GE-INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH
VOLUME -2, ISSUE -10 (December 2014) IF-3.022 ISSN: (2321-1717)

IDENTIFICATION OF LASER FORMING PROCESS USING RECURRENT AND FOCUSED TIME LAG RECURRENT NEURAL NETWORKS

Sanidhya Naikwad[1], I. A. Palani[2] , S. N. Naikwad[3]

[1, 2] Mechatronics and Instrumentation laboratory, Department of Mechanical Engineering, Indian Institute of Technology, Indore, (India)

[3] Department of Electrical Engineering, College of Engineering and Technology, Babhulgaon, Akola-444 104 (India)

ABSTRACT

Laser forming is a non-contact type recently developed technique that uses a defocused laser beam to form sheet metal by inducing thermal stresses rather than external forces. Mechanism of laser forming is described by many models in the literature but analytical models become difficult and cumbersome, and numerical models are more time consuming to predict the deformed shape due to multiple heating lines and multiple scans. Recently, potential of neural networks has attracted attention of researchers in developing a model to identify the behavior of system. This article presents neural network based models for identification of laser forming process in which it is shown that the recurrent neural network model and focused time lag recurrent neural network model closely follows the desired output of laser forming process i.e. bending angle for testing instances. Prediction error of the estimated model is less than models reported earlier by the researchers. It has been shown that most of the information about the rich nonlinear dynamics of the system has been extracted successfully from the training data set and the proposed model approximates the given system with reasonable accuracy.

KEYWORDS - Laser Forming, Recurrent Neural Network, Focused Time Lag Recurrent Neural Network, System Identification

1. INTRODUCTION

Laser forming**,** as a non-contact method, offers controlled shaping of metallic and nonmetallic components. The process is achieved by inducing thermal stresses into the work-piece by irradiation with a defocused laser beam, which gives rise to controlled elastic-plastic deformation or buckling. Laser forming process offers significant potential value to industry such as aerospace, shipbuilding, microelectronics, etc. However, the process has not gained momentum and is not applied to large scale in industries because of the lack of automation or due to the difficulty in determining the process parameters and scanning patterns to produce any desired shape within a reasonable time. To apply laser forming process in reality, it is required to find the relationships between the deformed shape and scanning paths along with heating conditions. The deformation due to laser scanning depends on various factors such as laser power, scan speed, spot diameter, scan position, number of scans, and many others. Modeling of the laser forming process may help to provide a basis for determining the heating pattern required, therefore making application of laser forming feasible and profitable to industry. The mechanism of laser forming is considerably complex hence performance prediction becomes difficult due to high degree of non-linearity. Research to-date on laser forming has been largely focused, theoretically and experimentally, on the problem of characterization of process parameters on the forming results, and computational simulations of laser forming remain limited only to provide an insight into the process. Modeling of laser forming started with numerical simulation model in 1993. Wenchuan Li and Y. Lawrence Yao [3] developed several numerical models then followed by Hu et al.[8], Zhang and Michaleris [10] and Hu et al. [9] who used finite element method to predict bending angles in laser forming process. But these models needed temperature dependent material properties and require long computational time particularly for the multiple laser scanning. Some models focused on determining the heating lines and heat conditions to achieve a desired (deformed) shape by laser forming. Researcher Vollertsen and Geiger [5] and Zhang and Michaleris [10] used a finite element method and a finite difference method for estimation of the bending angle formed by a laser. There after Vollertsen and Rodle [6] and Kao [11] proposed analytical models to predict bending angle. But as these analytical models are generally established based on over-simplified conditions could not give proper insight of the process. To overcome the problem, artificial intelligence techniques are then used by some researchers. Consequently, due to development of neural networks, it became possible to develop a model which can learn from experimental data.. Thus a system model can be developed by estimating unknown plant parameters using neural networks. Cheng and Lin [4] used neural network in 2000 for the first time to predict bending angle of sheet metal formed by a laser. Recently, in 2012 Kuntal Maji et al. [7] successfully applied neural networks to carry out analysis and synthesis of laser forming process. These papers hint at the scope and utility of neural networks in modeling the laser forming process. Neural networks like recurrent neural network and focused time lag recurrent neural network, being suitable for modeling systems involving time component, are used in this paper to generate a model. This paper describes a systematic approach to estimate optimal neural network model for identification of laser forming process using recurrent and focused time lag recurrent neural network models. Results obtained are stated in section 5 and conclusion is discussed in section 6.

