Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

An Innovative Method for Identification of Dynamic Systems Based on LoLiMoT

M. A. Sharbafi*, A. Mohammadi Nejad**

**Electrical and Computer Engineering Department, Azad University of Qazvin*
Qazvin, Iran, (e-mail: m.sharbafi@ece.ut.ac.ir)
*** Electrical Engineering Department, Khaje Nasir Toosi University of Technology*
Tehran, Iran, (e-mail: a.mohammadinejad@ee.kntu.ac.ir)}

Abstract: There are many methods in identification developing every day. But identification of dynamic systems has still remained a complex open problem. One of the new effective methods of identification in nonlinear problems is identification with Neurofuzzy approach. Compared with classic neural network and wavelet network this method is faster and more accurate, demonstrated with an example in this paper. The main aim of this paper is developing a Locally Linear Model Tree based algorithm which can be used in structure identification. Our method inspired from the concepts utilized in definition of this Neurofuzzy network and its quality was shown by a bench mark structure identification problem.

1. INTRODUCTION

When the mission is designing a controller for a plant, uncertainty of data is inevitable; also even after identification; some unconsidered events may change its characteristics like aging, friction, and etc. Noises and disturbances may cause some difficulties in the systems too. Nonlinear system identification has been a topic of wide interest in the past; however considerable progress both in theory and practice has been reported during the past decade (Chen *et al*., 1990), (Chen *et al*., 1992). In recent years, data-driven and rule-base approaches are utilized more than model-based approaches to decision making (Nelles., 1997), (Trabelsi. *et al*., 2004), (Nelles., 1996), (Eppler *et al*., 1999).

These new methods have shown better results comparing with the classic ones and some neural network based methods such as Multi Layer Perceptron (MLP). Recently some wavelet networks are presented too (Zhang. *et al*., 1992). The main method that we emphasize on, is a Neurofuzzy method named Locally Linear Model Tree (LoLiMoT). Although the advantages of this method were investigated before (Hafner. *et al*., 1999), (Schmidt. *et al*., 1998), (Schwarz. *et al*., 1997), (Miller. *et al*., 1999), in this paper we show its ability by implementing MLP, Wavelet network and LoLiMoT in a problem and comparing them. The results show the performance of LoLiMoT and we use it for the following research. The final goal is to present a method for structure identification.

Structure identification is one of the most complex problems in identification and the fewer data, the more difficult problem. As described bellow LoLiMoT uses discretizing the space to fit a model to the nonlinear system and it selects the partitioning with minimum error. A new algorithm based on the basic concepts of this method is presented in this paper to identify the dynamic of the system. Note that a dynamic system means a system which uses its previous outputs in addition to the input vector. Identifying which part of the previous data has a significant effect in producing the output, is a difficult structure identification problem. The results of testing this approach, in a hand-designed nonlinear dynamic problem and a bench mark one, show its prefect quality.

The rest of this paper is organized as follows: Section 2 describes a summary about LoLiMoT. In section 3, we compare the results of using wavelet network, MLP and LoLiMoT. In section 4, our innovative approach is explained. Finally, Section 5 discusses the simulation results obtained by this way and section 6 concludes this paper.

2. IDENTIFICATION WITH LOLIMOT

Modelling and identification of nonlinear dynamic systems is a challenging task because nonlinear processes are unique in the sense that they do not share many properties. An alternative approach is to design a nonlinear model consisting of several linear functions. The major output function is derived from a combination of linear models. Many training algorithms and structures are suggested for the mentioned networks such as TSK, ANFIS, LoLiMoT and PLN networks (Nelles., 1997), (Eppler. *et al*., 1999), (Jang., 1993), (Sugeno. *et al*., 1988a, b), (Nelles., 2001). Because of the accuracy, speed and various additional advantages for modelling and identification of dynamic processes, we prefer LoLiMoT (Nelles., 2001). In the following, the identification of nonlinear dynamic processes using LoLiMoT algorithm is described.

