# Estimation of the Dynamic Spinal Forces Using a Recurrent Fuzzy Neural Network

Yanfeng Hou, *Student Member, IEEE*, Jacek M. Zurada, *Fellow, IEEE*, Waldemar Karwowski, William S. Marras, and Kermit Davis

***Abstract*—Estimation of the dynamic spinal forces from kinematics data is very complicated because it involves the handling of the relationship between kinematic variables and electromyography (EMG) signals, as well as the relationship between EMG signals 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 EMG 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 costly procedure of measuring EMG signals and avoiding the use of a biomechanics model. A learning algorithm is derived for the RFNN model.**

***Index Terms*—Fuzzy neural networks, spinal force.**

## I. Introduction

THE MUSCULAR activities in manual materials-handling tasks are complex and dynamic. 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 lifting motions, as shown in Fig. 1(a). Electromyography (EMG) signals are measured because they directly reflect muscular activities [2]. However, the measuring of EMG signals and the use of biomechanical models are costly and time consuming.

Fuzzy systems and neural network models have been used to replace the biomechanical model in Fig. 1(a). In [3], Wang and Buchanan predicted the muscular activations from EMG signals using a four-layer feedforward neural network model trained by a backpropagation learning algorithm. Luh *et al.* built a neural network to model the relationship between EMG activity and elbow joint torque [4]. Liu *et al.* used a neural network to predict dynamic muscle forces from EMG signals [5]. In [6] and [7], neurofuzzy models were developed for EMG signal classification and prosthesis control. These findings focus on building the relationship between the EMG signals of the muscles and the forces on the joint. They all require the EMG signals to be measured, which is often difficult to perform in industrial environments.

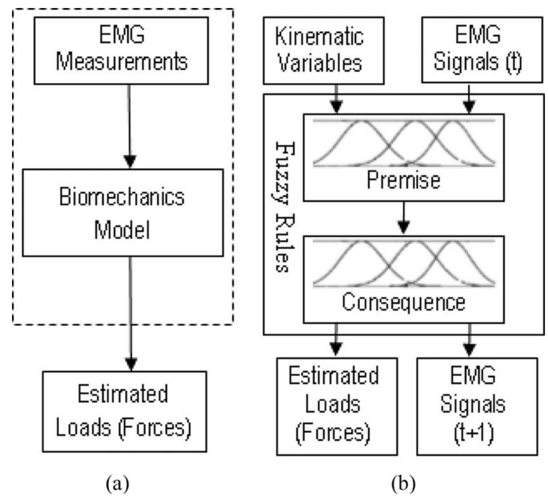


Fig. 1. Load (forces on spine) prediction systems. (a) Conventional EMG-driven load prediction system. (b) RFNN-based direct load prediction system.

EMG signals are also related to kinematic characteristics in motion. The kinematic variables (with other auxiliary variables) can be used to estimate the EMG signals generated in the muscles during motion [8], [9]. Thus, we may be able to connect the spinal forces with kinematic variables through EMG signals. We want to develop a model that can express the kinematics–EMG–force relationship and predict forces on the lumbar spine without the procedure of measuring EMG signals and the use of a biomechanics model. Since the information obtained for the evaluation of body stresses is normally uncertain, imprecise, and noisy, and the input–output relationship between the multiple variables is not clear in many situations, neural networks and fuzzy logic methods are used here. By using the fuzzy neural approach, we can avoid establishing a complex mathematical model to express the muscle activation dynamics. The adaptive fuzzy neural inference system [10]–[12] is a hybrid method that combines the advantages of the neural network and fuzzy logic approach. When feedback connections are added, it becomes a recurrent fuzzy neural network

Manuscript received June 1, 2005; revised November 18, 2005 and March 28, 2006. This work was supported by the National Institute for Occupational Safety and Health (DHHS) under a Research Grant on the "Development of a Neuro-Fuzzy System to Predict Spinal Loading as a Function of Multiple Dimensions of Risk." This paper was recommended by Associate Editor D. D. Nauck.

Y. Hou and J. M. Zurada are with the Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY 40292 USA (e-mail: y0hou002@louisville.edu; jacek.zurada@louisville.edu).

W. Karwowski is with the Department of Industrial Engineering, University of Louisville, Louisville, KY 40292 USA (e-mail: karwowski@louisville.edu).

W. S. Marras is with the Institute for Ergonomics, The Ohio State University, Columbus, OH 43210 USA (e-mail: marras.1@osu.edu).

K. Davis is with the Department of Environmental Health, University of Cincinnati, Cincinnati, OH 45267 USA (e-mail: kermit.davis@uc.edu).

Digital Object Identifier 10.1109/TSMCB.2006.881298

(RFNN). The feedback makes it possible to take past information into account. The output of the model is computed by the current data as well as the preceding data. The time delay is incorporated in the feedback connections. It serves to preserve the past information, so that the RFNN is able to handle the dynamics. It expands the basic ability of the fuzzy neural network to include temporal problems [13]. To establish the kinematics–EMG–force relationship and estimate the dynamic forces on the lumbar spine, we build an RFNN model. By providing EMG feedback to the model, a straightforward way to express the kinematics–EMG–force relationship can be obtained. The relationship between the feedback and the output coincides with the physical EMG–force relationship.

### A. Related Work

There are several ways to provide feedback connections. In [14] and [15], the output of each membership function is fed back to itself to achieve the recurrent property. However, the fuzzy rules obtained from the model cannot give us a clear understanding of the system. In the premise of the rules, the inputs are combined with the feedback of the outputs of their own membership functions, i.e.,

IF the external variables (at $t$) are $A_i$ and the outputs of membership functions (at $t$) are $B_j$, THEN the outputs (at $t+1$) are $C_k$ and the outputs of membership functions (at $t+1$) are $D_j$.

$A_i$, $B_j$, $C_k$, and $D_j$ are fuzzy sets in the above rule.

The rules become difficult to understand and are not so meaningful to us in explaining the behavior of the system. The only function of the feedback is to add a memory element to the model.

In [16] and [17], 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 the following form:

IF the external variables (at $t$) are $A_i$ and the firing strengths (at $t$) are $B_j$, THEN the outputs (at $t+1$) are $C_k$ and the firing strengths (at $t+1$) are $D_j$.

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. The firing strength (the internal variable) is not what we care about. What we want to know is the relationship between the input and output of the system.

In [18] and [19], the final output of the network is fed back to the input layer. In [18], the feedback is multiplied with the external inputs of the model. Thus, the inputs of the first layer become

$$\text{net}_I^1 = \prod_o x_I^1 \cdot w_{oi} \cdot y_o^4(t-1) \quad (1)$$

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 $t-1$, and $o$ is the number of outputs. The rules obtained from the model are as follows:

IF the products of external variables and the feedback (at $t$) are $A_i$, THEN the outputs (at $t+1$) are $C_k$ and the feedback (at $t+1$) are $D_j$.

