A Hybrid Neuro-fuzzy Approach for Spinal Force Evaluation in Manual Materials Handling Tasks

Yanfeng Hou[1], Jacek M. Zurada[1],
Waldemar Karwowski[2], and William S. Marras[3]

[1] Department of Electrical and Computer Engineering, University of Louisville,
y0hou002@louisville.edu
[2] Department of Industrial Engineering,University of Louisville,
karwowski@louisville.edu
[3] Biodynamics Laboratory, Institute for Ergonomics,The Ohio State University,
marras.1@osu.edu

Abstract. Evaluation of the spinal forces from kinematics data is very complicated because it involves the handling of relationship between kinematic variables and electromyography (EMG) responses, as well as the relationship between EMG responses and the forces. A recurrent fuzzy neural network (RFNN) model is proposed to establish the kinematics-EMG-force relationship and model the dynamics of muscular activities. The EMG signals are used as an intermediate output and are fed back to the input layer. Since the EMG signal is a direct reflection of muscular activities, the feedback of this model has a physical meaning. It expresses the dynamics of muscular activities in a straightforward way and takes advantage from the recurrent property. The trained model can then have the forces predicted directly from kinematic variables while bypassing the procedure of measuring EMG signals and avoiding the use of biomechanics model. A learning algorithm is derived for the RFNN.

1 Introduction

The loads on the lumbar spine during manual lifting are very useful in judging if such a task is risky. Studying the forces applied to the lumbar spine is fundamental to the understanding of low back injury [1]. Biomechanical models are often used to obtain the forces applied to the lumbar spine from the measured electromyographic responses of trunk muscles during the lifting motions. The EMG signals are measured because they directly reflect the muscular activities [2]. However, the measuring of EMG signals is costly and the use of biomechanical models is time consuming.

EMG signals are also related to the kinematic characteristics in the motion. The kinematic variables (with other auxiliary variables) can be used to evaluate the EMG signals generated in the muscles during the motion [8][9]. Thus we may be able to connect the spinal forces with kinematic variables through EMG

L. Wang, K. Chen, and Y.S. Ong (Eds.): ICNC 2005, LNCS 3612, pp. 1216–1225, 2005.

signals. We want to develop a model that can express the kinematics-EMG-force relationship and predict forces on lumbar spine without the procedure of measuring EMG signals and the use of biomechanics model.

To Evaluate the dynamic forces on lumbar spine we build a recurrent fuzzy neural network model. There are several ways to provide feedback connections. In [13] and [14], the output of each membership function is fed back to itself to achieve the recurrent property. However, the fuzzy rules obtained from the model can not offer a clear understanding to the system. In the premise of the rules, the inputs are combined with the feedback of the outputs of their own membership functions. The rules become hard to understand and not meaningful in explaining the behavior of the system. The only function of the feedback is to add a memory element to the model.

In [15] and [16], the output of all rule nodes, the firing strength, is fed back. It serves as an internal variable. The rules generated by the model have a form like:

IF *the external variables* (at t) are A and *the internal variables* (at t) are B, THEN *the outputs* (at $t+1$) are C and the *internal variables* (at $t+1$) are D.

A, B, C, D are fuzzy sets in the above rule.

Although the internal variables play a role in the fuzzy rules and contribute to the model, it is not useful to us in understanding the system under consideration. What we attempt to know is the relationship between the input and output of the system.

In [17] and [18], the final output of the network is fed back to the input layer. In [17], the feedback is multiplied with the external inputs of the model. Thus, the inputs of the first layer becomes:

$$net_i^1 = \prod_o x_i^1 \cdot w_{oi} \cdot y_o^4(t-1) \tag{1}$$

where x_i^1 is the external input; w_{oi} are the weights of the feedback connections; $y_o^4(t-1)$ is the output of the model at $t-1$; o is the number of outputs. As we can see, the rules obtained from the model also lose their clear physical meaning. In [18], the feedback of the outputs is not combined with other signals. It is fed to the input layer as regular input variables. However, the membership functions used for the feedback connections are of this form:

$$\mu = exp(-(w \cdot y_o^4(t-1))^2), \tag{2}$$

where w denotes the weights of the feedback connections. Formula (2) is actually a Gaussian membership function centered at zero with one adjustable parameter of width. The advantages of doing so are that the network has less parameters and the update rules for the tuning parameters are easier to calculate. However, setting all the feedback membership functions' centers as a fixed value of zero may decrease the effectiveness of the feedback variables.