2. NEURAL NETWORK APPROACH

System identification is an integral part of a control system design. It is required to establish a model first so that a controller based on it can be designed and it is also useful for tuning and simulation before applying the controller to the real system. Hence, models of real systems are of fundamental importance in virtually all disciplines. Since the quality of the model is often the bottleneck in the development of the whole system, a strong demand for advanced modeling and identification arises. Accurate system representation requires a perfect study of the system dynamics. In essence, the problem lies in finding a convenient way to model the behavior of the system using certain mathematical tools. The effectiveness of the model can be judged from various standpoints including simplicity, accuracy, computational considerations and physical validity. The development of accurate, yet mathematically tractable models is a challenge before researcher. Due to recent development in the area of neural networks and fuzzy logic, it became possible to model almost all possible aspects of the system behavior. This allowed for development of relatively simple models that are reasonably accurate with respect to practical applications. Neural networks are an effective tool to perform any nonlinear input output mappings. It was the Cybenko [1], who first showed that, under appropriate conditions, they are able to uniformly approximate any continuous nonlinear function to any desired degree of accuracy. This fundamental results that allowed scientists to employ neural network for system identification purpose. One of the primary reasons for employing neural network is to create a machine which learns from experience. They have the capability to learn the complex nonlinear mappings from a set of observations and predict the next outcome as discussed by Dudul [2]. A major goal for any nonlinear system modeling and identification scheme is universalness; that is, the capability of describing total behavior of structurally different systems. A model should represent the behavior of a system as closely as possible. The model quality is typically adjudged in terms of a function of the error between the process output and, model output. This error is utilized to adjust the parameters of the model.

2.1. System Identification

System identification is carried out in four basic steps involves four steps which are;

1) Preparation of data record.

2) Choosing the set of models or model structure.

3) Determining the 'best' model in the set.

4) Model Validation.

Experiment

Neural network is a learning machine which learns from a set of data that describes how the system behaves over its entire range of operation. With this motive, an idea is to vary all possible input(s) and observe their effect on the output(s). The data set comprising of such inputs and outputs is later used for deducing a model of the system. Once a set of data has been acquired, it is required to carry out a visual inspection of data to determine whether it requires additional filtering, or there are outliers to be removed and to check the redundancy. Finally, the input and output sequences should be scaled to zero mean and unity variance.

Selection of Model Structure

A model structure is a set of hopeful models in which one should search for a optimal model. It is not easy to determine the optimal model structure, so instead one should follow a path that is reasonably effortless to follow and that finds a structure, which is sufficiently close to the optimal. The choice of approach depends heavily on the data available for work.

- If the amount of data is small, it is not possible to select a model structure that is "large enough". Thus, in this case a bias error will typically contribute significantly to the average generalization error.

- If the amount of data is large, the effect of regularization and pruning will be insignificant. Then it is only required to consider fully connected networks trained with a regularization term.
- If the amount of data is medium-sized, a suitable compromise between bias and variance error is found. In this case, *Regularization by weight decay* is used to deal with the issue.

Model Estimation

Estimation of model involves two basic tasks. First, a criterion is selected specifying how the weights should be determined from the set of data then an iterative search method is chosen to minimize the criterion (a training algorithm). Generally, a mean square error (MSE) criterion with and without regularization term is chosen. The data set is then divided into training set and test set respectively. A common choice is half-and-half, but the amount of data may motivate other decisions. If the data set is small, it is common to include a larger fraction of the data in the training set. In fact, one can get valuable insight from repeating the procedure for different *split ratios*. The graph of the estimated test error as function of the size of the training data set is called the *learning curve.* Learning curve demonstrates the effectiveness of the learning procedure.

Validation of model

Once a model has been estimated or trained it must be evaluated to check whether it meets the necessary requirements or not. The validation is closely connected to the intended use of the model hence it is considered as the most difficult stage in the system identification procedure because the requirements for acceptance are often somewhat fuzzy. It is seen that the trained neural network model is validated on a set of data that was not used for training the network. The exact nature of validation procedure is of course dependent on the intended use of the model.

Going Backwards in the Procedure

To determine optimal models, it is necessary to go back in the procedure to try out various model structures, and in the worst case even redo the experiment. Accordingly, when the final model has been found, the weights should be rescaled, so that the model can be applied directly to the un-scaled data. The model is then accepted and used for designing a controller of the system.

2.2. Recurrent Neural Network

Recurrent neural networks have been an important focus of research and development around year1990. They are designed to learn sequential or time varying patterns. A recurrent net is a neural network with closed loop feedback connections. Recurrent neural networks have feedback connections from neurons in one layer to neurons in a previous layer. Various modifications in the network have been explored. A typical recurrent network has concepts related to the nodes whose output values feedback as inputs to the network. Hence the next state of a network depends not only on the connection weights and the currently presented input signals but also on the previous states of the network. Recurrent neural network techniques have been largely applied to a variety of problems involving dynamical systems with time sequence of events. Initially, time lagged feed forward networks (TLFNs) are considered for temporal processing. Basically, they could implement static mappings from the present input and its memory traces to the desired output. There is often a need to extend the network capabilities to time-dependant nonlinear mappings. Thus, it means that short-term memory mechanism have to be brought inside the feedforward network topologies or that the network have to be made spatially recurrent, i.e. recurrent connections must be created among some or all PEs. Such spatially recurrent networks are called simply recurrent networks. The complexity of two solutions is very different. TLFNs have locally recurrent connections and they can be made stable by enforcing the stability of the short-term memory mechanisms, while it is more difficult to guarantee stability of recurrent networks. Moreover, TLFNs are easier to train than recurrent neural networks. So they are more practical. Still the complex interconnectivity of the recurrent system usually inhibits one's ability to study the system. The recurrent architecture ranges from fully interconnected to partially connected nets including multilayer feedforward networks with distinct input and output layer. Fully connected recurrent networks do not have distinct input layers of nodes, and each node has input from all other nodes. Feedback to the node itself is possible.