As we can see, the rules also lose their clear physical meaning.

In [19], the feedback of the outputs is not combined with other signals. They are fed to the input layer as regular input variables. However, the membership functions used for the feedback connections are of the following form:

$$\mu = \exp\left(-\left(w \cdot y_o^4(t-1)\right)^2\right) \quad (2)$$

where $w$ denotes the weights of the feedback connections. Formula (2) is in fact a Gaussian membership function centered at zero with one adjustable parameter of width. The advantages of doing so are that the network has less parameter 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 to obtain the kinematics–EMG–force relationship [Fig. 1(b)]. By doing that, 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 are utilized. The rules generated from the model can be easily interpreted and can help us understand the muscular activities better. Measured EMG signals are only required at the training stage. After training, EMG signals will be the feedback from output of the model.

## II. Model Construction

We come up with an RFNN model that takes the kinematics data and EMG data at time $t$ and estimates the spinal forces and EMG signals at time $t+1$. The EMG signals of six trunk muscles are scaled and delayed before they are fed back to the input layer. The time delay (from time $t$ to time $t+1$) is about 1 ms. It is decided by the time difference between two sampling data points. The previous data point of EMGs is fed back because the forces are directly affected by them. Earlier EMGs could also be added to the feedback, but it will make the model too complex and rules difficult to interpret. 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 model can take such interaction into account. The structure of the proposed model is shown in Fig. 2.

As we can see in Fig. 2, the direct physical relationships (kinematics–EMG and EMG–force) reside in the model. Three forces on the lumbar spine and six EMG signals of trunk

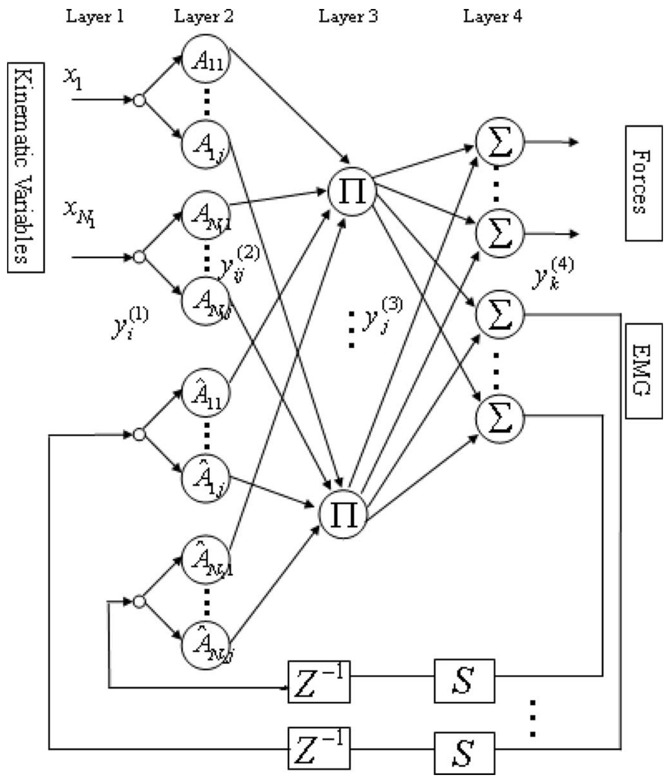


Fig. 2. Proposed RFNN structure. $Z^{-1}$ is a unit delay operator, and $S$ is a scale operator.

muscles are the model outputs. Twelve kinematic variables and six EMG feedback signals are the model inputs.

The kinematic variables includes the sagittal trunk moment, lateral trunk moment, axis trunk moment, sagittal trunk angle, lateral trunk angle, axis trunk angle, sagittal trunk velocity, lateral trunk velocity, axis trunk velocity, sagittal trunk acceleration, lateral trunk acceleration, and axis trunk acceleration.

The six trunk muscles are the right latissimus dorsi (RLD), left latissimus dorsi (LLD), right erector spine (RES), left erector spine (LES), right internal oblique (RIO), and left internal oblique (LIO). The EMG signals of these muscles are normalized EMG magnitude. They range from zero to one. To make them have the same range as the external input variables (kinematic variables), they are scaled before being fed back to the input layer.

The three spinal forces are the lateral shear force, anterior–posterior (A–P) shear force, and spinal compression. They are not the forces measured from the experiments since they cannot be measured directly. They are in fact the forces obtained from the biomechanics model. After the direct prediction model is built, the biomechanics model will no longer be needed in the future.

The function of each layer in Fig. 2 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

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

$i = 1, 2, \ldots, N_1$, where $N_1$ stands for the 12 kinematic variables.

For the internal (feedback) inputs

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

$y_k^{(4)}(t-1)$ is the $k$th output of layer 4 at time $(t-1)$, which denotes the EMG feedback. $i = N_1+1, N_1+2, \ldots, N$, 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 functions are used:

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

$m_{ij}$ and $\sigma_{ij}$ are centers and widths of the membership functions, respectively. $i = 1, 2, \ldots, N_1$, and $j = 1, 2, \ldots, M$, where $M$ is the number of rules.

For the internal inputs, the following membership functions are used:

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

$\hat{m}_{ij}$ and $\hat{\sigma}_{ij}$ are centers and widths of the membership functions, respectively. $i = N_1+1, N_1+2, \ldots, N$, and $j = 1, 2, \ldots, M$.

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, i.e.,

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

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

Layer 4 is the defuzzification layer. Each node in this layer is called an output linguistic node and corresponds to one output linguistic variable, i.e.,

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

$w_{kj}$ is the weights of the connections between layers 3 and 4. $k = 1, 2, \ldots, K$, 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 the minimum operator is used for the calculation of the firing strength because the calculation of the partial derivatives is easier for the product operator. In fact, the forces in the output could also be fed back. However, to achieve the direct kinematics–EMG–force relationship, we did not do that.

The rules generated by the above model are in the form of the $j$th rule, i.e.,where $\mu_{ij}$ $(i = 1, 2, \ldots, N1;\ j = 1, 2, \ldots, M)$ are fuzzy sets of $\text{Kine}_i$ (the $i$th kinematic variable). $\hat{\mu}_{ij}$ $(i = 1, 2, \ldots, N2;\ \ j = 1, 2, \ldots, M)$ are fuzzy sets of $\text{EMG}_i$. $O_{kj}$ $(k = 1, 2, \ldots, K1)$ are the output singletons for forces. $Y_{kj}$ $(k = 1, 2, \ldots, N2)$ are the output singletons for EMG signals.