In our model, we use the EMG signals as an intermediate output and feed them back to the input layer. By doing that, more information (EMG) was provided to the model and the feedback of the intermediate output has a physical meaning (the direct relationship of EMG-force). This reflects the dynamics of the system in a clear and straightforward way. At the same time, the advantages of recurrent property is utilized. The rules generated from the model can be easily interpreted and can help us understand the muscular activities better.

2 Model Construction

We come up with a recurrent fuzzy neural network model which takes the kinematics data and EMG data at time t and evaluates the spinal forces and EMG signals at time $t+1$. The EMG signals of ten trunk muscles are scaled and delayed before they are fed back to the input layer. The delay of EMG is used to represent the muscular activation dynamic properties. The interaction between muscles influences the EMG and the forces on the spine. By presenting the previous EMG to the input, we hope the modle can take such interaction into account. The proposed system structure is shown in Figure 1. As we can see

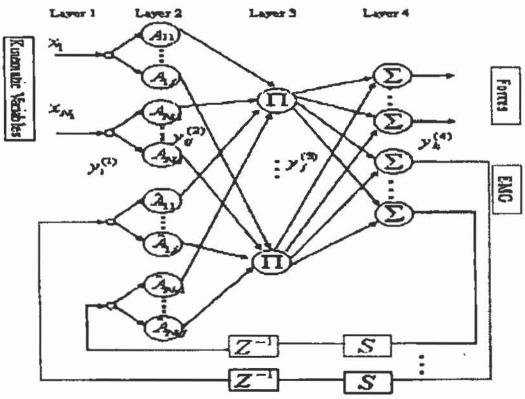

Fig. 1. The proposed recurrent fuzzy neural network structure (Z^{-1} is a unit delay operator and S is a scale operator)

in Figure 1, the direct physical relationships (kinematics-EMG and EMG-force) reside in the model. Three forces on the lumbar spine and ten EMG signals of trunk muscles are the model outputs. Twelve kinematic variables and ten EMG feedback signals are the model inputs.

The function of each layer in Figure 1 is described as follows:

Layer 1 is the input layer. It includes two parts. One is the kinematic variables and the other one is the feedback of EMG signals. They are passed to the second layer.

For external inputs,

$$y_i^{(1)} = x_i, \tag{3}$$

$i = 1, 2, ..., N_1$, where $N1$ stands for the twelve kinematic variables.

For the internal (feedback) inputs,

$$y_i^{(1)} = y_k^{(4)}(t-1) \tag{4}$$

$y_k^{(4)}(t-1)$ is the kth output of layer 4 at time $(t-1)$, denoting the EMG feedback. $i = N_1 + 1, N_1 + 2, ..., N$, and $N = N_1 + N_2$, where N_2 stands for the number of EMG feedback signals.

Layer 2 is the input fuzzification layer, which represents linguistic sets in antecedent fuzzy membership functions. Each neuron describes a membership function and encodes the center and width of membership functions. The output of this layer is the degree of membership of each input:

For external inputs, the following Gaussian membership function is used:

$$y_{ij}^{(2)} = exp(-(\frac{y_i^{(1)} - m_{ij}}{\sigma_{ij}})^2) \tag{5}$$

$i = 1, 2, ..., N_1$, $j = 1, 2, ..., M$, where M is the number of rules.

For the internal inputs, the following sigmoid membership function is used:

$$y_{ij}^{(2)} = exp(-(\frac{y_i^{(1)} - \hat{m}_{ij}}{\hat{\sigma}_{ij}})^2) \tag{6}$$

$i = N_1 + 1, N_1 + 2, ..., N$ and $j = 1, 2, ..., M$.