Simple partially recurrent neural networks are used to learn strings of characters. Although some nodes are part of a feedforward structure, other nodes provide the sequential context and receive feedback from other nodes. Like input units, weights from the context units are processed using back propagation. The context units receive time-delayed feedback from the second layer units. Training data consists of inputs and their desired successor outputs. Using this data, a net can be trained to predict the next letter in the string of characters and to validate a string of characters. As weights assigned to the feedback connection are not necessarily adjustable, a standard MLP training algorithm is used to adjust the feedforward connection weights. In addition to feedback connections, which uses a previous-time moment output value for calculating the next one, a buffer of output values for more than one step back in time can be used as an additional input buffer. Such recurrent networks are called *buffered networks*. A recurrent network which keeps of its k-previous states can be represented as unfolded MLP with k lagers of connections. The idea is to duplicate the modes in space in order to achieve time dependence.

2.3. Focused Time Lag Recurrent Neural Network

When there lies a time structure underlying the data collected after rigorous experimentation, dynamic modeling will certainly help to improve the performance. Dynamic neural networks are topologies designed to explicitly include time relationships in the input-output mappings. Time constitutes an indispensable and important component of the learning process. It is through the inclusion of time into operation of neural networks that it is enabled to follow statistical variations in non-stationary processes. Time lagged recurrent networks (TLRNs) are MLPs (Multilayer perceptron) extended with short-term memory structures. Here, a "Static" neural network that is MLP is endowed with dynamic properties. This, in turn, makes the network reactive to the temporal structure of information bearing signals. For a neural network to be dynamic, it must be given a memory. This memory is classified into "short-term" and "long-term" memory. Long-term memory is built into a neural network through supervised learning, whereby the information content of training data set is stored (in part or in full) in the synaptic weights of the network. However if the task at hand has a temporal dimension then some form of "short-term" memory is needed to make the network dynamic. One simple way of building short-term memory into the structure of a neural network is through the use of time delays, which can be applied at the input layer of the network (focused). A short-term memory structure transforms a sequence of samples into a point in the reconstruction space. This memory structure is incorporated inside the learning machine. This means that instead of using a window over the input data, PEs (processing elements) created are dedicated to storing either the history of the input signal or the PE activations. The input PEs of MLP are replaced with a tap delay line, which is followed by an MLP neural networks. This topology is called focused time-delay neural networks (TDNN). The focused topology only includes the memory kernels connected to the input layer thus only past of the input is remembered. The delay line of the focused TDNN stores the past sample of the input. The combination of tap delay line and the weights that connect the tap to the PEs of the first hidden layer are simply linear combiners followed by a static non-linearity.

3. EXPERIMENTAL WORK

This section showcase experimental set-up used and procedure adopted to carry out experiments .for data collection.

3.1. Experimental set-up

A TRUMPF LASERCELL 1005 (5 axis laser) model in MOGOD LASER Machining Pvt. Ltd., Bangalore was used for experimentation. The machine has a rated capacity of 4.5 KW.