IF $\text{Kine}_1(t)$ is $\mu_{1j}$ and ... and $\text{Kine}_{(N1)}(t)$ is $\mu_{(N1)j}$
and $\text{EMG}_1(t)$ is $\hat{\mu}_{1j}$ and ... and $\text{EMG}_{(N2)}(t)$ is $\hat{\mu}_{(N2)j}$
THEN $\text{Force}_1(t+1)$ is $O_{1j}$ and ... and $\text{Force}_{K1}(t+1)$ is $O_{(K1)j}$
and $\text{EMG}_1(t+1)$ is $Y_{1j}$ and ... and $\text{EMG}_{(N2)}(t+1)$ is $Y_{(N2)j}$

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 fuzzy rules represent the relationships between kinematic variables, EMG signals, and forces. Since they are related variables, the rules can be decomposed into three subsets of fuzzy rules as follows:

The kinematics–EMG relationship
IF $\text{Kine}_1$ is $\mu_{1j}$ and ... and $\text{Kine}_{(N1)}$ is $\mu_{(N1)j}$, THEN $\text{EMG}_1$ is $Y_{1j}$ and ... and $\text{EMG}_{(N2)}$ is $Y_{(N2)j}$
The EMG–force relationship
IF $\text{EMG}_1$ is $\hat{\mu}_{1j}$ and ... and $\text{EMG}_{(N2)}$ is $\hat{\mu}_{(N2)j}$, THEN $\text{Force}_1$ is $O_{1j}$ and ... and $\text{Force}_{K1}$ is $O_{(K1)j}$
The kinematics–force relationship
IF $\text{Kine}_1$ is $\mu_{1j}$ and ... and $\text{Kine}_{(N1)}$ is $\mu_{(N1)j}$, THEN $\text{Force}_1$ is $O_{1j}$ and ... and $\text{Force}_{K1}$ is $O_{(K1)j}$.

These kinematics–EMG–force relationships are knowledge we would like to find out.

### *A. Structure Adaptation and Parameter Tuning*

During the learning process, structure adaptation and parameter tuning are carried out, as defined in [20]. Initial fuzzy partition for the data is not needed since structure of the model and parameters of the rules will be adjusted during the learning process. Similar methods were also used in [13] and [16]. Initially, no rules exist in the model. Rules are created during the learning process using training data pairs. Since a rule corresponds to a cluster in the input space, the firing strength of a rule can be regarded as the degree the incoming pattern belongs to the corresponding cluster. Therefore, the spatial firing strength, i.e.,

$$y_j^{(3)} = \prod_{i=1}^{M} y_{ij}^{(2)} = \prod_{i=1}^{N_1} \exp\left(-\left(\frac{y_i^{(1)} - m_{ij}}{\sigma_{ij}}\right)^2\right) \prod_{i=N_1+1}^{N} \exp\left(-\left(\frac{y_i^{(1)} - \hat{m}_{ij}}{\hat{\sigma}_{ij}}\right)^2\right) \tag{9}$$

is used as the criterion to decide if a new fuzzy rule should be generated. If the firing strength $y_j^{(3)} > \beta$, then the rule base is unchanged, and the gradient training is performed to match the new sample pair. $\beta$ is a threshold defined as the least acceptable degree of excitation of the existing rule base. It decays during the learning process to limit the size of the network. If the firing strength $y_j^{(3)} < \beta$, then a new rule is generated.

The free parameters to be initialized for the new rule include the membership functions of the external variables, the membership functions of the internal variables, and the weights of the consequence singleton. These parameters are all adjustable later in the parameter learning phase to minimize an objective function. Therefore, it is not necessary to spend much time on the assignment of centers and widths for finding a perfect cluster. For an input pattern $(x_i, y_i)$, the parameters of a new fuzzy rule are initialized as follows:

$$\begin{aligned} m_{ij} &= x_i \\ \hat{m}_{ij} &= a\left(y_{\text{EMG}}(t-1) - 0.5\right) \\ \sigma_{ij} &= \hat{\sigma}_{ij} = b \\ W_{kj} &= y_i \end{aligned}$$

where $a$ is a scalar, and $b$ is a prespecified constant.

The centers of the new fuzzy membership functions of external variables $(m_{ij})$ are set to be $x_i$. The centers of the new fuzzy membership functions of internal variables $(\hat{m}_{ij})$ are set to be the rescaled previous EMG output, which are fed back. The width of all new Gaussian membership functions is kept at a prespecified constant value to reduce the computational load. Similar methods are employed in [12] and [21] to allocate the membership function of a new fuzzy rule. The initial link weight $W_{kj}$ (the output singleton) is set to $y_i$.

The free parameters in the fuzzy inference mechanism are then tuned after new rules are created. Parameter tuning is carried out concurrently with the structure adaptation. The ordered derivative [22] is used to derive the learning algorithm.

The error function to be minimized is

$$E(t+1) = \frac{1}{2}\sum_{k=1}^{K}\left(d_k(t+1) - y_k^{(4)}(t+1)\right)^2 = \frac{1}{2}\sum_{k=1}^{K}\varepsilon(t+1)^2 \tag{10}$$

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

*1) Tuning the Output Singleton:* The update rule for the output singleton $w_{kj}$ (the weights of the connections between layer 3 and layer 4) is

$$w_{kj}(t+1) = w_{kj}(t) - \eta \frac{\partial E(t+1)}{\partial w_{kj}} \tag{11}$$

where

$$\frac{\partial E(t+1)}{\partial w_{kj}} = \frac{\partial E(t+1)}{\partial y_k^{(4)}} \frac{\partial y_k^{(4)}}{\partial w_{kj}} = \varepsilon(t+1) \frac{y_j^{(3)}}{\sum_{j=1}^{M} y_j^{(3)}}. \tag{12}$$

*2) Tuning the Membership Functions of the External Variables:* The centers of the membership functions of external variables are $m_{ij}$. The update rule is

$$m_{ij}(t+1) = m_{ij}(t) - \eta \frac{\partial E(t+1)}{\partial m_{ij}} \tag{13}$$

where

$$\frac{\partial E(t+1)}{\partial m_{ij}} = \frac{\partial E(t+1)}{\partial y_j^{(3)}} \frac{\partial y_j^{(3)}}{\partial m_{ij}} = \sum_{k=1}^{K} \varepsilon(t+1) \cdot D \cdot \frac{\partial y_j^{(3)}}{\partial m_{ij}} \tag{14}$$

in which $D$ is defined as follows for notation simplicity:

$$D = \frac{\left(w_{kj} - y_k^{(4)}(t+1)\right)}{\sum_{j=1}^{M} y_j^{(3)}}. \tag{15}$$