Layer 3 computes the firing strength. Nodes in this layer perform the product operation. The links establish the antecedent relation with an "AND" operation for each fuzzy set combination (both the external input and the feedback). The output of this layer is the firing strength of each fuzzy rule:

$$y_j^{(3)} = \prod_{i=1}^{M} y_{ij}^{(2)} = \prod_{i=1}^{N_1} exp(-(\frac{y_i^{(1)} - m_{ij}}{\sigma_{ij}})^2) \prod_{i=N_1+1}^{N} exp(-(\frac{y_i^{(1)} - \hat{m}_{ij}}{\hat{\sigma}_{ij}})^2) \tag{7}$$

where $j = 1, 2, ..., M$.

Layer 4 is the defuzzification layer. The output of this layer is the overall output:

$$y_k^{(4)} = \sum_{j=1}^{M} W_{jk} y_j^{(3)} (\sum_{j=1}^{M} y_j^{(3)})^{-1} \tag{8}$$

$k = 1, 2, ..., K$, where K is the number of outputs.

This is a fuzzy system model with learning capabilities. It uses a singleton to represent the output fuzzy set of each fuzzy rule. The product operator instead of minimum operator is used for the calculation of the firing strength because the calculation of the partial derivatives is easier for the product operator.

The rules generated for the above model are in such form:
the jth rule:

IF $Kine_1(t)$ is μ_{1j} and ... and $Kine_{(N1)}(t)$ is $\mu_{(N1)j}$
and $EMG_1(t)$ is $\hat{\mu}_{1j}$ and ... and $EMG_{(N2)}(t)$ is $\hat{\mu}_{(N2)j}$
THEN $Force_1(t+1)$ is O_{1j} and ... and $Force_{K1}(t+1)$ is $O_{(K1)j}$
and $EMG_1(t+1)$ is Y_{1j} and ... and $EMG_{(N2)}(t+1)$ is $Y_{(N2)j}$

where μ_{ij} $(i = 1,2,...,N1; j = 1,2,...,M)$ are fuzzy sets of $Kine_i$ (the ith kinematic variable). $\hat{\mu}_{ij}$ $(i = 1,2,...,N2; j = 1,2,...,M)$ are fuzzy sets of EMG_i. O_{kj} $(k = 1,2,...,K1)$ are the output singletons for forces. Y_{kj} $(k = 1,2,...,N2)$ are the output singletons for EMG signals.

The forces predicted for time $t+1$ depend on not only the inputs at time t, but also the predicted EMG at time t, which again depend on the previous inputs. This is a dynamic approach that can represent the dynamic properties of the forces better than a feedforward network.

The above rules represent the relationships between kinematic variables, EMG signals and forces. They can be decomposed into three subsets of fuzzy rules as follows.

The Kinematics-EMG relationship:
IF $Kine_1$ is μ_{1j} and ... and $Kine_{(N1)}$ is $\mu_{(N1)j}$
THEN EMG_1 is Y_{1j} and ... and $EMG_{(N2)}$ is $Y_{(N2)j}$

The EMG-Force relationship:
IF EMG_1 is $\hat{\mu}_{1j}$ and ... and $EMG_{(N2)}$ is $\hat{\mu}_{(N2)j}$
THEN $Force_1$ is O_{1j} and ... and $Force_{K1}$ is $O_{(K1)j}$

The Kinematics-Force relationship:
IF $Kine_1$ is μ_{1j} and ... and $Kine_{(N1)}$ is $\mu_{(N1)j}$
THEN $Force_1$ is O_{1j} and ... and $Force_{K1}$ is $O_{(K1)j}$

These Kinematics-EMG-Force relationships are knowledge we would like to find out.

2.1 Structure Adaptation and Parameter Tuning

During the training process, both the premise and the consequence parameters are tuned simultaneously. This approach involves two phases, structure adaptation and parameter tuning. The fuzzy rules are created and tuned based on the training data.