The locally linear modelling approach is based on divide-and-conquer strategy. The network structure of a locally linear Neurofuzzy model (Nelles., 1997), (Nelles., 2001) is depicted in Fig. 1. Each neuron realizes a locally linear model

978-3-902661-00-5/08/$20.00 © 2008 IFAC

10.3182/20080706-5-KR-1001.2870

(LLM) and an associated validity function that determines the region of validity of the LLM. The validity functions form a partition of unity, i.e., they are normalized such that:

$$\sum_{i=1}^{M} \varphi_i(\underline{x}) = 1. \tag{1}$$

For any model input $\underline{x}$, the output is calculated as:

$$\hat{y} = \sum_{i=1}^{M} (w_{i0} + w_{i1}x_1 + \cdots + w_{in}x_n)\varphi_i(\underline{x}). \tag{2}$$

Thus, the network output is calculated as a weighted sum of the outputs of the local linear models where the weights w_{ij} are linear network parameters and φ_i is interpreted as the operating point dependent weighting factors. The network interpolates between different LLMs with the validity functions. The validity functions are typically chosen as normalized Gaussians. If these Gaussians are furthermore axis-orthogonal the validity functions are

$$\varphi_i(\underline{x}) = \frac{\mu_i(\underline{x})}{\sum_{j=1}^{M} \mu_j(\underline{x})}. \tag{3}$$

With

$$\mu_i(\underline{x}) = \exp\left(-\frac{1}{2}\left(\frac{(x_1 - c_{i1})^2}{\sigma_{i1}^2} + \cdots + \frac{(x_1 - c_{in})^2}{\sigma_{in}^2}\right)\right) \tag{4}$$

The centres and standard deviations are nonlinear network parameters. In the fuzzy system interpretation each neuron represents one rule. The validity functions represent the rule premise and the LLMs represent the rule consequents.

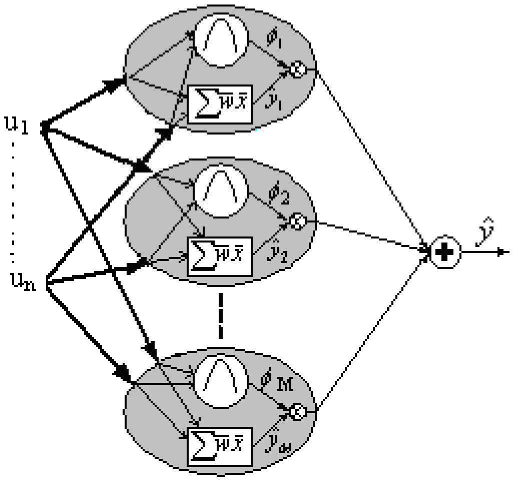

Fig. 1. Network structure of a local linear Neurofuzzy model with M neurons.

The LoLiMoT algorithm consists of an outer loop in which the rule premise structure is determined and a nested inner loop in which the rule consequent parameters are optimized by local estimation. In this loop the worst partition is selected in each step and split to two parts in every direction, and then the best new division is chosen to continue till the stop condition (when the desired goal is gained) is reached.

We can summarize the algorithm in following steps:

1. Start with an initial model: Construct the validity functions for the initially given input space partitioning and estimate the LLM parameters by the local weighted least squares algorithm. Set M to the initial number of LLMs. If no input space partitioning is available at the beginning, then start with a single LLM which in fact is a global linear model since its validity function covers the whole input space with $\varphi_1(\underline{x}) = 1$, and set $M = 1$.

2. Find the worst LLM: Calculate a local loss function for each of i ($i = 1,..., M$) LLMs. The local loss functions can be computed by weighting the squared model errors with the degree of validity of the corresponding local model.

3. Check all divisions: The first LLM is considered for further refinement. The hyper-rectangle of this LLM is split into two halves with an axis-orthogonal split. Divisions in each dimension are tried. For each division ($\dim = 1,..., n$), the following steps are carried out:

(a) Construction of the multi-dimensional Mean Square Errors (MSE) for both hyper-rectangles.

(b) Construction of all validity functions.

(c) Local estimation of the rule consequent parameters for both newly generated LLMs.

(d) Calculation of the loss functions for the current overall model.

4. Find the best division: After checking the n alternatives the best one is selected (the one with the most improvement). The validity functions constructed in Step 3(a) and the LLMs optimized in Step 3(c) are adopted for the model. The number of LLMs is incremented: $M \rightarrow M + 1$.