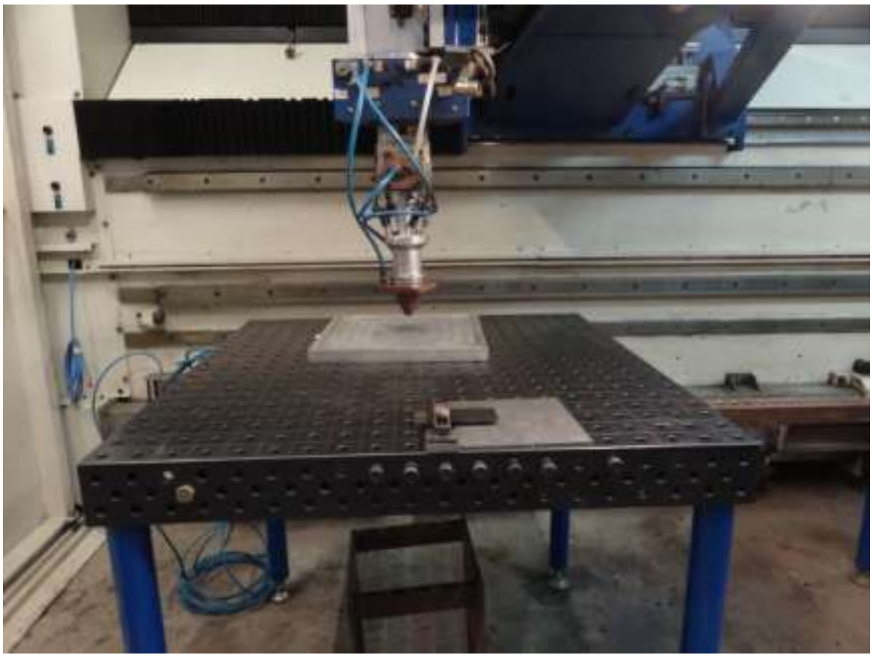

Figure 1: Experimental Set-up

3.2. Method of Experiment

Five input variables, namely laser power, scan speed, spot diameter, sheet thickness and number of scans are taken as the inputs and bending angle is considered as the output. Scan position (denoted by r) is the non-dimensional distance from the free edge of the work-piece. After laser scanning, deflections of the bent samples are measured using a coordinate measuring gauge and the bending angles are calculated. The parameters affecting bend angle are varied over its entire range to develop experimental data needed for system identification. These parameters are: Power (1.5KW to 3KW), Number of Passes (0 to 130), Scan Speed (0.6 to 1.8 m/min), Laser spot Diameter (4mm to 13mm), Frequency (500Hz to 20000Hz) and Sheet thickness (3mm and 5mm). CNC controlled unidirectional passes were given with approximate dwell time of 1.5sec between two consecutive passes. However in some cases, dwell time was more due to investigation of bend angle manually i.e. after every 10 passes, dwell time is more. The dwell time seems to have a very minor effect on the bend angle and therefore can be neglected from the investigation. Also a defocused laser beam was used to bend the sheet. The spot diameter was adjusted by varying the laser tip distance from the surface of the sheet. The effect of major significant factors i.e. laser energy and number of passes on bending angle is depicted in figure 2. In the experiment, total 78 pieces of sheet are used to generate 320 samples. A dataset comprising of 320 samples with five input (Power, Number of Passes, Scan Speed, spot diameter and sheet thickness) and one output (bending angle} is used for simulation. Frequency, being least significant factor affecting bend angle, is not considered for simulation.

