NEURO-FUZZY MODELING FOR DYNAMIC SYSTEM IDENTIFICATION

JYH-SHING ROGER JANG
CS Dept., Tsing Hua Univeristy, Hsinchu, Taiwan

This paper presents the continued work of a previously proposed ANFIS (Adaptive Neuro-Fuzzy Fuzzy Inference System) architecture with emphasis on the applications to dynamic system identification. We demonstrate the use of ANFIS for the hair dryer modeling problem and compare its performance with the ARX model.

1 Introduction

System identification[1] is the process of constructing a model to predict the behavior of a target system. Conventional system identification techniques are mostly based on linear models with fast computation and rigorous mathematical support. On the other hand, neuro-fuzzy modeling represents nonlinear identification techniques that require massive computation but without mathematical proofs of convergence to global minima or the like. This paper applies two representative approaches (ANFIS and ARX) from both disciplines and compare their performance on a classic system identification problem of hair dryer modeling.[1].

This paper is organized into five sections. In the next section, the basics of ANFIS are briefly introduced. Section 3 explains the problem of hair dryer modeling and how to use the ARX model to find a linear model. Section 4 exhibits the use of ANFIS for the same problem and compare the results with the ARX model. Section 5 gives concluding remarks.

2 ANFIS

A first-order Sugeno fuzzy inference system[2] with two fuzzy rules can be expressed as

Rule 1: If X is A_1 and Y is B_1, then $f_1 = p_1x + q_1y + r_1$,
Rule 2: If X is A_2 and Y is B_2, then $f_2 = p_2x + q_2y + r_2$.

Figure 1(a) illustrates the fuzzy reasoning mechanism that infers an output f from a given input vector $[x, y]$. Accordingly, we can derive an equivalent adaptive network[3] representation, call ANFIS[4] (Adaptive Neuro-Fuzzy Inference System), as shown in Figure 1(b). Layer 1 computes the membership grades; layer 2 combines the membership grades to form the firing strengths; layer 3 normalizes the firing strengths; layer 4 generates the contribution from

0-7803-3687-9/96©1996 IEEE

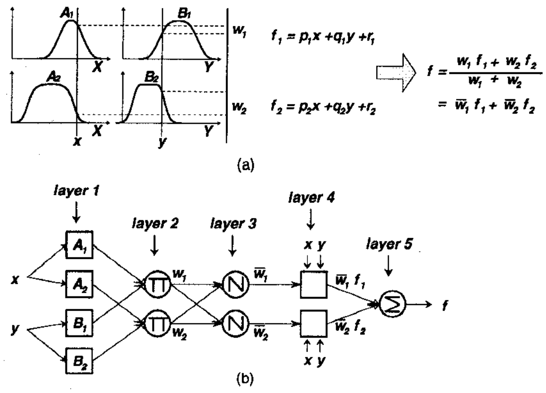


Figure 1: *(a) Fuzzy reasoning mechanism; (b) corresponding ANFIS.*

each rule; and layer 5 produces the final output. Note that the modifiable parameters in layer 1 determine the shapes and positions of membership functions, and those in layer 4 specify the output linear equation of each rule. Obviously layer-1 parameters are nonlinear, so we can use the back-propagation gradient descent[5] to update them. For those linear parameters in layer 4, we can apply more efficient least-squares methods[1] to identify them. This is the hybrid learning method[4] used in our simulation. In fact, other advanced techniques in nonlinear regression such Gauss-Newton method, Levenberg-Marquardt method, and the extended Kalman filter algorithm can also be applied to ANFIS directly.[3]

3 Hair Dryer Modeling and ARX Models

The system identification problem under consideration is a laboratory device called Feedback's Process Trainer PT326, as described in Chapter 17 of Ljung's seminal book[1] on system identification. The device's function is like a hair dryer: air is fanned through a tube and heated at the inlet. The air temperature is measured by a thermocouple at the outlet. The input $u(k)$ is the voltage over the a mesh of resistor wires to heat incoming air. The output $y(k)$ is the outlet air temperature. The device is well-behaved: it has reasonably simple dynamics with small disturbances, and allows high-precision measurements.