5. Test of converging: If the termination criterion is met then stop, else go to Step 2.

For the termination criterion various options exist, e.g., a maximal model complexity, that is a maximal number of LLMs, statistical validation tests, or information criteria. We use the accuracy validation test to terminate the learning process.

Fig. 2 illustrates the operation of the LoLiMoT algorithm in the first four iterations for a two-dimensional input space and clarifies the reason for the term ”tree” in the acronym LoLiMoT. In practical two features make LoLiMoT extremely fast. First, at each step all possible LLMs are not considered for division. Rather, Step 2 selects only the worst LLM whose division most likely yields the highest performance gain. For example, in iteration 3 in Fig. 2 only LLM 3-2 is considered for further refinement and all other LLMs are kept fixed. Second, in Step 3 the local estimation approach allows to estimate only the parameters of those two LLMs which are newly generated by the division. For example, when in iteration 3 in Fig. 2 the LLM 3-2 is divided into LLM 4-2 and 4-3 the LLMs 3-1 and 3-3 can be directly passed to the LLMs 4-1 and 4-3 in the next iteration without any estimation.

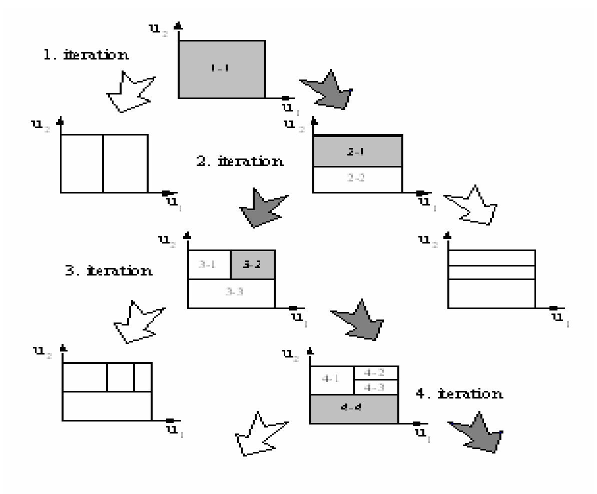

Fig. 2: Operation of the LOLIMOT structure search algorithm in the first four iterations for a two-dimensional input space (p = 2).

The training algorithm LoLiMoT is found out to be rapid, precise, self tuned and more user friendly than other conventional methods for training of Neurofuzzy networks which make it more acceptable in online control applications (Nelles., 1997), (Nelles., 1996).

3. COMPARISON BETWEEN LOLIMOT, WAVELET AND MLP

In this section, we consider a problem utilized in (Postalcioglu. *et al.*, 2003). The author considered this example to show the power of identification of nonlinear problem with wavelet network. The function is formulated as follows:

$$f(x) = \begin{cases} -2.186x - 12.864 & x \in [-10,-2[\\ 4.246x & x \in [-2,0[\\ 10e^{(-0.05x-0.5)}\sin(x(0.03x+0.7)) & x \in [0,10] \end{cases} \tag*{5)}$$

In (Postalcioglu. *et al.*, 2003), the wavelet network is considered with 10 neurons and the wavelet function is the differentiation of the Gaussian formulated as (6) with the network structure depicted in Fig. 3. We trained this network with the mentioned conditions and the result of identification is illustrated in Fig. 4. In this figure, the accuracy is computed as the MSE resulted from the difference between the real and identified output.

The important problem in this method was the sensitivity of the training to the initial condition. With little changes in initial condition and learning parameters the network diverges. This is the most significant drawback of this kind of identification.

$$\phi(x) = -xe^{-\frac{1}{2}x^2} . \tag{6}$$

The next step is identification of this nonlinear function with a favourite neural network method, MLP. The real and identified output was depicted in Fig. 5. Ten neurons are considered and the weights are trained by gradient descent similar to training method in wavelet network. The difficulties in MLP are the number of iterations and neurons and tuning the learning rate. The learning rate is assumed equal to 0.03 and the training was too sensitive to this parameter. Another interesting point in this problem is what the best training method is. Some faster methods such as Levenberg-Marquardt back propagation have a drawback which is involving in local minima.