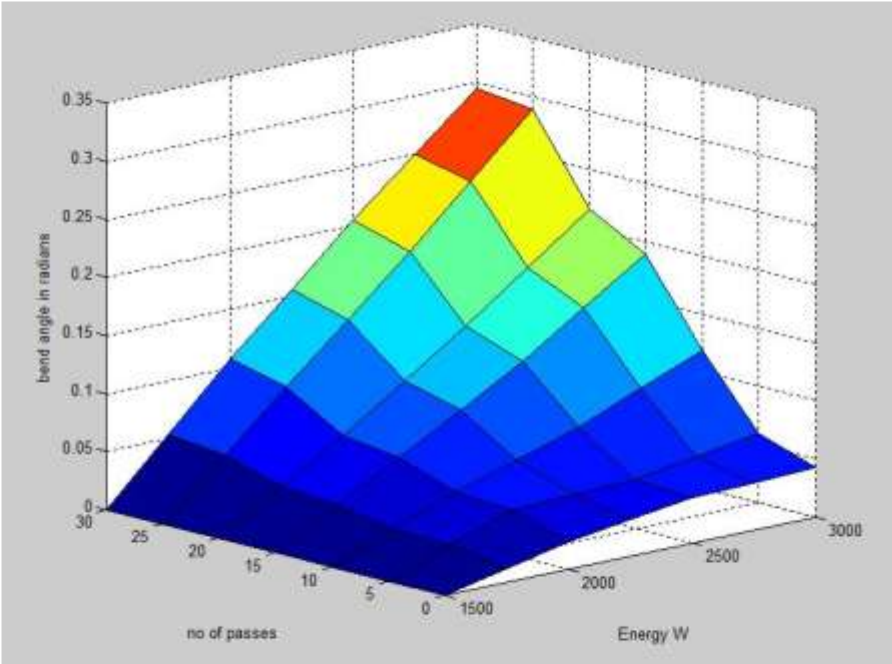

Figure 2: Effect of Energy and Number of Passes on Bending Angle

4. ESTIMATION OF NN MODEL

The simulation data constitute 320 samples in each set. This input-output experimental data has been obtained through vigorous experimentation carried out on TRUMPF LASERCELL 1005 model. In fact the process is multi-input - single output, where the output variable is bend angle and input variables are laser power, scan speed, spot diameter, sheet thickness and number of scans. As process exhibits nonlinear dynamics, versatile NN models are used to describe the system behavior. The weights are adjustable parameters of the system and they are determined from a set of examples through a process called training. The exemplars, or the training data as they are usually called, are the sets of inputs and corresponding desired outputs. The objective of the training is then to determine a mapping from the set of training data to the set of possible weights so that the network will produce prediction which in some sense is "Close" to the true output. The prediction error approach is based on the introduction of a measure of closeness in terms of a mean square error (MSE) criterion. When neural network has been trained, the next step is to evaluate it. This is done by standard method in statistics called *Independent Validation.* This method divides the available data into training set and a test set. The entire data is usually randomized first. The training data is then next split into two partition; the first partitions is used to update the weights in the network, and the second partition is used to assess (or cross validate) the training performance. The test data is then used to assess how well the network has generalized. The learning and generalization ability of the estimated neural network based model is assessed on the basis of certain performance measures such as normalized mean squared error (NMSE), correlation coefficient (r), and the regression ability of the neural network by visual inspection of the regression characteristics for different output of the system under study. Neurosolutions (version 5.0) is specifically used for obtaining results.

4.1 Performance Measures

- *Mean Square Error (MSE)*

The mean squared error is determined as

$$MSE = \frac{\sum_{j=0}^{P}\sum_{i=0}^{N}\left(d_{ij} - y_{ij}\right)^2}{NP} \quad \text{...(1)}$$

Where P is number of output PEs; N is number of exemplars in the data set; y_{ij} is network output for exemplar i at PE j; d_{ij} is desired output for exemplar i at PE j.

- *Normalized Mean Square Error (NMSE)*

The normalized mean squared error is determined as

$$NMSE = \frac{P\,N\,MSE}{\sum_{j=0}^{P}\frac{N\sum_{i=0}^{N} d_{ij}^{\,2} - \left(\sum_{i=0}^{N} d_{ij}\right)^2}{N}} \quad \text{..(2)}$$

Where P is number of output PEs; N is number of exemplars in the data set; MSE is mean square error; d_{ij} is desired output for exemplar i at PE j.

- *Correlation Coefficient (r)*

The size of the mean square error (MSE) can be used to determine how well the network output follows the desired output. But it doesn't necessarily reflect whether the two sets of data move in the same direction. For instance, by simply scaling the network output, we can change the MSE without changing the directionality of the data. The correlation coefficient (r) is used to solves this problem. Correlation coefficient between a network output x and a desired output d is given by

$$r = \frac{\frac{\sum_i\left(x_i - \bar{x}\right)\left(d_i - \bar{d}\right)}{N}}{\sqrt{\frac{\sum_i\left(d_i - \bar{d}\right)^2}{N}}\sqrt{\frac{\sum_i\left(x_i - \bar{x}\right)^2}{N}}} \quad \text{where } \bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \text{ and } \bar{d} = \frac{1}{N}\sum_{i=1}^{N} d_i \quad \text{...(3)}$$

The correlation coefficient is confined to the range [-1, 1]. When $r = 1$ there exists a perfect positive linear correlation between x and d, that is, they covary, which means that they vary by the same amount. When $r = -1$, there is a perfectl negative correlation between x and d, that is, they vary in opposite ways (when x increases, d decreases by the same amount). When $r = 0$, there is no correlation between x and d, i.e. the variables are called uncorrelated. Intermediate values of r describe partial correlations. For example a correlation coefficient of 0.88 means that the fit of the model to the data is reasonably good.

5. SIMULATION AND RESULTS

As data involves time domain parameters, recurrent and focused time lag recurrent neural network models are particularly selected for estimation of optimal model [2, 12]. An exhaustive and careful experimental study has been carried out to determine optimal configuration of the different NN models. All possible variations are tried to decide number of hidden layer and number of neurons in each hidden layer on the basis of performance measures. Training and testing percentage of exemplar are then varied to get optimum training-testing exemplars for each NN model. Different supervised learning rules, different transfer functions and different transfer functions in output layer are investigated in simulation. Finally, different error norms are applied to decide optimal neural network. After meticulous examination of performance measures like MSE, NMSE, correlation coefficient and the regression ability of the NN models on test data set, the optimal parameters are decided for the model. Neural network parameters so devised for optimal recurrent and focus time lag recurrent networks are displayed in Table 1 and Table 2 respectively.

Table 1: Parameters of Optimal Recurrent NN Model

Training - Testing % = 50-50

Max. epoch =1000, Memory - Axon, Norm = L_2

Sr. No.	Parameters	Hidden layer 1	Output layer
1	Processing Elements	4	1
2	Transfer function	tan h	tan h
3	Learning rule	Delta-bar-delta	Delta-bar-delta
4	Step size	0.01	0.1

5	Additive	0.001	0.01
6	Multiplicative	0.1	0.1
7	Smoothing	0.5	0.5

Figure 3 displays graph of average MSE verses epoch generated during simulation for optimal recurrent neural network and for the same training iteration, Table 3 showcase training error report for the best training run of recurrent neural network.