At first, the rule base contains only one rule defined by the first input-output data pair. Then Additional rules are created during the training process using other input-output pairs. When the new training pattern does not excite any of the existing fuzzy rules, a new fuzzy rule should be created. If the firing strength $S_u > \beta$, then the rule base is unchanged and perform the gradient training to match the new sample pair. If the firing strength $S_u < \beta$, then a new rule is created. β is a threshold defined as the least acceptable degree of excitation of the existing rule base. It is important that this predefined threshold should decay during the learning process. Otherwise new rules may continually be added to the model.

The free parameters (the membership functions of the external variables, the membership functions of the internal variables, and the weights of the consequence singleton) in the fuzzy inference mechanism are then tuned after new rules are created. Parameter tuning is carried out simultaneously with the structure adaptation. The ordered derivative [19] is used to derive the learning algorithm.

The error function to be minimized is

$$E(t+1) = \frac{1}{2}\sum_{k=1}^{K}\varepsilon(t+1)^2 = \frac{1}{2}\sum_{k=1}^{K}(d_k(t+1) - y_k^{(4)}(t+1))^2 \tag{9}$$

where $d_k(t+1)$ is the target and $y_k^{(4)}(t+1)$ is the output of the model (the output of layer 4).

The update rule for the output singleton w_{kj} (the weights of the connections between layer 3 and layer 4) is

$$w_{kj}(t+1) = w_{kj}(t) - \eta\frac{\partial E(t+1)}{\partial w_{kj}} \tag{10}$$

where

$$\frac{\partial E(t+1)}{\partial w_{kj}} = \frac{\partial E(t+1)}{\partial y_k^{(4)}}\frac{\partial y_k^{(4)}}{\partial w_{kj}} = \varepsilon(t+1)\frac{y_j^{(3)}}{\sum\limits_{j=1}^{M} y_j^{(3)}} \tag{11}$$

The centers of the membership functions of external variables are m_{ij}. The update rule is

$$m_{ij}(t+1) = m_{ij}(t) - \eta\frac{\partial E(t+1)}{\partial m_{ij}} \tag{12}$$

where

$$\frac{\partial E(t+1)}{\partial m_{ij}} = \frac{\partial E(t+1)}{\partial y_j^{(3)}}\frac{\partial y_j^{(3)}}{\partial m_{ij}} = \sum_{k=1}^{K}\varepsilon(t+1)\cdot D\cdot\frac{\partial y_j^{(3)}}{\partial m_{ij}} \tag{13}$$

in which D is defined as follows for notation simplicity

$$D = \frac{(w_{kj} - y_k^{(4)}(t+1))}{\sum\limits_{j=1}^{M} y_j^{(3)}} \tag{14}$$

From formula 7 we get

$$y_j^{(3)} = exp\Big(-\sum_{i=1}^{N_1}\frac{(y_i^{(1)}(t) - m_{ij})^2}{\sigma_{ij}^2} - \sum_{i=N_1+1}^{N}\frac{(y_i^{(4)}(t) - \hat{m}_{ij})^2}{\hat{\sigma}_{ij}^2}\Big) \tag{15}$$

in which $y_i^{(4)}(t)$ again depends on m_{ij}.

Then the derivative can be written as

$$\frac{\partial y_j^{(3)}}{\partial m_{ij}} = y_j^{(3)}\Big(A_1 - \sum_{i=N_1+1}^{N} B \cdot \frac{\partial y_j^{(4)}(t)}{\partial m_{ij}}\Big) \tag{16}$$

where A_1 and B are defined as

$$A_1 = \frac{2(y_i^{(1)}(t) - m_{ij})}{\sigma_{ij}^2} \tag{17}$$

$$B = \frac{2(y_i^{(4)}(t) - \hat{m}_{ij})}{\hat{\sigma}_{ij}^2} \tag{18}$$

Finally a recursive function is obtained for $\frac{\partial y_j^{(4)}}{\partial m_{ij}}$.