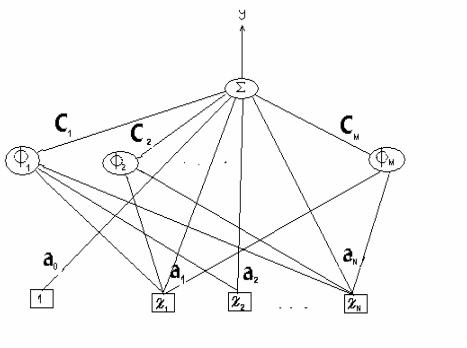

Fig. 3. The structure of a Wavelet network model

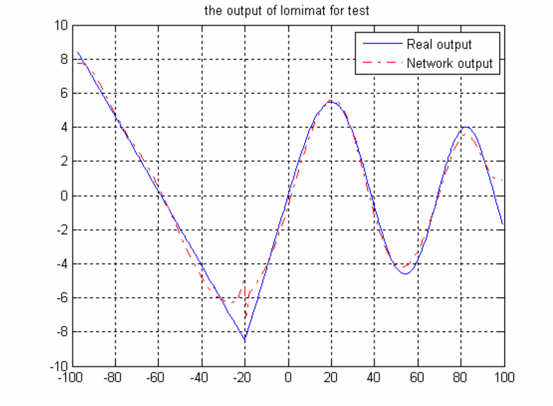

Fig. 4. The output of the wavelet network (dashed) compared with real outputs (iteration=600, neuron=10, accuracy=0.065)

At the end LoLiMoT is applied for identification. The results are illustrated in Fig. 6. In this method in addition to its accuracy and time of learning, there is not any problem in defining the number of the neurons and learning rates. It is clear that the quality of the identification is better without any sensitivity problem. In Fig. 6 the system was identified with 15 neurons. Note that previous networks have worse performance with more than 10 neurons and increasing the neurons does not conclude better results for every problem. Over parameterization, long training time and local minima are some problems happened with more neurons. The initial condition does not matter here and so on. Least Square is the way of updating the weights which is another definition of gradient descent when the error is mean square error.

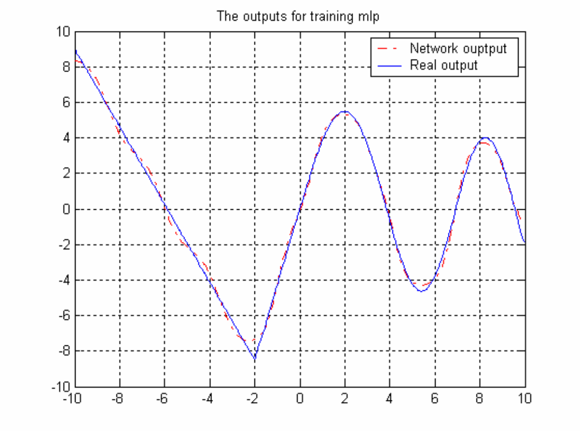

Fig. 5. The output of the MLP (dashed) compared with real outputs (iteration=1300, neuron=10, accuracy=0.082)

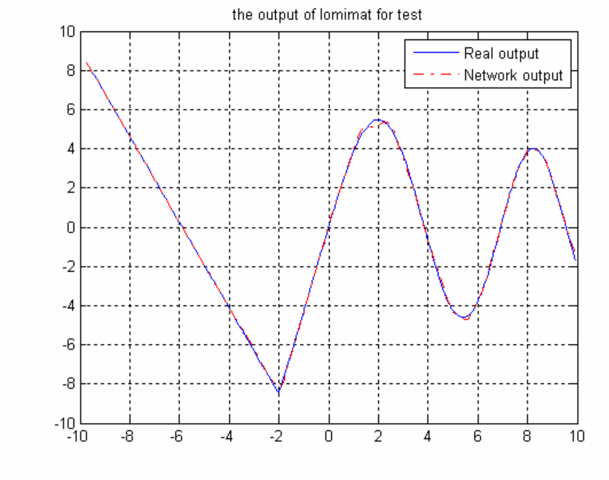

Fig. 6. The output of the LoLiMoT (dashed) compared with real outputs (iteration=15, neuron=15, accuracy=0.0088)

4. THE ALGORITHM OF STRUCTURE IDENTIFICATION

In this part our new method of dynamic system identification is presented. At first the concepts which are used in Neurofuzzy networks are developed and the unnecessary data from the basic data set are removed. This led us to achieve the architecture of the system. Two points are considered to determine the useful data.