Table 2: Parameters of Optimal FTLR NN Model

Training - Testing % = 70-30, Max. epoch =1000
Focused, Memory - Gamma, Norm = L_1

Sr. No.	Parameters	Hidden layer 1	Output layer
1	Processing Elements	5	1
2	Transfer function	axon	axon
3	Learning rule	Momentum	Momentum
4	Step size	0.1	0.1
5	Momentum	0.7	0.7

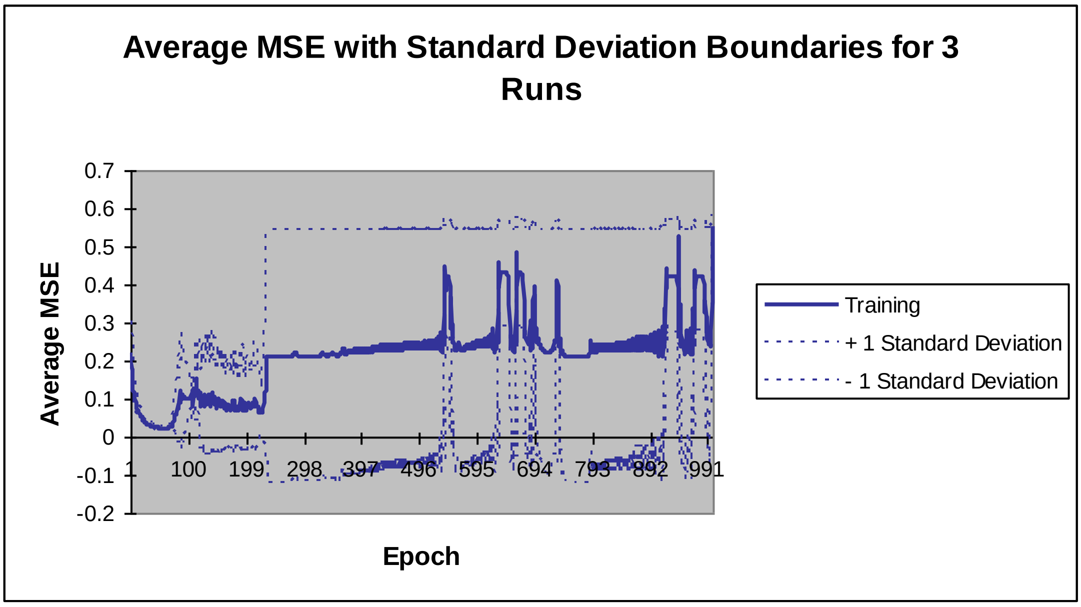

Figure 3: Variation of average MSE with epoch during training (Recurrent NN)

Table 3: Training Report of Optimal Recurrent NN Model

Best Network	*Training*
Run #	1
Epoch #	207
Minimum MSE	0.015597411
Final MSE	0.530913747

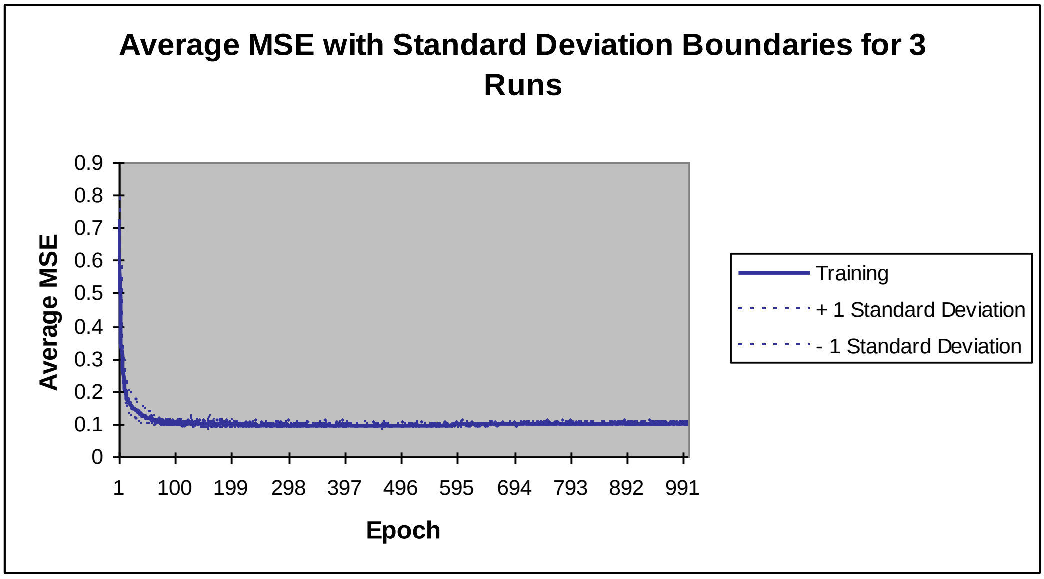

Figure 4: Variation of average MSE with epoch during training (FTLR NN)

Figure 4 displays graph of average MSE verses epoch generated during simulation for optimal FTLR neural network. For the same training iteration, Table 4 showcase training error report for the best training run of FTLR neural network. Identification capabilities of the neural networks used are decided on the basis of performance measures i.e. value of MSE, NMSE and r on testing data set and more importantly by the visual inspection of regression characteristic i.e. the graph showing desired output and actual network output.

