this regard, finding optimum amount of finished goods has received extensive attention among the researchers. Holding raw materials, work-in-process and finished goods are involved with various forms of inventory. Finished goods inventory is the amount of manufactured products on hand that awaits sale to customer. In this finished goods inventory model, demand is known and can be changed; products are produced and stored in a lot and set-up, inventory holding and material costs are considered. Successful inventory control involves a tradeoff between the costs of inventory and the benefits of inventory. Finished goods are some of the most important forms of inventories, which involve a substantial amount of cost. The importance of determining the appropriate level of finished goods inventory helps for proper planning and control of factory operations and optimization of the overall process to minimize production cost and time to produce products of desired quality.

Most of the companies apply EOQ model to determine the maximum level of inventory or ordering lot size. The EOQ model is very straightforward and assumes several unrealistic assumptions such as constant demand and purchasing price, deterministic setup, no shortage of finished goods and

* Corresponding author. Tel: +8801712983250
 E-mail: sanjoy@ipe.buet.ac.bd (S. K. Paul)

instantaneous receipt of finished goods. There are, however, several interrelated variables, which are not fixed in nature such as demand, price, etc. Such variables vary within some ranges in time horizon and with regards to market and socio-economic aspects. Simple EOQ model cannot incorporate such variables and, therefore, the usefulness of EOQ model for lot sizing is not satisfactory for the organizations.

Our motivation came from a real-world case study where a manufacturing industry was visited in 2009, which was suffering from some inventory related problems. Historical data are collected which includes the information of one year, 2009, of the concerned industry. After researching the company's database and talking with supply chain experts, it was observed that the company was using EOQ model. The failure of EOQ model was identified to determine the optimum quantity of finished goods inventory and, hence, to decrease the performance of the supply chain. By analyzing the industry case variables, we determined demand, setup, holding and material cost which are fluctuating in nature and affect finished goods inventory. The relationship between input and output is so complex that it is very difficult to handle without a relationship model.

ANN technique is a promising tool for optimizing inventory in manufacturing environment. ANN behaves as model free estimators i.e. they can capture and model complex input-output relationships even without the help of a mathematical model (Ntuen, 1991). ANN has been widely used for modeling complex process due to their learning and generalization capabilities, accommodation of non-linear variables, adaptability to change environments and resistance the missing data. Adaptability in response for changing environment means ANN can be applied for solving inventory problems where input parameters such as the nature of business, types of product, customer preferences, economic circumstances, and brand preferences are changing. The ANNs have been successfully used in many applications such as network reliability (Chat et al., 2002), production scheduling (Yildirim et al., 2006) and in job shop problem (Fonseca & Navarrese, 2002). Lotfi and Choueiki (2000) applied neural network model in MRP problem of lot sizing. Partovi and Anandarajan (2002) conducted ABC classification of stock keeping units for a manufacturing company by ANN. As a learning method, authors use back propagation and genetic algorithm. Koh et al. (2002) developed a model for optimal ordering and recovery policy. Maiti and Maiti (2006) used fuzzy logic to develop an inventory model. Wee and Chung (2009) developed an integrated production inventory-deteriorating model to optimize replenishment policy. Wee et al. (2007) considered imperfect quality and shortage backorder to develop optimal inventory model. Chiu (2003) considered random defective rate, the rework process, and backlogging to determine optimal lot size in inventory control. Chang (2004) applied fuzzy set theory to the EOQ model. Lin et al. (2009) focused on finding an optimal work-in-process inventory value of wafer fabrication processes by developing an algorithm integrating an ANN and the sequential quadratic programming (SQP) method. With this approach, it offered an effective and systematic way to identify an optimal WIP level. Au et al. (2008) developed optimized neural networks structure for the forecasting of apparel sales. They proposed an evolutionary computation approach in searching for the ideal network structure for a forecasting system and found that the proposed algorithms are useful for fashion retail forecasting, and the method performed better than the traditional models for products with features of low demand uncertainty and weak seasonal trends. Gutierrez et al. (2008) applied neural network (NN) modeling in forecasting lumpy demand. They compared the performance of neural network forecasts to those using three traditional time series methods (single exponential smoothing, Croston's method, and the Syntetos-Boylan approximation) and found that neural network models generally perform better than the traditional methods, using three different performance measures.