Finding the effective portion of the input set producing the output is the first important step. With a wrong input selection the best identification methods may loose their capabilities. In neural network there is not any sense about the fitness of input set and it is not detectable easily. But in Neurofuzzy networks we can determine this portion that will be shown in the following description.

As it was described before, the output of the network is determined by (7). In this formula the inputs multiply with weights and the multiplication of their summation and memberships add to each other to make the output.

$$y = \sum_{i=1}^{n} \phi_i(x) \sum_{k=1}^{r} x_k w_{ik}$$

(7)

Now if we rearrange this formula as (8) the portion that was discussed will be appeared. In this formula we changed the place of inputs and memberships. Thus it is obvious that each input is multiplied by a factor and their summation makes the output.

$$y = \sum_{k=1}^{r} x_k \sum_{i=1}^{n} \phi_i(x) w_{ik} \tag{8}$$

From this point of view when the inputs are normalized before entering the network, their coefficients calculated by (9), show the effect of each input in producing the output. In other words, if W_k is near zero k^{th} input's role can be ignored and we can understand that this input does not exist in the dynamic of the system. Therefore the first detection method for determining the necessity of each input is calculating W_k and comparing its quota with a threshold.

$$W_k = \sum_{i=1}^{n} \phi_i(x) w_{ik} \tag{9}$$

Our second strategy is deleting inputs that do not participate in dividing process. When you enter an unnecessary data to the input set the space of inputs will not split in that direction. The reason is that with dividing in that orientation the error will not be reduced. Hence the scarcity of division in one direction can show that related input is not useful.

Thus in evaluating the information with this approach we can decrease the importance of some data that the partitions will not be divided in those directions.

Two reasonable techniques are described to evaluate the importance of each input. Combination of them results in an applicable method leading to a more reliable measurement tool. In our approach each strategy computes an importance degree for each input. Then our final evaluating gauge will be made up of these values. We named it EG (Evaluating Gauge). The final step is comparison of each input's EG and two lowest ones which are less than the threshold will be removed from the set. This omission is done one by one, to understand the error reduction of eliminating each one. In other words we try to identify the system with evaluating each $(n-1)$ combinations of data set. Therefore the elimination which reduces the identification error more than others can be discovered. This element will be removed and so on. When EG of all choices remain more than threshold the process will be stopped.

5. RESULTS

At first this method is applied on the system defined by (10). In this formula a nonlinear system with complex dynamic is depicted. The input vector (X) is defined as shown in (11) including four useless data. Both strategies can determine the dynamicity of the function. Thus the problem can be solved easily with the algorithm described in previous section. It means that the space is not partitioned in unused data directions and the portions of these inputs are insignificant in producing output.

$$y(t) = y(t-1)u(t-1)^{\frac{1}{2}} + \sin(u(t-4))e^{-y(t-2)}$$
(10)

$$X = \left[y(t-1) \quad y(t-2) \quad u(t-1) \quad \ldots \quad u(t-6) \right] \qquad (11)$$

The results of identification are depicted in figures 7 to 9. They illustrate that the accuracy is very satisfying and the identification error is very low for the test data. This precise identification is gained by powerful identification tool and the strategy of detecting the dynamic of the system. It is clear that after 18 iterations the errors become nearly constant and it shows that the LoLiMoT network can identify the system with 18 neurons.

We tested many such cases and this method of structure identification was perfect for all of them. In these problems more than 400 data were considered for identification. With less data the identification problem will become more complicated.

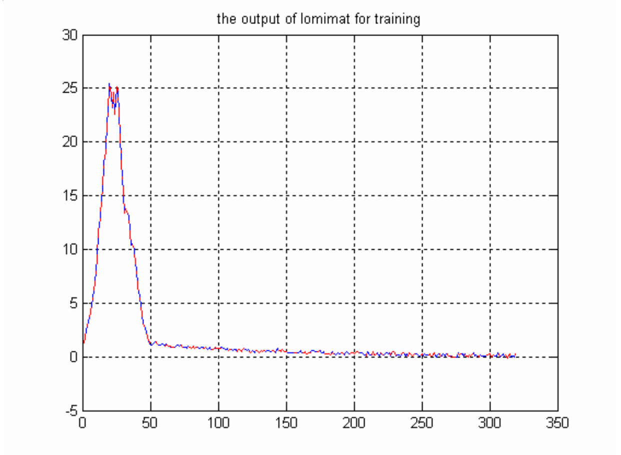

Fig. 7. The output of Real system (solid line) and LoLiMoT (dashed line) for training data