From (7), we get

$$y_j^{(3)} = \exp\left(-\sum_{i=1}^{N_1} \frac{\left(y_i^{(1)}(t) - m_{ij}\right)^2}{\sigma_{ij}^2} - \sum_{i=N_1+1}^{N} \frac{\left(y_i^{(4)}(t) - \hat{m}_{ij}\right)^2}{\hat{\sigma}_{ij}^2}\right) \tag{16}$$

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

Then the derivative can be written as

$$\frac{\partial y_j^{(3)}}{\partial m_{ij}} = y_j^{(3)} \left(A_1 - \sum_{i=N_1+1}^{N} B \cdot \frac{\partial y_j^{(4)}(t)}{\partial m_{ij}}\right) \tag{17}$$

where $A_1$ and $B$ are defined as

$$A_1 = \frac{2\left(y_i^{(1)}(t) - m_{ij}\right)}{\sigma_{ij}^2} \tag{18}$$

$$B = \frac{2\left(y_i^{(4)}(t) - \hat{m}_{ij}\right)}{\hat{\sigma}_{ij}^2}. \tag{19}$$

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

$$\frac{\partial y_j^{(4)}(t)}{\partial m_{ij}} = D \cdot y_j^{(3)} \times \left(A_1(t-1) - \sum_{i=N_1+1}^{N} B(t-1) \cdot \frac{\partial y_j^{(4)}(t-1)}{\partial m_{ij}}\right). \tag{20}$$

The update rule for the width of the membership functions of external variables $(\sigma_{ij})$ is

$$\sigma_{ij}(t+1) = \sigma_{ij}(t) - \eta \frac{\partial E(t+1)}{\partial \sigma_{ij}} \tag{21}$$

in which

$$\frac{\partial E(t+1)}{\partial \sigma_{ij}} = \frac{\partial E(t+1)}{\partial y_j^{(3)}} \frac{\partial y_j^{(3)}}{\partial \sigma_{ij}} = \sum_{k=1}^{K} \varepsilon(t+1) \cdot D \cdot \frac{\partial y_j^{(3)}}{\partial \sigma_{ij}} \tag{22}$$

$$\frac{\partial y_j^{(3)}}{\partial \sigma_{ij}} = y_j^{(3)} \left(A_2 - \sum_{i=N_1+1}^{N} B \cdot \frac{\partial y_j^{(4)}(t)}{\partial \sigma_{ij}}\right) \tag{23}$$

$$A_2 = \frac{2\left(y_i^{(1)}(t) - m_{ij}\right)^2}{\sigma_{ij}^3}. \tag{24}$$

The recursive function of $\partial y_j^{(4)}/\partial \sigma_{ij}$ is in the following form:

$$\frac{\partial y_j^{(4)}(t)}{\partial \sigma_{ij}} = D \cdot y_j^{(3)} \times \left(A_2(t-1) - \sum_{i=N_1+1}^{N} B(t-1) \cdot \frac{\partial y_j^{(4)}(t-1)}{\partial \sigma_{ij}}\right). \tag{25}$$

*3) Tuning the Membership Functions of the Internal Variables:* The update rule for the center of the membership functions of internal variables $(\hat{m}_{ij})$ is

$$\hat{m}_{ij}(t+1) = \hat{m}_{ij}(t) - \hat{\eta} \frac{\partial E(t+1)}{\partial \hat{m}_{ij}} \tag{26}$$

where

$$\frac{\partial E(t+1)}{\partial \hat{m}_{ij}} = \frac{\partial E(t+1)}{\partial y_j^{(3)}} \frac{\partial y_j^{(3)}}{\partial \hat{m}_{ij}} = \sum_{k=1}^{K} \varepsilon(t+1) \cdot D \cdot \frac{\partial y_j^{(3)}}{\partial \hat{m}_{ij}} \tag{27}$$

and

$$\frac{\partial y_j^{(3)}}{\partial \hat{m}_{ij}} = y_j^{(3)} \left(\sum_{i=N_1+1}^{N} B \cdot \frac{\partial y_j^{(4)}(t)}{\partial \hat{m}_{ij}} - C_1\right) \tag{28}$$

where

$$C_1 = \frac{2\left(y_i^{(1)}(t) - \hat{m}_{ij}\right)}{\hat{\sigma}_{ij}^2} \tag{29}$$

$$\frac{\partial y_j^{(4)}(t)}{\partial \hat{m}_{ij}} = D \cdot y_j^{(3)} \times \left(\sum_{i=N_1+1}^{N} B(t-1) \cdot \frac{\partial y_j^{(4)}(t-1)}{\partial \hat{m}_{ij}} - C_1(t-1)\right). \tag{30}$$

The update rule for the width of the membership functions of internal variables $(\hat{\sigma}_{ij})$ is

$$\hat{\sigma}_{ij}(t+1) = \hat{\sigma}_{ij}(t) - \hat{\eta}\frac{\partial E(t+1)}{\partial \hat{\sigma}_{ij}} \tag{31}$$

where

$$\frac{\partial E(t+1)}{\partial \hat{\sigma}_{ij}} = \frac{\partial E(t+1)}{\partial y_j^{(3)}} \frac{\partial y_j^{(3)}}{\partial \hat{\sigma}_{ij}} = \sum_{k=1}^{K} \varepsilon(t+1) \cdot D \cdot \frac{\partial y_j^{(3)}}{\partial \hat{\sigma}_{ij}} \tag{32}$$

and

$$\frac{\partial y_j^{(3)}}{\partial \hat{\sigma}_{ij}} = y_j^{(3)} \left(\sum_{i=N_1+1}^{N} B \cdot \frac{\partial y_j^{(4)}(t)}{\partial \hat{\sigma}_{ij}} - C_2\right) \tag{33}$$

where

$$C_2 = \frac{2\left(y_i^{(1)}(t) - \hat{m}_{ij}\right)^2}{\hat{\sigma}_{ij}^3} \tag{34}$$

$$\frac{\partial y_j^{(4)}(t)}{\partial \hat{\sigma}_{ij}} = D \cdot y_j^{(3)} \times \left(\sum_{i=N_1+1}^{N} B(t-1) \cdot \frac{\partial y_j^{(4)}(t-1)}{\partial \hat{\sigma}_{ij}} - C_2(t-1)\right). \tag{35}$$

The initial values of $\partial y_j^{(4)}(t)/\partial m_{ij}$, $\partial y_j^{(4)}(t)/\partial \sigma_{ij}$, $\partial y_j^{(4)}(t)/\partial \hat{m}_{ij}$, and $\partial y_j^{(4)}(t)/\partial \hat{\sigma}_{ij}$ are set to zero. As training goes on, these parameters will be updated together with other free parameters in the network.