Finished goods inventory is an important form of inventory which involves a substantial cost. Level of finished goods inventory depends on some factors i.e. product demand, setup cost, material holding cost, material cost, etc. Finished goods inventory changes as the other influencing factors change. In this model, four input variables are considered, so the relationship between input and

output is complex in nature and it is very difficult to develop the mathematical model. As ANN can capture and model complex input-output relationships even without the help of a mathematical model, an ANN technique is used to determine the optimal level of finished goods inventory. A brief review of the literature to determine the finished goods inventory reveals that it is promising to use ANN. However, no particular research has been proposed to optimize finished goods inventory level considering the issues of products demand, setup cost, holding and material cost as input. To fill this gap, new tools or methods are important which can accommodate the fluctuating nature of the identified variables. By realizing the importance and the research gap, this paper deals with developing an ANN model which can be used to optimize finished goods inventory in a fuzzy environment. The input parameters of the ANN model are products demand, setup cost, holding cost, and material cost. Real data is used from a manufacturing industry to establish and to validate the proposed model. Further details of the model are described in next sections.

2. Proposed ANN model for optimal finished goods inventory

The structure and basic elements for designing artificial neural networks is shown in Fig. 1. For the present study, an ANN model is developed to determine the optimum quantity of finished goods inventory in a fuzzy environment. The capability of the ANN model is to generalize unseen data dependents on several factors. These factors are appropriate selection of input-output parameters, the distribution of the input-output dataset, and the format of the presentation of the dataset to the neural network. Selected input parameters are the significant variables that affect the finished goods inventory of a manufacturing organization. The input parameters include product demand, setup cost, holding cost and materials cost. The output parameter of the model is the optimum amount of finished goods inventory. Details of the input/output parameters of the proposed ANN model are illustrated in Fig. 2.

Based on the relationship between finished goods inventory and input parameters, an ANN model is developed in MATLAB 7.6 software. In the proposed model, four basic steps are adopted which are the collection of input-output dataset; pre-processing of the input-output dataset; designing and training of the neural network; and finally performance evaluation of the designed neural network.

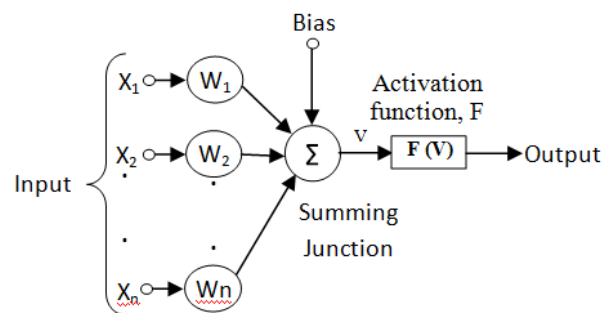


Fig. 1. A single ANN neuron with its elements

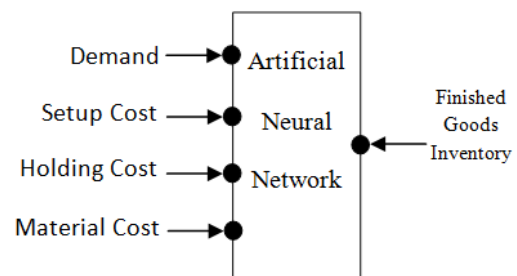


Fig. 2. Schematic diagram of ANN

Real data from a manufacturing company is used for establishing and validating the proposed ANN model. In this ANN model, as an input we use four parameters, which influence inventory decisions. The output dataset of the model consists of the amount of finished goods inventory. Before training the neural network, one significant task is to process the experimental data into input-output patterns. Training and testing pattern vectors are formed before input-output dataset are fed to network. Each pattern is formed with an input condition vector (P_i) and the corresponding target vector (T_i), which is shown in the matrix. Before training the network, the input-output dataset were normalized within the range of -1 to +1 using the MATLAB command 'premnmx'.