$$\frac{\partial y_j^{(4)}(t)}{\partial m_{ij}} = D \cdot y_j^{(3)} \cdot \Big(A_1(t-1) - \sum_{i=N_1+1}^{N} B(t-1) \cdot \frac{\partial y_j^{(4)}(t-1)}{\partial m_{ij}}\Big) \tag{19}$$

The update rules for other parameters (σ_{ij}, $\hat{m}_{ij}$, $\hat{\sigma}_{ij}$) are omitted here. The initial values of $\frac{\partial y_j^{(4)}(t)}{\partial m_{ij}}$, $\frac{\partial y_j^{(4)}(t)}{\partial \sigma_{ij}}$, $\frac{\partial y_j^{(4)}(t)}{\partial \hat{m}_{ij}}$ and $\frac{\partial y_j^{(4)}(t)}{\partial \hat{\sigma}_{ij}}$ are set to zero.

All the parameters are tuned during the training process when new data pairs are presented to the network.

3 Simulations and Results

This section shows the results and the performance of the proposed model. We evaluated the performance of the proposed recurrent fuzzy neural network with two kinds of data. One is the sagittal symmetric motions, while the other one is unsymmetrical motions. To make the results comparable, similar task variables are selected for these two motions. Both motions are done with two hands and controlled placement. The lift frequency is 2 lifts/min; the weight of object is 25 lbs; the origin height is 60 cm; the origin distance is 45; the destination height is 105 cm; and the destination distance is 55 cm.

For the sagittal symmetric motions, 720 training patterns are used. The learning rate of the parameters of feedback connections ($\hat{m}_{ij}$ and $\hat{\sigma}_{ij}$) is $\hat{\eta} = 0.02$. The learning rate for other parameters (m_{ij}, σ_{ij} and w_{kj}) is $\eta = 0.01$. The initial threshold β for firing strength is set as 0.2.

As stated above, the learning rates for the parameters of external inputs (kinematic variables) and for the parameters of internal inputs (EMG feedback) are different. Since the initial values of parameters of internal inputs are small random values while the initial values of parameters of external inputs are good values with physical meaning, the convergence of the latter is faster than the convergence of the former. Figure 2 was obtained after 200 epoches. In this figure, both the forces and the EMG signals are predicted well, which means the

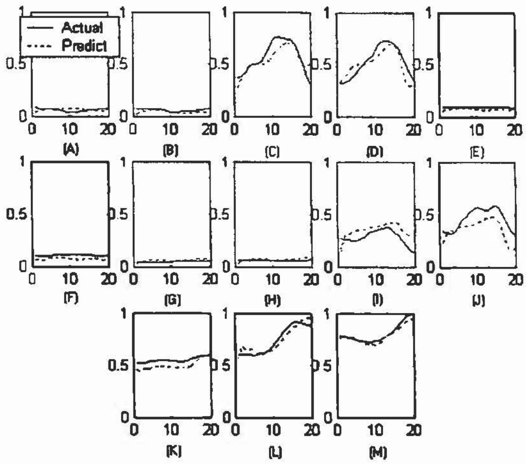

Fig. 2. Output after 200 training epoches (the first 10 are EMG signals, the last three are forces). (A) RLD, (B) LLD, (C) RES, (D) LES, (E) RRA, (F) LRA, (G) REO, (H) LEO, (I) RIO, (J) LIO, (K) Lateral shear force, (L) A-P shear force, and (M) spinal compression

parameters of both the external inputs and the feedback are well trained after 200 epoches.