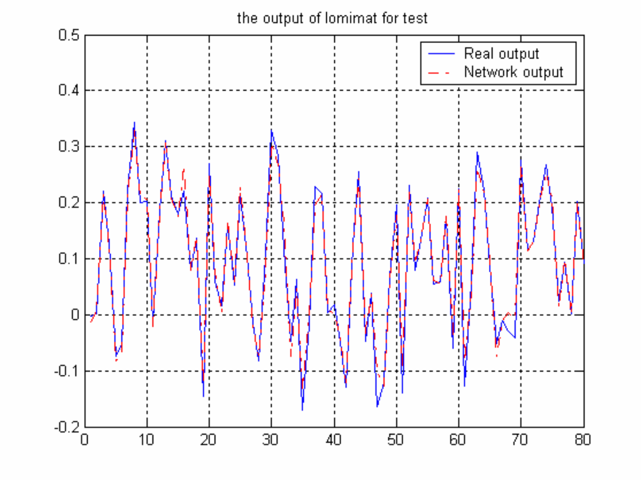

Fig. 8. The output of Real system (solid line) and LoLiMoT (dashed line) for testing data

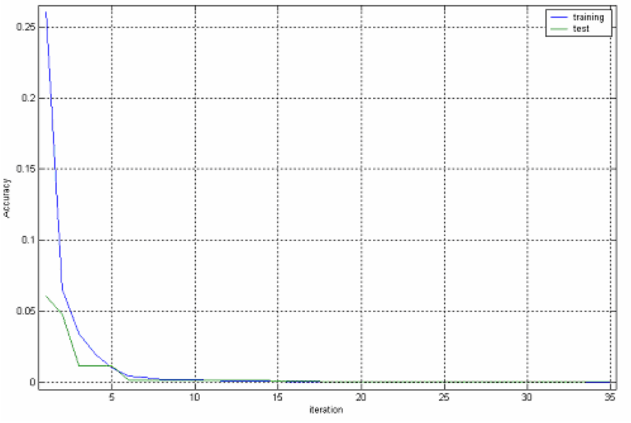

Fig. 9. The accuracy of identification for data of test and training during 35 steps (The number of neurons becomes 35) the accuracy of training is 0.000128 and the error of test is 0.000337.

The performance of this method is investigated by applying it in an unsolved structure identification. It was the box-jenkins problem which is one of the benchmark problems in this field (Sugeno. *et al.*, 1988a). In this problem we have only 290 data and the suggested input vector is defined as (12).

$$X = \left[y(t-1) \quad \ldots \quad y(t-4) \quad u(t-1) \quad \ldots \quad u(t-6) \right]$$
(12)

In (12), there are 10 suggested inputs for identification. We should determine the best set. The most difficult part of this problem is that the number of data is not sufficient respect to the X vector with 10 elements (Sugeno. *et al.*, 1988b). Most of the researchers said that the main set should be $X = [y(t-1) \quad u(t-4)]$ but Sugeno and Yasukawa think that the input should be $X = [y(t-1) \quad u(t-3) \quad u(t-4)]$. We applied Our Method on this system and the resulted set was $X = [y(t-1) \quad y(t-2) \quad y(t-3) \quad u(t-2)]$. The correct response of this problem is not clear, but we can compare the results of identification with these three sets. Our approach could reach to the accuracy of 0.16 for test regarded to 0.55 for Sugeno suggestion that is better than another suggestion.

The results of identification with our structure and Sugeno's ones are illustrated in figures 10 to 14. The result of identification for training data is similar for both methods and it is shown in Fig. 10 only for one of them. Suitable quality of this algorithm is apparent in this figure. The performance is desirable for structure suggested by Sugeno too. But the main parameter that shows the power of identification is comparing the results for test data displayed in Fig. 11 and Fig.13. The better quality of our structure is obvious in these figures. Fig. 12 and Fig. 14 show the accuracy of two methods. These figures demonstrate that our strategy in structure identification is more successful than Sugeno's one.