$$P_i = \begin{bmatrix} Demand, D \\ SetupCost, S \\ HoldingCost, H \\ MaterialCost, M \end{bmatrix} \quad T_i = [Finished\ Goods\ Inventory] \quad (1)$$

Table 1

Examples of input-output data set

| Input [D, S, H, M] | Output [Quantity] | Input [D, S, H, M] | Output [Quantity] |
|--------------------|-------------------|--------------------|-------------------|
| [1000, 20, 0.4, 1] | [500] | [1000, 20, 1.2, 5] | [410] |
| [1000, 20, 0.4, 3] | [450] | [1000, 20, 2.0, 1] | [460] |
| [1000, 20, 0.4, 5] | [400] | [1000, 20, 2.0, 3] | [425] |
| [1000, 20, 1.2, 1] | [475] | [1000, 20, 2.0, 5] | [405] |
| [1000, 20, 1.2, 3] | [440] | [1000, 40, 0.4, 1] | [625] |

In total, 81 combinations of input-output dataset pairs are collected from a manufacturing company product. In response to different combinations of input parameters, several pairs of output variables are also obtained. A sample of input-output dataset is shown in Table 1. MATLAB 7.6 is used for training the network architecture. The training performance of the optimal network architecture is shown in Fig. 3. The input-output dataset consisting of 81 patterns is divided randomly into two categories: training dataset consist of 78% of the data and test data set which consists 22% of the data. There are 63 training patterns considered for ANN modeling of finished goods inventory for a manufacturing industry. After the training, the weights are frozen and the model is tested for validation. The network is validated with actual data collected from the industry. For this purpose, the input parameters to the network are sets of values (in this case 18 pairs of dataset which are shown in Table 2) that have not been used for training the network (raw untrained data) but are in the same range as those used for training. This enables to test the network with regards to its capability for interpolation regarding unseen data.

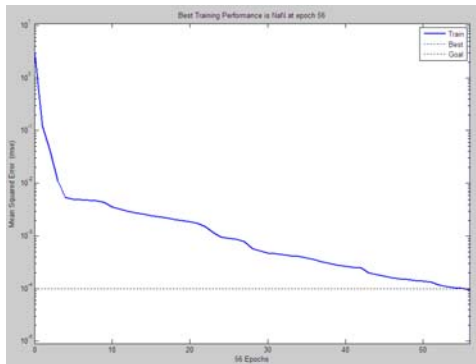


Fig. 3. ANN training performance

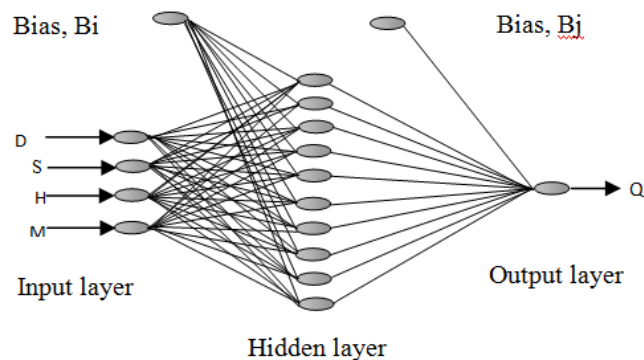


Fig. 4. Proposed ANN architecture

Each neuron in the network acts as a processing element, which performs a weighted sum of all input variables that are fed to the system. Depending on the value of weighted sum of the variables, the neuron gives a signal to the neurons in the adjacent layer through a non-linear transfer function (sigmoid function in this case). The amount of finished goods inventory of training samples is treated as the desired and target output. The algorithm used for the neural network learning is 'the backward-propagation algorithm' with scaled conjugate (SCG) version. The neural network learning is adaptive in nature that means vector pairs from the training model are mapped respectively to reinforce the weights until deviation between the training output and the desired output of each training vector sample converges to a negligible error of 0.0001 in this application.