The rules obtained are of the following form:

IF $Kine_1(t)$ is $\mu(0.443, 0.832)$ and $Kine_2(t)$ is $\mu(0.521, 1.334)$ and $Kine_3(t)$ is $\mu(0.714, 1.587)$ and $Kine_4(t)$ is $\mu(-1.654, 1.583)$ and $Kine_5(t)$ is $\mu(0.476, 1.011)$ and $Kine_6(t)$ is $\mu(-0.803, 1.486)$ and $Kine_7(t)$ is $\mu(-1.770, 2.118)$ and $Kine_8(t)$ is $\mu(0.746, 1.342)$ and $Kine_9(t)$ is $\mu(0.833, 1.535)$ and $Kine_{10}(t)$ is $\mu(0.493, 1.566)$ and $Kine_{11}(t)$ is $\mu(-0.017, 1.833)$ and $Kine_{12}(t)$ is $\mu(-0.387, 1.322)$

and $EMG_1(t)$ is $\mu(0.025, 1.258)$ and $EMG_2(t)$ is $\mu(0.025, 1.259)$ and $EMG_3(t)$ is $\mu(0.006, 0.992)$ and $EMG_4(t)$ is $\mu(0.005, 0.074)$ and $EMG_5(t)$ is $\mu(0.023, 1.249)$ and $EMG_6(t)$ is $\mu(0.025, 1.259)$ and $EMG_7(t)$ is $\mu(0.029, 1.263)$ and $EMG_8(t)$ is $\mu(0.019, 1.266)$ and $EMG_9(t)$ is $\mu(0.009, 0.805)$ and $EMG_{10}(t)$ is $\mu(0.104, 1.246)$

THEN $Force_1(t+1)$ is 0.443 and $Force_2(t+1)$ is 0.559 and $Force_3(t+1)$ is 0.758

and $EMG_1(t+1)$ is 0.033 and $EMG_2(t+1)$ is 0.120 and $EMG_3(t+1)$ is 0.267 and $EMG_4(t+1)$ is 0.318 and $EMG_5(t+1)$ is 0.055 and $EMG_6(t+1)$ is 0.057 and $EMG_7(t+1)$ is 0.061 and $EMG_8(t+1)$ is 0.044 and $EMG_9(t+1)$ is 0.218 and $EMG_{10}(t+1)$ is 0.102

We also can decompose the above fuzzy rule into three subsets as we did previously to understand the system better.

4 Conclusions

A spinal force prediction model was developed using a recurrent fuzzy neural network. The EMG feedback represents the muscular activation dynamics better. At the same time, it brings more information to the model and utilizes the advantages of recurrent properties. The model predicts forces directly from kinematics data, avoiding EMG measurements and the use of biomechanics model. EMG signals are obtained as byproduct. It can help us understand the relationships between kinematic variables and EMG signals and spinal forces. An adaptive learning algorithm is derived for the recurrent fuzzy neural network.

Acknowledgment

This study was conducted under a research grant on the "Development of a Neuro-Fuzzy System to Predict Spinal Loading as a Function of Multiple Dimensions of Risk", sponsored by the National Institute for Occupational Safety and Health (DHHS).

References

1. D.G. Lloyd, Besier, T.F., "An EMG-driven Musculoskeletal Model to Estimate Muscle Forces and Knee Joint Moments in Vivo", *Journal of Biomechanics*, 36 (2003), pp. 765–776.
2. P.A. Crosby, "Use of surface electromyogram as a measure of dynamic force in human limb muscles", *Med. and Biol. Eng. and Comput.*, vol. 16, 1978, pp. 519–524.
3. L. Wang, T.S. Buchanan, "Prediction of Joint Moments Using a Neural Network Model of Muscle Activations from EMG Signals", *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, vol. 10, no. 1, Mar. 2002, pp. 30–37
4. J.J. Luh, G.C. Chang, C.K. Cheng, J.S. Lai, T.S. Kuo,, "Isokinetic elbow joint torques estimation form surface EMG and joint kinematic data: Using an artificial neural network model", *J. Electromyogr. Kinesiol.*, vol. 1, no. 9, 1999, pp. 173-183.
5. M.M. Liu, W. Herzog, H. H. Savelberg, "Dynamic muscle force predictions from EMG: An artificial neural network approach", *J. Electromyogr. Kinesiol.*, vol. 9, 1999, pp. 391-400.
6. S.E. Hussein, M.H., Granat, "Intention detection using a neuro-fuzzy EMG classifier", *Engineering in Medicine and Biology Magazine, IEEE*, vol. 21, no. 6, Nov.-Dec. pp. 123–129
7. K. Kiguchi, T. Tanaka, T. Fukuda, "Neuro-fuzzy control of a robotic exoskeleton with EMG signals", *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, Aug. 2004 pp. 481–490
8. Y. Hou, J.M. Zurada, W. Karwowski, "Prediction of EMG Signals of Trunk Muscles in Manual Lifting Using a Neural Network Model", *Proc. of the Int. Joint Conf. on Neural Networks*, Jul. 25-29, 2004, pp. 1935–1940
9. Y. Hou, J.M. Zurada, W. Karwowski, "Prediction of Dynamic Forces on Lumbar Joint Using a Recurrent Neural Network Model", *Proc. of the 2004 Int. Conf. on Machine Learning and Applications (ICMLA'04)*, Dec. 16-18, 2004, pp. 360-365