To proceed system identification on the hair dryer device, first we need to collect input-output data pairs. The sampling time is 0.08 second. One thousand input-output data points were collected from the process as the input was was chosen to be a binary random signal shifting between 3.41 and 6.41

V. The probability of shifting the input at each sample was 0.2.

A conventional method is to remove the means from the data and assume a linear model of the form:

$$y(k)+a_1y(k-1)+\cdots+a_{n_a}y(k-n_a)=b_1u(k-d)+\cdots+b_{n_b}u(k-d-n_b+1),$$

where a_i ($i=1$ to n_a) and b_j ($j=1$ to n_b) are linear parameters to be determined by least-squares methods. This structure is called the ARX model[1] and it is exactly specified by three integers $[n_a, n_b, d]$, where n_a is equal to the number of poles; n_b-1 is the number of zeros; and d is the pure time-delay in the system. Because of the availability of inexpensive and fast CPUs, now we can almost do an exhaustive search on the model structure $[n_a, n_b, d]$ using a personal computers.

To find an ARX model for the hair dryer device, the data set was divided into training ($k=1$ to 300) and test ($k=301$ to 600) data sets. We performed an exhaustive search on the ARX structure $[n_a, n_b, d]$, where each of the structure parameters is allowed to changed from 1 to 10 independently. In other words, we constructed 1000 ARX models whose parameters are identified by least-squares methods. The best model was selected as the one with the smallest test error. This process took about 7 seconds on a Pentium-120 PC with 32MB RAM, and the best ARX model is $[n_a, n_b, d]=[5,9,2]$, with training RMSE (root-mean-squared error) 0.1142 and test RMSE 0.07021. Figure 2(a) demonstrates the fitting results of the best ARX model, where the mean value of the data set is added back to compare to the original signals.

The ARX model is inherently linear and the most significant advantage is that we can perform model structure and parameter identification rapidly. The obtained results in Figure 2(a) appear to be satisfactory. However, if a better performance is desired, then we have to resort to nonlinear models. In the next section, we shall explore the use of ANFIS to see if we can push the performance level by introducing nonlinearity into the model.

4 ANFIS Models

To use ANFIS for system identification, the first thing we need to do is **input selection**, i.e., to determine which variables should be the input arguments to the ANFIS model. Once the input arguments are fixed, then we can specify the ANFIS model structure, such as the style for input space partitioning, the numbers and types of membership functions on each input, and so on.

For the dryer modeling problem, we can partition the input candidates

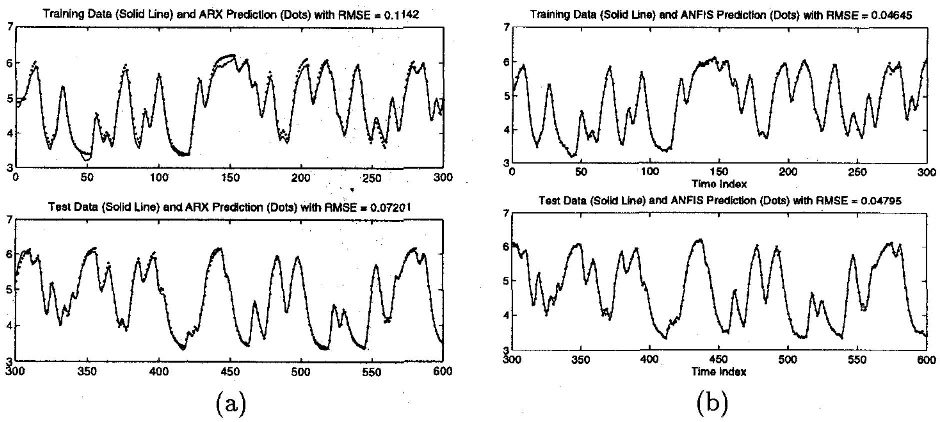


Figure 2: *Hair dryer modeling: (a) ARX model; (b) ANFIS model.*

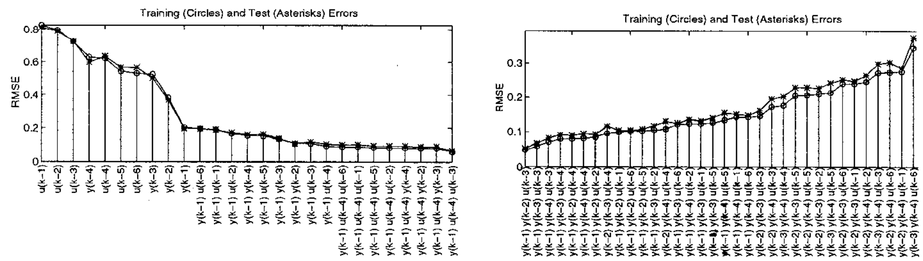
ıto two disjoint sets:

$$\begin{cases} Y = \{y(k-1), y(k-2), y(k-3), y(k-4)\}, \\ U = \{u(k-1), u(k-2), u(k-3), u(k-4), u(k-5), u(k-6)\}. \end{cases}$$