## III. Simulations and Results

This section shows the results and the performance of the proposed model. EMG data were collected in the Biodynamics Laboratory. The properties of the filtering methods are high pass (30 Hz), low pass (500 Hz), and averaging window width (40 ms). All the EMG data were normalized by the maxima (EMGmax) recorded from each muscle during a series of six static calibration exertions for the purposes of magnitude comparison and data uncertainty reduction.

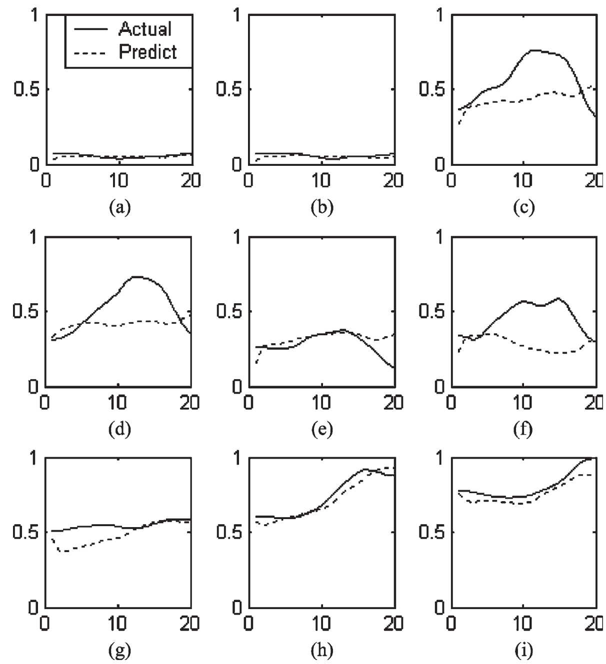


Fig. 3. Output after 20 training epochs (the first six are EMG signals, and the last three are normalized forces). (a) RLD. (b) LLD. (c) RES. (d) LES. (e) RIO. (f) LIO. (g) Lateral shear force. (h) A–P shear force. (i) Spinal compression.

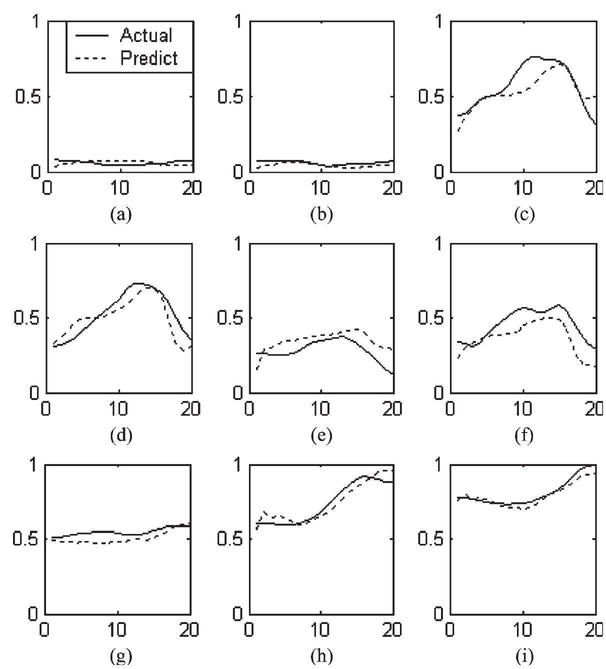


Fig. 4. Output after 400 training epochs (the first six are EMG signals, and the last three are normalized forces). (a) RLD. (b) LLD. (c) RES. (d) LES. (e) RIO. (f) LIO. (g) Lateral shear force. (h) A–P shear force. (i) Spinal compression.

Different models were built for different task conditions. For each task condition, the training and test were conducted as follows. Each time, we take 1/4 of the data out for test and use the other 3/4 to train the model. We take a different 1/4 for test and use the rest for training the next time. Continue this process until all the data are evaluated for test. The principle is that the data used for test should not be used for training the model. Otherwise, the generalization capabilities of the model cannot be proven.

Rule 1:

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 $\mu(0.009, 0.805)$ and $EMG_6(t)$ 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 0.218 and $EMG_6(t+1)$ is 0.102

Fig. 5. First fuzzy rule generated by the RFNN model.

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 ($m_{ij}$, $\sigma_{ij}$, and $w_{kj}$) is $\eta = 0.01$. The initial threshold $\beta$ for firing strength is set as 0.2. As stated before, the learning rates for the parameters of external inputs (kinematic variables) and internal inputs (EMG feedback) are different. Since the initial values of the parameters of internal inputs are small random values while the initial values of the parameters of external inputs are good values with physical meaning, the convergence of the latter is faster than the convergence of the former. This can be seen in Figs. 3 and 4. Fig. 3 shows the results obtained after a small number of epochs (20 epochs). Notice that the predicted forces (the last three figures) are already quite close to the actual forces. However, for the EMG signals (the feedback), the prediction is still very poor. Fig. 4 was obtained after 400 epochs. In this figure, both the forces and the EMG signals are predicted well, which means the parameters of both the external inputs and the feedback are well trained.

Fuzzy rules obtained are of the following form as shown in Fig. 5.

Values in the above fuzzy rule are normalized. To interpret the rule, they need to be converted to meaningful values. Membership functions of actual values of Rule 1 are listed in Tables I and II. All variables in the fuzzy rules generated by the proposed model are physical variables, which is an advantage of this model. As we have mentioned in Section I, the fuzzy rules generated by other types of feedback contain internal variables that have no physical meaning and, therefore, are difficult to interpret. For the rules generated by the proposed RFNN model, we can decompose the kinematics–EMG–force relationship into kinematics–EMG relationship, EMG–force relationship, and kinematics–force relationship, as stated previously. In the rule listed in Tables I and II, the kinematic variables and EMG signals are small, and the output forces are also relatively small. In other rules, we found that forces change in the same direction as the EMG signals. However, the kinematic–force relationship is not so simple. After examining all fuzzy rules, we can conclude that larger EMG signals normally lead to larger spinal forces, while the relationship does not hold in the kinematics–force relationship. By comparing and analyzing all fuzzy rules generated by the RFNN model, some useful information can be obtained. For example, some kinematic variables (such as axis trunk velocity and lateral trunk acceleration) have less influence on the spinal forces than other kinematic variables, according to the widths of their membership functions (the widths are always large, indicating that the output is not sensitive to the variable).