10. S. Wu and M. J. Er, "Dynamic fuzzy neural networks-a novel approach to function approximation," *IEEE Trans. on Systems, Man and Cybernetics* B, vol. 30, Apr. 2000, pp. 358-364.
11. S. Wu, M. J. Er, and Y. Gao, "A fast approach for automatic generation of fuzzy rules by generalized dynamic fuzzy neural networks," *IEEE Trans. on Fuzzy Systems*, vol. 9, Aug. 2001, pp. 578-594.
12. C.F. Juang and C.T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications", *IEEE Trans. on Fuzzy Systems*, vol. 6, pp. 12-32, Feb. 1998.
13. C.H. Lee and C.C. Teng, "Identification and control of dynamic systems using recurrent fuzzy neural networks", *IEEE Trans. on Fuzzy Systems*, vol. 8 , no. 4 , Aug. 2000 pp. 349–366.
14. C.M. Lin and C.F. Hsu, "Supervisory recurrent fuzzy neural network control of wing rock for slender delta wings", *IEEE Trans. on Fuzzy Systems*, vol. 12 , no. 5 , Oct. 2004 pp. 733–742.
15. C.F. Juang, "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms", *IEEE Trans. on Fuzzy Systems*, vol. 10 , no. 2 , Apr. 2002 pp. 155–170
16. F.J. Lin, R.J. Wai, "Hybrid control using recurrent fuzzy neural network for linear-induction motor servo drive", *IEEE Trans. on Fuzzy Systems*, vol. 9 , no. 1 , Jan. 2001 pp. 68–90.
17. F.J. Lin, R.J. Wai, C.M. Hong, "Hybrid supervisory control using recurrent fuzzy neural network for tracking periodic inputs", *IEEE Trans. on Neural Networks*, vol. 12 , no. 1 , Feb. 2001 pp. 102–115.
18. Y.C Wang, D. Zipser, "A learning algorithm for continually running recurrent neural networks", *Neural Comput.*, vol. 1 , no. 2 , 1989 pp. 270–280.
19. R.J. Williams, C.J Chien, C.C. Teng, "Direct adaptive iterative learning control of nonlinear systems using an output-recurrent fuzzy neural network", *IEEE Trans. on Systems, Man, and Sybernetics*, vol. 34 , no. 3 , Jun. 2004 pp. 1348–1359.
20. Y. Hou, J.M. Zurada, W. Karwowski, "Identification of Input Variables using Fuzzy Average with Fuzzy Cluster Distribution"; submitted to *IEEE Trans. on Fuzzy Systems.*
21. S. Auephanwiriyakul, J.M. Keller, "Analysis and efficient implementation of a linguistic fuzzy c-means"; *IEEE Trans. on Fuzzy Systems*, vol. 10 , no. 5 , Oct. 2002 pp. 563 - 582
22. W. Lee, W. Karwowski, W.S. Marras, D. Rodrick, "A neuro-fuzzy model for estimating electromyographical activity of trunk muscles due to manual lifting", *Ergonomics* 46 (1-3), JAN 15 2003, pp. 285-309