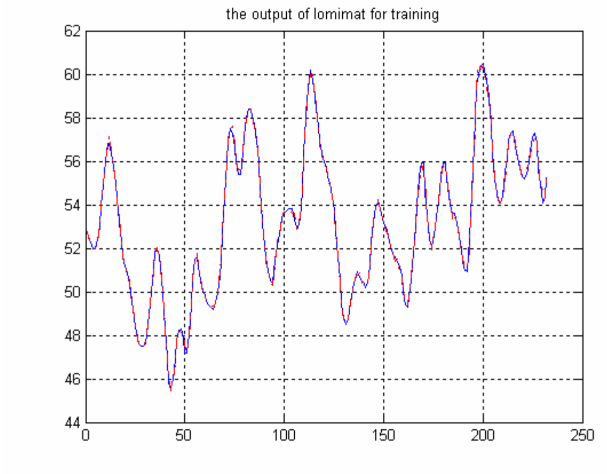

Fig. 10. Identification result of *box-jenkins* problem with LoLiMoT for training data. The structure Identified by our suggested method.

6. CONCLUSION

We suggested an ad-hoc method inspired from some logic heuristics for structure identification with LoLiMoT algorithm. Using LoLiMoT as a powerful identification tool is sufficiently frequent. We compared this method with MLP and Wavelet Network to confirm its superiority. With respect to this fact, we extended the concept of identification with LoLiMoT to retain an algorithm for structure identification. We used two key points in LoLiMoT construction to reach a strategy to detect the dynamic of the unknown systems. The results of the structure identification for a nonlinear system with complex dynamic and more accurate identification in box-jenkins problem which is a benchmark problem in identification demonstrated the perfect quality of this algorithm. More concentration on defining this method to expand the method to a general structure identifier can be gained in future.

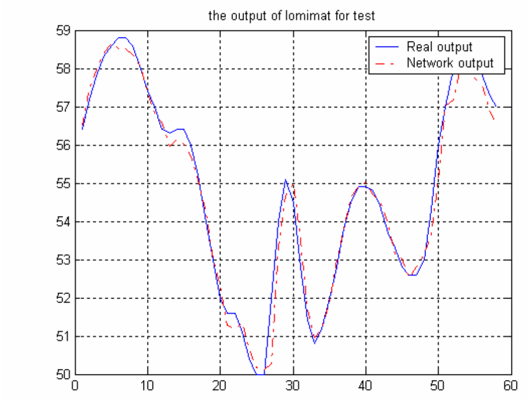

Fig. 11. Identification result of *box-jenkins* problem with LoLiMoT for testing data. The structure Identified by our suggested method.

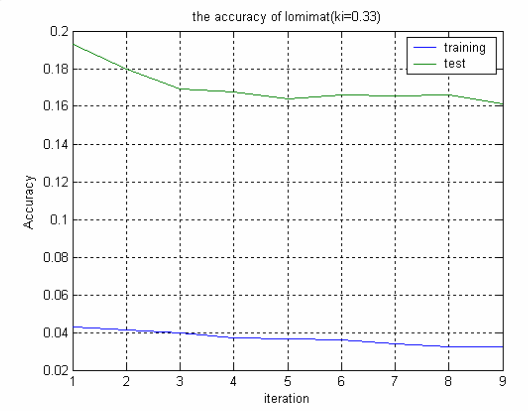

Fig. 12. Identification accuracy of *box-jenkins* problem with LoLiMoT. The structure Identified by our suggested method (k_i is the variance of Gaussians).

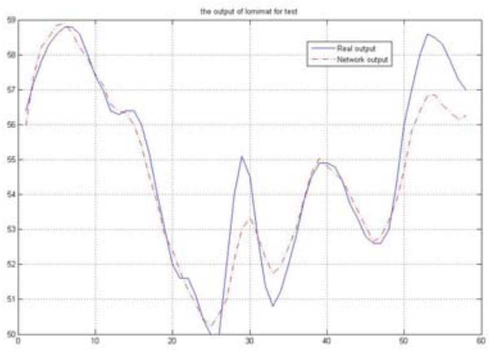

Fig. 13. Identification result of *box-jenkins* problem with LoLiMoT for testing data. The structure Identified by Sugeno suggested structure.