TABLE I
MEMBERSHIP FUNCTIONS OF KINEMATIC VARIABLES AND EMG SIGNALS AT TIME $t$ (RULE 1)

| Variable Name | Membership Function |
|---|---|
| Sagittal trunk moment | (65.8033, 110.3139) |
| Lateral trunk moment | (75.7623, 47.4398) |
| Axis trunk moment | (29.1712, 20.5732) |
| Sagittal trunk angle | (-52.2221, 13.3591) |
| Lateral trunk angle | (1.0038, 7.5505) |
| Axis trunk angle | (1.2644, 13.3500) |
| Sagittal trunk velocity | (19.6938, 44.5477) |
| Lateral trunk velocity | (-5.0671, 2.4169) |
| Axis trunk velocity | (-4.2402, 1.4377) |
| Sagittal trunk acceleration | (64.8514, 71.5178) |
| Lateral trunk acceleration | (1.4666, 15.5876) |
| Axis trunk acceleration | (6.5360, 20.5340) |
| RLD $(t)$ | (0.0630, 1.2659) |
| LLD $(t)$ | (0.0715, 1.2661) |
| RES $(t)$ | (0.3736, 1.2397) |
| LES $(t)$ | (0.3733, 1.2286) |
| RIO $(t)$ | (0.2561, 1.2566) |
| LIO $(t)$ | (0.2854, 1.2453) |

TABLE II
OUTPUT SINGLETON OF FORCES AND EMG SIGNALS AT TIME $t+1$ (RULE 1)

| Variable Name | Output Singleton |
|---|---|
| Lateral shear force | -38.325 |
| A-P shear force | -68.556 |
| Spinal compression | -2165.342 |
| RLD $(t+1)$ | 0.0931 |
| LLD $(t+1)$ | 0.1004 |
| RES $(t+1)$ | 0.3035 |
| LES $(t+1)$ | 0.4131 |
| RIO $(t+1)$ | 0.2399 |
| LIO $(t+1)$ | 0.3976 |

### *A. Predictions for Sagittal Symmetric Motions and Asymmetrical Motions*

In a sagittal symmetric lifting motion, the subject does not turn his body. The motion is done sagittally. This kind of

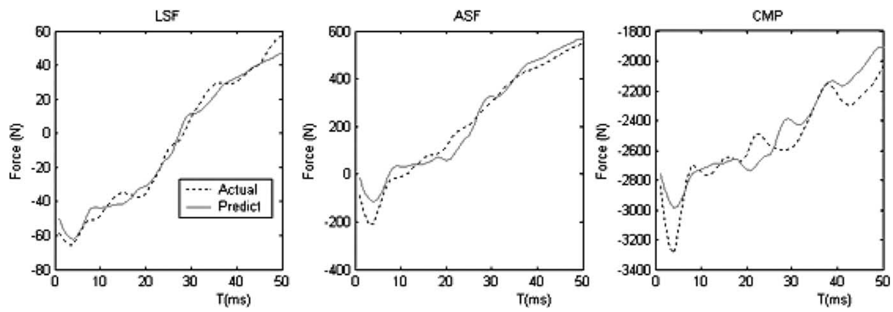


Fig. 6. Prediction of a sagittal symmetric motion.

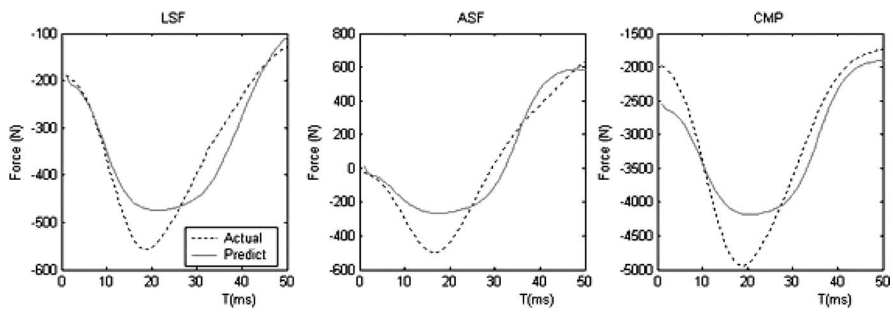


Fig. 7. Prediction of an asymmetrical motion.

motion is simpler and easier to model as compared with the asymmetrical motion. Fig. 6 gives an example of predicting such motion. The dotted curves are the targets, whereas the solid curves are the predicted forces.

If we are predicting the asymmetrical motions, we could expect that the errors will be bigger than those of the sagittal symmetric motions. This is because the motion is asymmetrical and, thus, more complex than the symmetric motion. The subjects were required to turn their bodies during the lifting task. One example of such predictions is shown in Fig. 7. As we expected, the predicted curve does not fit the target curve as well as predicting sagittal symmetric motions. The statistical information is given in Section III-B. Considering the complexity of such motions, the prediction quality of the asymmetrical motions is still acceptable.

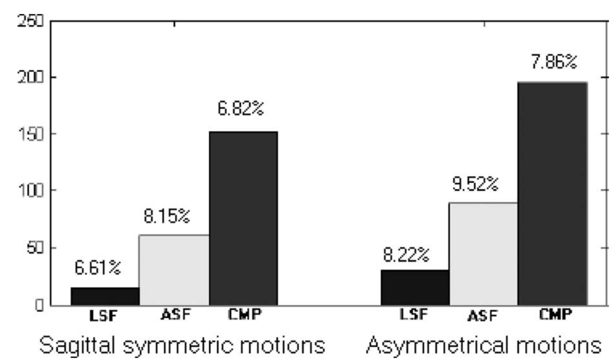


Fig. 8. Overall MAEs (force in newtons) and percentage errors of different types of tasks.

## *B. Statistical Results*

Statistical results are used to evaluate the system performance on different types of tasks. The overall mean absolute errors (MAEs) and percentage errors of different tasks are shown in Fig. 8. The variations of the lateral shear force, A–P shear force, and spinal compression are around 300, 800, and 2500 N, respectively. The MAEs are out of such ranges. Since spinal forces cannot be measured directly, the target spinal forces were obtained through a biomechanical model, as mentioned in Section I. From the figure, we can see that the MAEs of the predicted sagittal symmetric tasks are much smaller than those of the predicted asymmetrical tasks. It is reasonable since the muscular activities are much more complicated in the asymmetrical tasks.

According to the statistical results, the MAEs are acceptable, and the performance of the RFNN model is superior to that of a recurrent neural network (RNN) model presented in [9]. In [9], the MAEs of sagittal symmetric motions for lateral shear force, A–P shear force, and spinal compression are 14.5, 60.3, and 152, respectively. For the RFNN model, those values are 12.5, 52.7, and 147.7, respectively. The prediction quality is improved in the RFNN model. For asymmetrical motions, the improvement is even more significant.