Table 2
Examples of test data set

| Test Data Set | [D (units/month), S(\$), H (\$), M (\$)] | Test Data Set | [D (units/month), S(\$), H (\$), M (\$)] |
|---------------|---|---------------|---|
| 1 | [1000, 20, 0.4, 5] | 10 | [3000, 40, 2.0, 5] |
| 2 | [1000, 20, 2.0, 3] | 11 | [3000, 60, 0.4, 5] |
| 3 | [1000, 20, 1.2, 3] | 12 | [5000, 20, 1.2, 3] |
| 4 | [1000, 40, 1.2, 3] | 13 | [5000, 20, 0.4, 3] |
| 5 | [1000, 40, 0.4, 5] | 14 | [5000, 20, 2.0, 1] |
| 6 | [1000, 60, 2.0, 5] | 15 | [5000, 40, 0.4, 3] |
| 7 | [1000, 60, 2.0, 3] | 16 | [5000, 40, 2.0, 5] |
| 8 | [3000, 20, 2.0, 3] | 17 | [5000, 60, 0.4, 3] |
| 9 | [3000, 40, 1.2, 1] | 18 | [5000, 60, 2.0, 1] |

The optimum ANN architecture is shown in Fig. 4. The momentum constant and learning rates used in this model are 0.5 and 0.1, respectively. The maximum number of training epochs is set to be 10,000 and the training error goal is 0.0001. After completion of training, the model stores the actual weight values in a separate file. After post processing the network predicted values by using the MATLAB command 'postmnmx', we adopt it for regression analysis.

Table 3
 R^2 and MAPE values between the network predictions and the experimental values

| | Training performance | | | | Testing performance | | | |
|----------------|----------------------|--------|--------|--------|---------------------|--------|--------|--------|
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Hidden layer | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Hidden neurons | 8 | 10 | 12 | 15 | 8 | 10 | 12 | 15 |
| R^2 | 0.9998 | 0.9998 | 0.9998 | 0.9997 | 0.9919 | 0.9848 | 0.9846 | 0.9833 |
| MAPE | 0.0093 | 0.005 | 0.0429 | 0.0119 | 0.0047 | 0.001 | 0.0015 | 0.001 |

The coefficient of determination value (R^2) for both training and testing phases are found to judge the performance of each network. To judge the training and testing performance, in this analysis another index term is used as mean absolute percentage of error (MAPE). The values of R^2 and MAPE for different network predictions and the experimental values are shown in Table 3. Within the table, some examples of the hidden neurons are shown that were considered during designing and developing the ANN model. The summary of the proposed ANN model is shown in Table 4.

It can be seen from the Table 3 and Table 4 that network with 1 hidden layer and 10 hidden neurons, and network with 'tansigmoid' and 'purelin' transfer function (in the hidden and output layer respectively and trained with SCG algorithm) provide the best result. It can also be seen from Table 3 that an increase on the number of neurons from 8 to 15 has no significant improvement on the performance of the network. Therefore, 4-10-1-network architecture is selected as the optimum ANN model. The performance of the ANN model is shown in Fig. 5 for the response variable namely the amount of finished goods inventory. For different test dataset, we compared ANN prediction values and observed values for the output variable depicted in Fig. 5. It is apparent that the proposed model can predict values, which are close to actual observations for the output parameter. The results conclude that the ANN can be easily used to determine the amount of finished goods inventory in a fuzzy environment.