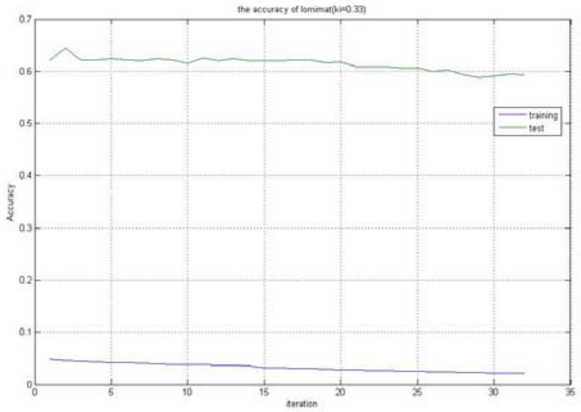

Fig. 14. Identification accuracy of *box-jenkins* problem with LoLiMoT for testing data. The structure Identified by Sugeno suggested structure (k_i is the variance of Gaussians).

REFERENCES

Chen, S., Billings, S.A., and Grant, P.M. (1990). Non-linear System Identification using Neural Networks. *International Journal of Control*, **51**, 6, 1191–1214.

Chen, S., and Bilings, S.A. (1992). Neural Networks for Nonlinear Dynamic System Modeling and Identification. *International Journal of Control*, **56,** 2, 319–346.

Eppler, W., and Beck, H. N. (1999). Piecewise Linear Networks (PLN) for Function Approximation. in Proc. IJCNN'99, Washington, USA, 388-391.

Hafner, M., Schuler, M., and Isermann, R. (1999). Fast Neural Networks for Diesel Engine Control Design. in 14th IFAC World Congress, Beijing, China.

Jang, J. S. R. (1993) Adaptive-network-based fuzzy inference system. IEEE Trans. System, Man and Cybernetics, **23**, 3, 665-685.

Miller, N., and Nelles, O. (1999). Closed-Loop Ignition Control Using On-Line Learning of Locally-Tuned Radial Basis Function Networks. in Proc. American Control Conference (ACC), San Diego, USA, 1356-1360.

Nelles, O. (1996). Local linear model tree for on-line identification of time variant nonlinear dynamic systems. in Proc. International Conference on Artificial Neural Networks (ICANN), Bochum, Germany, 115-120.

Nelles, O. (1997). Orthonormal Basis Functions for Nonlinear System Identification with Local Linear Model Trees (LoLiMoT). in Proc. IFAC Symposium on System Identification, Kitakyushu, Fukuoka, Japan.

Nelles, O. (2001) Nonlinear System Identification. Springer-Verlag, Berlin.

Postalcioglu, S. and Becerikli, Y. (2003). Nonlinear System Modeling Using Wavelet Networks. in Proc. of International Conference on Signal processing.

Schmidt, M., and Nelles, O. (1998). Filtering and Deriving Signals Using Neural Networks. in Proc. American Control Conference (ACC), Philadelphia, USA, 2730-2731.

Schwarz, R., and Nelles, O. (1997). Increasing Signal Accuracy of Automatic Wheel-Speed Sensors by On-Line Learning. in Proc. American Control Conference (ACC), Albuquerque, USA, 1131-1135.

Sugeno, M., and Kang, G. T. (1988a). Structure identification of fuzzy model. *Fuzzy Sets and Systems*, **28**, 15-33.

Sugeno, M., and Kang, G. T. (1988b). Structure identification of fuzzy model. *Fuzzy Sets and Systems*, **28**, 15-33.

Trabelsi, A., Lafont, F., Kamoun, M., and Enea, G. (2004). Identification of Nonlinear Multivariable Systems by Adaptive Fuzzy Takagi-Sugeno Model. *International Journal of Computational Cognition*, **2**, 3, 137–153.

Zhang, Q., and Benveniste, A. (1992). Wavelet Networks. *IEEE Trans. Neural Networks*, **3**, 6, 889-898.

http://neural.cs.nthu.edu.tw/jang/benchmark/

http://www.esat.kuleuven.ac.be/~tokka/daisydata.htm