## *C. Discussions*

EMG signals are influenced not only by kinematic variables but also by lifting task conditions and difference between subjects. Therefore, we have to consider those factors. Due to this fact, we collected task variables and subject variables (anthropometric characteristics of the subjects).

Task variables include object weight, lifting height, and asymmetry.

Subject variables include age, body weight, standing height, shoulder height, upper arm length, lower arm length, elbow height, spine length, upper leg length, lower leg length, trunk circumference, trunk depth (pelvis), trunk depth (xyphoid), trunk breadth (pelvis), and trunk breadth (xyphoid).

It is not possible to feed all 12 kinematic variables, three task variables, and 15 subject variables to the model (and also the feedback of six EMG signals). It will make the model very complicated and decrease the generalization ability. Instead, we developed different models for different task conditions. For example, the model for task condition "object weight 15 lb, from knee to elbow, from 30 counterclockwise to sagittal" and the model for task condition "object 30 lb, from floor to waist, from sagittal to 60 clockwise" are different. By this way, task variables do not need to be fed to the network, and the specific models for different task conditions can have better performance than a "universal model."

For other variables, a pruning process has been done before building the model. The importance of 27 variables including 12 kinematic variables and 15 subject variables were estimated. The influence rates of these variables to EMG and forces were identified using a method called fuzzy average with fuzzy cluster distribution (FAFCD) [23]. Results indicate that kinematic variables have much more influence on EMG than subject variables do. To reduce the input space dimension, subject variables were not used as input of the model. However, we still considered three most important subject variables as identified in [23]. They are standing height, lower arm length, and spine length. Subjects were grouped according to these variables, and different models were built for different groups. For some task conditions, lifting data of more than 70 subjects were collected. The more subjects are used to train a model, the better chance the model can predict a new subject. However, if a subject has very different muscular behaviors, the EMG and force prediction would be poor. The statistical results were obtained on a large data set.

In Section III-B, the target spinal forces were obtained through a biomechanical model. It is impossible to directly validate biomechanical models of the human spine *in vivo*; however, it is possible to indirectly validate the model predictions based on their output. In a biologically assisted model such as the one used in this paper, the muscle EMG activities are processed and used as input to a series of biomechanical relationships that predict spine load as well as moment imposed on the spine due to the task [24]. While it is not practical to compare the measured and predicted spine forces *in vivo*, it is possible to compare the measured dynamic moments with those predicted via the EMG signals. In this comparison, several measures are used to assess model validity: the $R^2$ statistic, which indicates the degree of variability that is accounted for by the model (since this is a dynamic signal, strong $R^2$ values indicate good model fidelity); the average absolute error, which considers the magnitude of the difference between the total measured and predicted dynamic moment signals; and the muscle gain estimate, which must be within physiologically realistic bounds for the model to be considered a realistic representation of muscle load. Several previously published papers [25]–[27] have established the validity of each of these measures for each of the three-dimensional forces imposed on the spine. Given these evaluations, the model is considered to have good fidelity, repeatability, and physiologically consistent predictions. Therefore, the output of the biomechanical model was used as target spinal forces in this paper.

## IV. Conclusion

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

## References


[1] D. G. Lloyd and T. F. Besier, "An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments *in vivo*," *J. Biomech.*, vol. 36, no. 6, pp. 765–776, Jun. 2003.

[2] P. A. Crosby, "Use of surface electromyogram as a measure of dynamic force in human limb muscles," *Med. Biol. Eng. Comput.*, vol. 16, no. 5, pp. 519–524, Sep. 1978.

[3] L. Wang and T. S. Buchanan, "Prediction of joint moments using a neural network model of muscle activations from EMG signals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 10, no. 1, pp. 30–37, Mar. 2002.

[4] J. J. Luh, G. C. Chang, C. K. Cheng, J. S. Lai, and 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. l, no. 9, pp. 173–183, 1999.

[5] M. M. Liu, W. Herzog, and H. H. Savelberg, "Dynamic muscle force predictions from EMG: An artificial neural network approach," *J. Electromyogr. Kinesiol.*, vol. 9, no. 6, pp. 391–400, Dec. 1999.

[6] S. E. Hussein and M. H. Granat, "Intention detection using a neuro-fuzzy EMG classifier," *IEEE Eng. Med. Biol. Mag.*, vol. 21, no. 6, pp. 123–129, Nov./Dec. 2002.

[7] K. Kiguchi, T. Tanaka, and T. Fukuda, "Neuro-fuzzy control of a robotic exoskeleton with EMG signals," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 4, pp. 481–490, Aug. 2004.

[8] Y. Hou, J. M. Zurada, and W. Karwowski, "Prediction of EMG signals of trunk muscles in manual lifting using a neural network model," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 25–29, 2004, pp. 1935–1940.

[9] ——, "Prediction of dynamic forces on lumbar joint using a recurrent neural network model," in *Proc. Int. Conf. Mach. Learning Appl. (ICMLA)*, 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. Syst., Man, Cybern. B, Cybern.*, vol. 30, no. 2, pp. 358–364, Apr. 2000.

[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. Fuzzy Syst.*, vol. 9, no. 4, pp. 578–594, Aug. 2001.

[12] C. F. Juang and C. T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 1, pp. 12–32, Feb. 1998.

[13] ——, "A recurrent self-organizing neural fuzzy inference network," *IEEE Trans. Neural Netw.*, vol. 10, no. 4, pp. 828–845, Jul. 1999.

[14] C. H. Lee and C. C. Teng, "Identification and control of dynamic systems using recurrent fuzzy neural networks," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 4, pp. 349–366, Aug. 2000.

[15] C. M. Lin and C. F. Hsu, "Supervisory recurrent fuzzy neural network control of wing rock for slender delta wings," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 5, pp. 733–742, Oct. 2004.

[16] C. F. Juang, "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 2, pp. 155–170, Apr. 2002.

[17] F. J. Lin and R. J. Wai, "Hybrid control using recurrent fuzzy neural network for linear-induction motor servo drive," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 1, pp. 68–90, Jan. 2001.

[18] F. J. Lin, R. J. Wai, and C. M. Hong, "Hybrid supervisory control using recurrent fuzzy neural network for tracking periodic inputs," *IEEE Trans. Neural Netw.*, vol. 12, no. 1, pp. 102–115, Feb. 2001.

[19] Y. C. Wang and D. Zipser, "A learning algorithm for continually running recurrent neural networks," *Neural Comput.*, vol. 1, no. 2, pp. 270–280, 1989.

[20] S. J. Lee and C. S. Ouyang, "A neuro-fuzzy system modeling with self-constructing rule generation and hybrid SVD-based," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 3, pp. 341–353, Jun. 2003.