Table 4: Training Report of Optimal Recurrent NN Model

Best Network	*Training*
Run #	2
Epoch #	285
Minimum MSE	0.087803357
Final MSE	0.101765051

Figure 5 displays regression characteristic of an optimal Recurrent NN. Figure 6 displays regression characteristics of optimal FTLR NN. Regression characteristics are displaying closeness between desired network output and actual network output thus it helps in deciding modeling ability of a neural network. Performance of optimal recurrent and focused time lag recurrent neural networks also can be accessed on the basis of performance measures on testing data set which is displayed in Table 5 and Table 6.

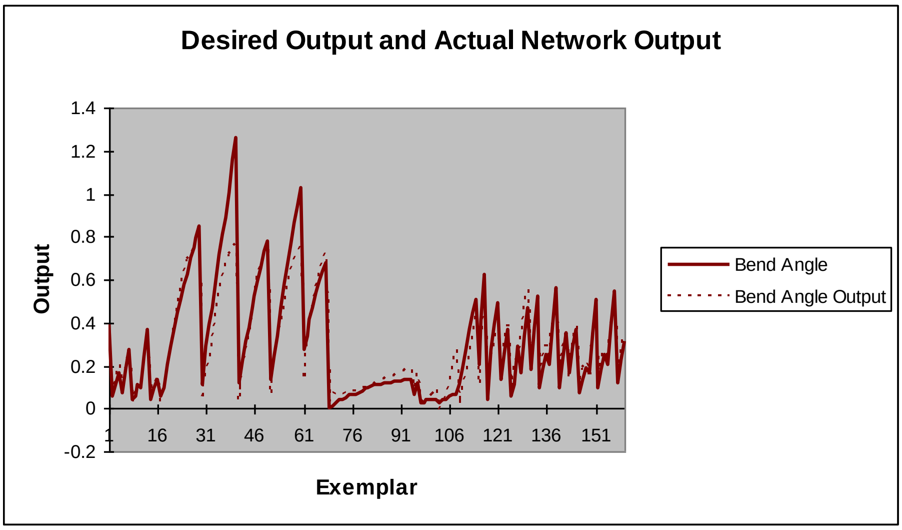

Figure 5: Desired output and Network output for testing data set (Recurrent NN Model)

Table 5: Performance of Optimal Recurrent NN Model (Test data)

Performance	*Bend Angle*
MSE	0.008415715
NMSE	0.121279045
MAE	0.057233748
Min Abs Error	0.00012137
Max Abs Error	0.500372024
r	0.950049203

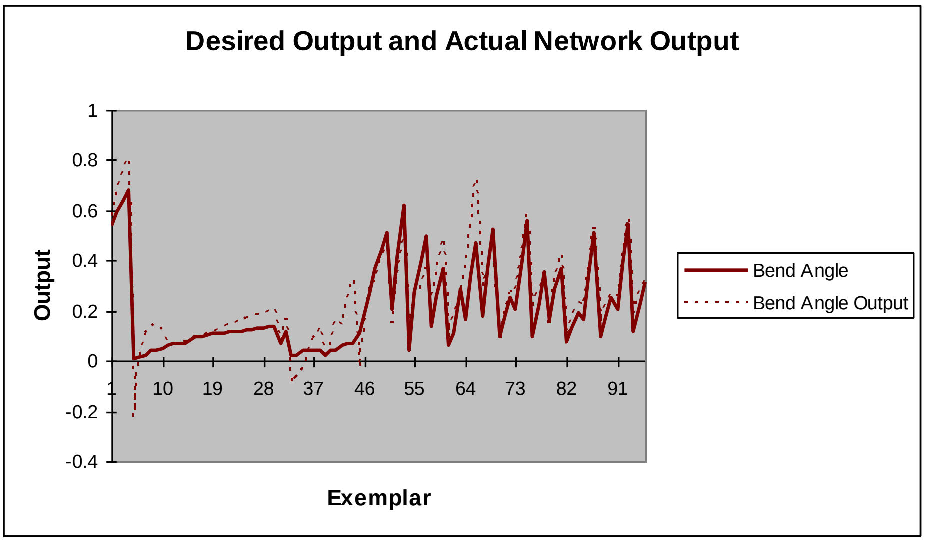

Figure 6. Desired output and Network output for testing data set (FTLR NN Model)

Table 6: Performance of Optimal FTLR NN Model (Test data)

Performance	*Bend Angle*
MSE	0.007077442
NMSE	0.23904937
MAE	0.062575325
Min Abs Error	0.000214215
Max Abs Error	0.264617217
r	0.90422239

Table 7: Comparison of Optimal NN Models

Sr. No.	Model	Test MSE	Test NMSE	Test r
1	Recurrent NN Model	0.008415	0.1212	0.9500
2	FTLR NN Model	0.00707	0.2390	0.9042

Table 7 enlists the MSE, NMSE and r for the estimated Recurrent and FTLR NN model on testing data set. Though values of MSE for both models is nearly matching but more important parameter being correlation coefficient and NMSE, here recurrent neural network clearly outperformed FTLR neural network. Through visual inspection of regression characteristics displayed in Figure 5 and Figure 6, it reveals that both the network models have capability of predicting bending angle to closeness more than 90%. Optimal recurrent NN model is capable of understanding nonlinear dynamics of laser forming process to an accuracy of 95% which in comparison to models reported by earlier researchers is far better. This fact is also evident from performance data given in Table 5 and 6.