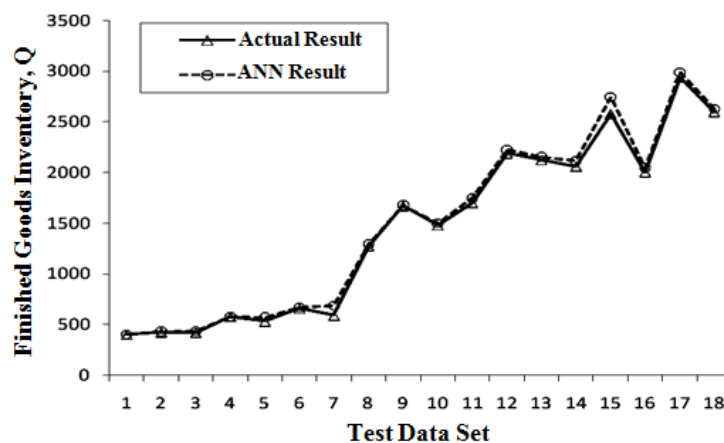


Fig. 5. Comparisons of ANN prediction and actual values

Table 4

Summary of ANN model

| | | |
|-----------------------------|---|--------------------------------------|
| Object model | : | Finished goods inventory |
| Input neuron | : | D, S, H, M |
| Output neuron | : | Q |
| Network structure | | |
| Network type | : | Feed-forward back propagation |
| Transfer function | : | Tansig/ Purelin |
| Training function | : | Trainscg |
| Learning function | : | Learngdm |
| Learning conditions | | |
| Learning scheme | : | Supervised learning |
| Learning rule | : | Gradient descent rule |
| Input neuron | : | Four |
| Output neuron | : | One |
| Sample pattern vector | : | 63(for training), 18 (for testing) |
| Number of hidden layer | : | 1 (one) |
| Neurons in hidden layer | : | 10 |
| Learning rate | : | 0.1 |
| Momentum constant | : | 0.5 |
| Performance goal/Error goal | : | 0.0001 |
| Maximum epochs (cycles) set | : | 10,000 |
| MSE at the end of training | : | 0.005 |

3. Results and discussion

In this study, an ANN with feed-forward back-propagation algorithm was trained and the training epoch (cycles) set for each network is 10,000. The purpose of the training is to minimize the mean squared error (*MSE*). The training performance of the proposed ANN architecture shows that the

network error goal (mean square error is 0.0001) is met at 56 epochs (shown in Fig. 3). Different training algorithms are tested to find the best network. Transfer functions in the hidden and output layer are changed and they are tested during design phase of the network. Finally, 'tansigmoid' function and 'purelin' function were used as the transfer function in the hidden and output layer, respectively. The numbers of neurons in the hidden layer were found by trial and error method and finally 10 hidden neurons were chosen for the suggested network. The proposed network can be represented as 4-10-1, i.e. the proposed ANN model consists of 4 inputs, 10 hidden neurons and 1 hidden layer. To find the optimal network architecture, R^2 and $MAPE$ between the network prediction and experimental values are calculated for every network for both training and testing phases. The coefficient of determination (R^2) represents the precision of data which is significantly close to the fitted line. The value of R^2 varies between 0 and 1. If correlation coefficient, $R=0.9487$ then $R^2=0.90$, which means that 90% of the total variation in network prediction can be explained by the linear relationship between experimental values and network predicted values. The other 10% of the total variation in network prediction remains unexplained.

The R^2 and mean absolute percentage of error $MAPE$ for different network topography are reported in Table 3. From Table 3, it is shown that the value of R^2 does not change by increasing the number of neurons from 8 to 15. The network architecture consisting of 1 hidden layer and 10 hidden neurons, shows the best values of R^2 for both training and testing stages of the network and $MAPE$ is minimum for both training and testing. Therefore, the network consisting of 10 hidden neurons was selected as the optimum one in this research work. The summary of the proposed network architecture has been presented in Table 4. As the input and the output vectors were supplied to the network, it was a supervised learning scheme. The back-propagation learning algorithm with SCG versions was used at the training stage of the network. Gradient decent learning rule is used in this study. The learning rate and momentum constant used here are 0.1 and 0.5, respectively. The co-efficient of determination R^2 obtained corresponding to the output variable is 0.9998 for training. Regarding testing, this value is 0.9848, The $MAPE$ between the experimental and the predicted values are 0.005 and 0.001 for training and testing, respectively.