[21] C. J. Lin and C. C. Chin, "Prediction and identification using wavelet-based recurrent fuzzy neural networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2144–2154, Oct. 2004.

[22] R. J. Williams, C. J. Chien, and C. C. Teng, "Direct adaptive iterative learning control of nonlinear systems using an output-recurrent fuzzy neural network," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 3, pp. 1348–1359, Jun. 2004.

[23] Y. Hou, J. M. Zurada, W. Karwowski, W. S. Maras, and K. Davis, "Identification of input variables using fuzzy average with fuzzy cluster distribution," *IEEE Trans. Fuzzy Syst.,* to be published.

[24] Y. Hou, J. M. Zurada, W. Karwowski, and W. S. Maras, "A hybrid neurofuzzy approach for spinal force evaluation in manual materials handling tasks," in *Proc. ICNC*, Changsha, China, Aug. 27–29, 2005, pp. 1216–1225.

[25] W. S. Marras, K. P. Granta, and K. G. Davis, "Variability in spine loading model performance," *Clin. Biomech.*, vol. 14, no. 8, pp. 505–514, Oct. 1999.

[26] K. P. Granata, W. S. Marras, and K. G. Davis, "Variation in spinal load and trunk dynamics during repeated lifting exertions," *Clin. Biomech.*, vol. 14, no. 6, pp. 367–375, Jul. 1999.

[27] W. S. Marras and K. P. Granata, "The development of an EMG-assisted model to assess spine loading during whole-body free-dynamic lifting," *J. Electromyogr. Kinesiol.*, vol. 7, no. 4, pp. 259–268, 1997.

**Yanfeng Hou** (S'05) received the B.S. degree in electrical engineering from Sichuan University, Sichuan, China, and the M.S. degree in electrical engineering from Wuhan University, Hubei, China. He is currently working toward the Ph.D. degree at the University of Louisville, Louisville, KY.

His research interests are in the areas of artificial neural networks, fuzzy logic, data mining, and other artificial intelligence and machine-learning-based methodologies with focus on ergonomics and biomedical applications.

Mr. Hou is a member of the IEEE CIS Standards Committee and the Tau Beta Pi Engineering Honor Society.

**Jacek M. Zurada** (M'82–SM'83–F'96) is the Samuel T. Fife Alumni Professor and the Chair of the Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY. He was a Visiting Professor with Princeton University, the National University of Singapore, Nanyang Technological University, and the University of Karlsruhe. He is the author or coauthor of more than 250 journal and conference papers in the areas of neural networks, computational intelligence, data mining, image processing, and very large scale integrated circuits. He has delivered numerous invited plenary conference presentations and seminars throughout the world. He was the coeditor of the 2000 MIT Press volume *Knowledge-Based Neurocomputing*, the author of the 1992 PWS text *Introduction to Artificial Neural Systems*, a contributor to the 1994 and 1995 Ablex volumes *Progress in Neural Networks*, and the coeditor of the 1994 IEEE Press volume *Computational Intelligence: Imitating Life*. He is an Associate Editor of *Neurocomputing*.

Dr. Zurada has received a number of awards for distinction in research and teaching, including the 1993 Presidential Award for Research, Scholarship and Creative Activity. In 2001, he received the University of Louisville President's Distinguished Service Award for Service to the Profession. In 2003, he was conferred the Title of the Professor by the President of Poland, Aleksander Kwasniewski, and the Honorary Professorship of Hebei University in China. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, PART I AND PART II. In 1998–2003, he was the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS. In 2001–2003, he was a member of the Editorial Board of the PROCEEDINGS OF THE IEEE. He is a past President and a Distinguished Speaker of the IEEE Computational Intelligence Society. Since 2005, he has been serving as a Foreign Member of the Polish Academy of Sciences.

**Waldemar Karwowski** received the M.S. degree in production engineering and management from the Technical University of Wroclaw, Wroclaw, Poland, in 1978, the Ph.D. degree in industrial engineering from Texas Tech University, Lubbock, in 1982, the D.Sc. (Dr. Hab.) degree in management science from the Institute for Organization and Management in Industry, Warsaw, Poland, in 2004, and the Doctor of Science Honoris Causa degrees from the South Ukrainian State K.D. Ushynsky Pedagogical University of Odessa, Odessa, Ukraine, and from the Technical University of Kosice, Kosice, Slovakia, in 2004 and 2006, respectively.

He is the J. B. Speed School of Engineering Alumni Scholar for Research, a Professor of industrial engineering, and the Director of the Center for Industrial Ergonomics, University of Louisville, Louisville, KY. His research, teaching, and consulting activities focus on work system design, organizational and management ergonomics, human–system integration and safety of advanced manufacturing, and neurofuzzy modeling and fuzzy systems applications.

Prof. Karwowski is a Fellow of the International Ergonomics Association, the Human Factors and Ergonomics Society, the Institute of Industrial Engineers, and The Ergonomics Society (U.K.).

**William S. Marras** is the Honda Endowed Chair of the Department of Industrial and Systems Engineering, The Ohio State University, Columbus. He is also the Director of the Biodynamics Laboratory and has joint appointments in the Departments of Orthopaedic Surgery, Physical Medicine, and Biomedical Engineering. His research is centered on occupational biomechanics issues. He has pioneered our understanding of how motion influences the risk of musculoskeletal disorders in the workplace. His research interest includes workplace biomechanical epidemiologic studies, laboratory biomechanics studies, mathematical modeling, and clinical studies of the lower back and upper extremities.

**Kermit Davis** received the Ph.D. degree from The Ohio State University, Columbus, in 2001.

He is currently with the Department of Environmental Health, University of Cincinnati, Cincinnati, OH. He was trained in industrial engineering with specialization in occupational ergonomics and low back biomechanics with special interest in multiple exposures, both physical and psychosocial stressors. He has published numerous articles about the impact of workplace stressors on the lower back, including studies evaluating warehousing, patient handling, alternative modes of handling (e.g., team lifting, one-hand lifting, and pushing/pulling), injured populations, and ergonomic interventions (e.g., back belts, lifting hoists, and adjustable fork lifts). His current research is focusing on the investigation of the effect of physical workplace demands as well as mental workload on the responses within the lower back.

Dr. Davis has received several major research awards, including the Alphonse Chapanis Student Paper Award from the Human Factors and Ergonomics Society, the Volvo Award for Low Back Pain Research in Biomechanical Studies from the International Society for the Study of the Lumbar Spine, the Alice Hamilton Award from the National Institute for Occupational Safety and Health, and the Promising Young Scientist Award from the International Society of Biomechanics.