6. CONCLUSION AND SCOPE

Exact mathematical modeling for laser forming process using conventional techniques was a distant dream due to certain nonlinearities. Hence artificial neural networks are currently being used to model the system. In system identification, the choice of neural network and data of system parameters for entire range holds a key in generating good model. With 320 samples of data having five input parameters and one output parameter, the identification results are very much encouraging. As per literature review reported so far, artificial intelligent techniques of modeling like neural networks and fuzzy logic could achieve accuracy up to 92.18 % and that too with few samples. (Maji et al. 2012). The present study has undertaken detailed systematic design procedure emphasizing step by step parameter optimization to obtain optimal neural network. It is seen that both selected neural networks are capable of identifying the laser forming process to a reasonable extent. From comparison table, it is observed that the value of MSE is very good for both networks. But more important parameter being correlation coefficient, it is seen that recurrent NN has highest value i.e. 0.9500. Thus an accuracy of model has reached to 95%. Also from visual inspection of regression characteristics of recurrent NN, it is clear that desired output closely matches with the actual network output whereas desired output of FTLR NN model lags a little in comparison to recurrent network. Thus it is concluded that recurrent NN model has an edge over other optimal neural network models reported so far. Still, there lies a great scope of work in identifying a laser forming process using other capable neural networks and to design an intelligent controller for automation.

7. ACKNOWLEDGMENT

The author thanks Mr. S. Shanmugam and Mr. Padmanabhan from WABCO India Pvt. Ltd and Mr. Swamy and Mr. Harshad Natu from MOGOD Laser Machining Pvt. Ltd for their overwhelming support and keen interest in this work.

REFERENCES

[1] Cybenko G.(1989), "Approximations by superposition of a sigmoidal functions." J. Math, Control Sig. & Syst., 2(4), pp.303-314.

[2] Dudul S. V.(2007), "Identification of a liquid saturated steam heat exchanger using focused time lagged recurrent neural network model." IETE J. of Res., Vol.53, no. 1, pp. 69-82.

[3] Li W., Yao Y. L.(2001), "Numerical and experimental investigation of convex laser forming process." J. of Manu. Process, Vol.3, no. 2, pp. 73-81.

[4] Cheng P. J., Lin S.C.(2000), "Using neural networks to predict bending angle of sheet metal formed by laser." Int. J. Machine Tools & Manuf., 40, pp.1185–1197.

[5] Vollertsen F., Geiger M., Li W. M.(1993), "FDM- and FEM-simulation of laser forming: a comparative study," Advanced Technology of Plasticity, Proc. of Fourth Int. Conf. on Tech. of Plasticity, pp. 1793–1798, 1993.

[6] Vollertsen F., Rodle M.(1994), "Model for the temperature gradient mechanism of laser bending." Laser Assisted Net Shape Engineering Proceedings of the LANE, 1, pp. 371–378.

[7] Maji K., Pratihar D. K.,. Nath A.K.(2012) "Analysis and synthesis of laser forming process using neural networks and neuron-fuzzy interference system." J of Methodology & Application, Soft. Compt. DOI 10.1007.

[8] Hu Z., Labudovic M., Wang H., Kovacevic R.,(2001) "Computer simulation and experimental investigation of sheet metal bending using laser beam scanning." Int. J. Mach. Tools Manuf., 41, pp. 589–607.

[9] Hu J., Dang D., Shen H., Zhang Z.(2012) "A finite element model using multi-layered shell element in laser forming." Opt Laser Tech., 44, pp. 1148–1155.

[10] Zhang L., Michaleris P.(2004) "Investigation of Lagrangian and Eulerian finite element methods for modeling the laser forming process." Finite Element Anal. Des., 40, pp. 383–405.

[11] Kao M. T.(1996), "Elementary study of laser sheet forming of single curvature." Master Thesis of Department of Power Mechanical Engineering, Tsing Hua University.

[12] Naikwad S. N., Dudul S. V. (2009), "Identification of a typical CSTR using optimal focused time lagged recurrent neural network model with gamma memory filter", Int. J. of Applied Compu. Intelligence and Soft Computing, Hindawi Publishing Corporation. Vol. 2009, 7 pages, doi:10.1155/2009/385757.