. heuristic approach to input selection is to treat all the input candidates qually and select the best input arguments sequentially based on a greedy olicy. That is, first, we construct 10 ANFIS model with single input, and elect the one with the smallest training error. Then based on the selected ıodel, we can add another different input and choose the best one from 9 .NFIS models with two inputs, and so on. For simplicity, we assume that the .NFIS models have grid partitioning and each input has two generalized bell ıembership functions defined by

$$\mu_{A_i}(x) = \left[1 + \left|\frac{x - c_i}{a_i}\right|^{2b_i}\right]^{-1}, \tag{1}$$

/here $\{a_i,\, b_i,\, c_i\}$ is the parameter set. Thus the number of fuzzy rules is 2^n, /here n is the number of input arguments. Each ANFIS is trained for a single epoch, which corresponds to a single application of least-squares methods o identify consequent parameters (that specify each rule's output equation). Vhen n is 3, we found that the best ANFIS model can outperform the ARX nodel already. The error curves for the input selection process is shown in 'igure 3(a), where the selected input are listed according to the order of de-



(a) (b)

Figure 3: *ANFIS input selection: (a) exhaustive search; (b) sequential search.*

Table 1: *A table of performance comparisons among the ARX model, the ANFIS models via sequential and exhaustive searches.*

Models	ARX	ANFIS (Seq. Srch.)	ANFIS (Exh. Srch.)
No. of Input Arguments	14	3	3
RMSE for Training Set	0.1142	0.0609	0.0474
RMSE for Test Set	0.0720	0.0670	0.0524
No. of Linear Parameters	15	32	32
No. of Nonlinear Parameters	0	18	18
Total Computation Time	7 Sec.	23 Sec.	48 Sec.

creasing training errors. Other information of the selected ANFIS model is shown in the third column of Table 1.

Another more computation intensive approach is to do an exhaustive search on all possible (and reasonable) combinations of the input candidates[6]. From the heuristic search described earlier, it seems that two input arguments are too few for ANFIS to get all necessary information. Moreover, since we are modeling a dynamical system, it is reasonable to take two inputs from Y and one from U to make a total of three inputs to the ANFIS model. This results in 36 ANFIS models, each with 8 fuzzy rules. Figure 3(b) show the performance of these 36 ANFIS models (according to the order of increasing training errors). The best one has a training RMSE of 0.0474 and test RMSE 0.0524. This ANFIS model outperforms the ARX model in terms of data fitting, but it takes much longer in computation.

Table 1 summarizes the comparisons among the ARX model and two ANFIS models obtained via heuristic and exhaustive searches. The ANFIS model has 32 linear parameters while the ARX model has only 15 (including the average level of the data set). However, the number of fitting parameters is not a crucial factor in determining the models' performance, since we did have an ARX model with 40 parameters with worse performance. Instead, the model structure is more important — a nonlinear model is required if the system to be identified is nonlinear. In the same vein, the number of input arguments is not really a crucial factor to a model's performance.

Although we list 18 nonlinear parameters under the ANFIS models in Table 1, they are not used at all. If we continue the ANFIS training with the hybrid learning method, then these 18 nonlinear parameters are utilized to refine the ANFIS model. Figure 2(b) demonstrates the performance of ANFIS (based on the one obtained via exhaustive search) after 100 epochs of training, with training RMSE 0.04645 and test RMSE 0.04795. (Note that these error measures are obtained obtained at the minimal test error, which occurs at epoch 30.)

5 Concluding Remarks

We have compared the performance of hair dryer modeling using three approaches: the ARX model, ANFIS via sequential search, and ANFIS via exhaustive search. It is obvious that the conventional ARX model can achieve a good performance in the shortest amount of time. However, to model nonlinearity that inherently come with a real-world dynamic system, a nonlinear neuro-fuzzy modeling technique such as ANFIS can enhance the performance at the expense of more computation time.

References

1. L. Ljung. *System identification: theory for the user.* Prentice Hall, Upper Saddle River, NJ, 1987.
2. T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15:116–132, 1985.
3. J.-S. Roger Jang, C.-T. Sun, and E. Mizutani. Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence, 1996. To be published by Prentice Hall.
4. J.-S. Roger Jang. ANFIS: Adaptive-Network-based Fuzzy Inference Systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(03):665–685, May 1993.
5. P. Werbos. *Beyond regression: New tools for prediction and analysis in the behavioral sciences.* PhD thesis, Harvard University, 1974.
6. J.-S. Roger Jang. Input selection for ANFIS learning. In *Proceedings of IEEE International Conference on Fuzzy Systems*, New Orleans, September 1996.