4. Conclusions

An ANN model has been developed for optimization of finished goods inventory as a function of demand, setup cost, holding and material cost. The model was proved to be successful in terms of agreement with actual values for a manufacturing industry. The back-propagation learning algorithm was used for the development of feed-forward single hidden layer network. Tansigmoid function and purelin function were used as the transfer function in the hidden and output layer, respectively. Training of the network was performed using scaled conjugate feed forward back-propagation algorithm. The multilayer feed forward network consisting of 4 inputs, 10 hidden neurons (tangent sigmoid neurons) and one output was found to be the optimum network for the model developed in this study. A good performance of the neural network was achieved with coefficient of determination (R^2) and mean absolute percentage of error ($MAPE$) value between the model prediction and actual values were 0.9848 and 0.001, respectively. Some other variables such as reliability of production system can be considered to optimize the model. Feed forward learning algorithm and some other functions can also be considered to get the better results. It can be concluded that ANN model performs accurately to determine the optimal amount of finished goods inventory. In conclusion, we stress that the proposed model can be used for optimization of finished goods inventory and selection of the parameters.

References

- Au, K. F., Choi, T.M., & Yu, Y., (2008). Fashion retail forecasting by evolutionary neural networks. *International Journal of Production Economics*, 114, 615-630.
- Chang, H. C., (2004). An application of fuzzy sets theory to the EOQ model with imperfect quality items. *Computers & Operations Research*, 31, 2079-2092.
- Chat, S., Konak, A., & Smith, A. E., (2002). Estimation of all-terminal network reliability using an artificial neural network. *Computers and Operations Research*, 29, 849-868.
- Chiu, Y. P., (2003). Determining the optimal lot size for the finite production model with random defective rate, the rework process, and backlogging. *Engineering Optimization*, 35(4), 427-437.
- Fonseca, D. J., & Navarrese, D., (2002). Artificial neural networks for job shop simulation. *Advanced Engineering Informatics*, 16, 241-246.
- Gutierrez, R. S., Solis, A. O., & Mukhopadhyay, S., (2008). Lumpy demand forecasting using neural networks. *International Journal of Production Economics*, 111, 409-420.
- Koh, S. G., Hwang, H., Sohn, K. I., and Ko, C. S., (2002). An optimal ordering and recovery policy for reusable items. *Computers and Industrial Engineering*, 43, 59-73.
- Lin, Y. H., Shie, J. R., and Tsai, C. H., (2009). Using an artificial neural network prediction model to optimize work-in-process inventory level for wafer fabrication. *Expert Systems with Applications*, 36, 3421-3427.
- Lotfi, K. G., & Choueiki, M. H., (2000). A neural network model for solving the lot sizing Problem. *Omega*, 28, 175-184.
- Maiti, M. K., and Maiti, M., (2006). Fuzzy inventory model with two warehouses under possibility constraints. *Fuzzy Sets and Systems*, 157, 52-73.
- Ntuen, C. A., (1991). A neural network model for a holistic inventory system. *Proceedings of the International Industrial Engineering Conference*, 435-444.
- Partovi, F. Y., & Anandarajan, M., (2002). Classifying inventory using an artificial neural network approach. *Computers and Industrial Engineering*, 41(4), 389-404.
- Wee, H. M., & Chung, C. J., (2009). Optimizing replenishment policy for an integrated production inventory deteriorating model considering green component-value design and remanufacturing. *International Journal of Production Research*, 47(5), 1343 - 1368.
- Wee, H. M., Yu, J., and Chen, M. C., (2007). Optimal inventory model for items with imperfect quality and shortage backordering. *Omega*, 35, 7-11.
- Yildirim, L. B., Cakar, T., Doguc, U., & Meza J. C., (2006). Machine number and the due date determination in flexible manufacturing systems using artificial neural networks. *Computers and Industrial Engineering*, 50, 185-194.