International Journal of Soft Computing 6 (3): 54-61, 2011
ISSN: 1816-9503

Minimizing Musculoskeletal Disorders Within Grinding Machine Workers Using Neural Networks

[1]Aman Sachdeva, [1]B.D. Gupta and [2]Sneh Anand
[1]Department of Mechanical Engineering, Anand Engineering College, Agra, India
[2]Center for Bio-Medical Engineering, Indian Institute of Technology, New Delhi, India

Abstract: Problems related to manufacturing system operation and controls are complex and time consuming because of the non-linearities involved in their formulation and solution. Fast solutions to these problems can be obtained only through parallel processing. Neural nets provide massive parallel processing facilities and may also be used efficiently to model systems with non-linearities. The capabilities of neural nets can therefore be well utilized in modelling and processing problems related to manufacturing systems. In order to reduce the burden on computers, algorithms involving optimization and complex equations can be converted to heuristics. These heuristics can then be represented in terms of rules and an expert system can be built, with the added advantage of obtaining solutions in a time intensive fashion. This study studies the application of neural nets to problem solving in manufacturing system operation and control and demonstrates how present methods for solving such problems can be converted to the neural net approach.

Key words: Musculoskeletal disorders, neural networks, computers, expert system, fashion, India

INTRODUCTION

Musculoskeletal disorders include a group of conditions that involve the nerves, tendons, muscles and supporting structures such as intervertebral discs. They represent a wide range of disorders which can differ in severity from mild periodic symptoms to severe chronic and debilitating conditions. Musculoskeletal disorders are the most common phenomenon that persists now a days among workers who are exposed to repetitive tasks. Workers tend to exert their bodies to these types of disorders unknowingly and tend to lose both on the financial end as well on the health end. Generally these types of disorders tend to decrease the efficiency of the workers over a length of time and most commonly among those who are highly exposed to repetitive tasks (Andersen *et al*., 2007; Choi, 2010; D'Angelo *et al*., 2006; Deeney and O'Sullivan, 2009). Repetitive, forceful or prolonged exertions of the hands; frequent or heavy lifting, pushing, pulling or carrying of heavy objects; prolonged awkward postures and vibration contribute to Work based Musculoskel Disorders (WMSDs). Jobs or working conditions that combine risk factors will increase the risk for musculoskeletal problems. The level of risk depends on how long a worker is exposed to these conditions, how often they are exposed and the level of exposure. The ligaments of the workers tend to depreciate over a period of time such that they lead to critical situations. These ligaments of the shoulder muscle as well as of the low back muscles are the major ones which tend to depreciate among repetitive lifting task workers (Antonopoulou *et al*., 2009; Barrero *et al*., 2009; Wang *et al*., 2007; Wiesinger *et al*., 2007).

When looking specifically at work-related musculoskeletal disorders, the Bureau of Labor Statistics (BLS) in India reports that in 1995, 62% (308,000) of all illness cases were due to disorders associated with repeated trauma. This figure does not include back injuries. BLS also reports that the number of cases of repeated trauma has increased significantly, rising from 23,800 cases in 1972-332,000 cases in 1994, a fourteen-fold increase. In 1995 the number of cases decreased by 7%-308,000 reported cases but this number still exceeds the number of cases in any year prior to 1994. When looking specifically at cases involving days away from work for which more detailed information is available, BLS reports that in 1994, approximately 32% or 705,800 cases were the result of overexertion or repetitive motion. Thus in order to reduce these work based musculoskeletal disorders the challenges in designing the workstation for repetitive lifting task workers can be summarized into three main categories:

- Optimize differential lifting heights
- Minimizing musculoskeletal disorders
- Increase the efficiency of workers

Corresponding Author: B.D. Gupta, Department of Mechanical Engineering, Anand Engineering College, Agra, India

This study addresses the aforementioned challenges by proposing a methodology with the help of which musculoskeletal disorders can be minimized. The main aim shall be to model the problem into a function and apply soft computing method to train the input data and thus give suitable results for the given inputs. To train the input data Neural Network soft computing tool is used. Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. Commonly neural networks are adjusted or trained so that a particular input leads to a specific target output (Brewer *et al*., 2006; Caban-Martinez *et al*., 2010; Cecchini *et al*., 2010).

BASIC PRINCIPLES OF NEURAL NETS

Neural nets attempt to simulate the thinking and processing procedures of the human brain by modelling the neuron. The basic component of a neuron is known as the soma which is attached to the axon (Fig. 1).

The axon is electrically active and produces the pulse emitted by the neuron. Electrically passive dendrites receive input from neurons by means of specialized contacts called synapses which act as weights to the input information. The neuron is fired only when the sum of the weighted inputs is above a certain threshold. Information received by neurons may be processed in parallel or sequentially or as a combination of both. For some kinds of stimuli, the reaction of the brain is typecast in the sense that for a certain input only a specific output is obtained. Although, the mechanism behind this process is not fully understood, research has started on the simulation of processes in neural nets which can match inputs to required outputs and incorporate variations in input patterns to account for output patterns. This pattern matching ability of neural nets has played a vital role in their implementation (Ebron *et al*., 1990; Fujiwara *et al*., 1986; Jansen and Puttgen, 1987; Wollenberg, 1986). Human behavior is the integration of hierarchical, modular, distributed computation and automatic learning processes. The learning process is based on decslarative and reflexive mechanisms. In the declarative state an assessment is made of why or how a certain action is to be performed. In the reflexive state, a reaction appears to the action. Tasks learnt from the declarative state normally become reflexive over a period of time. Neural nets simulate such processes by training.

Neurons interact in feedforward, feedback, fully connected or partially connected fashions. The connections of the neurons in the brain as well as in the neural network model are shown in Fig. 2. The feedback

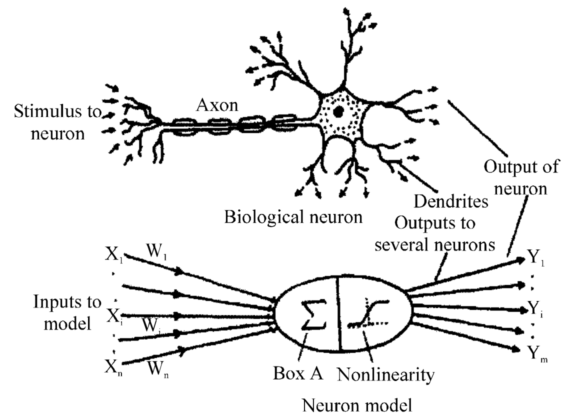

Fig. 1: Single biological neuron and its model

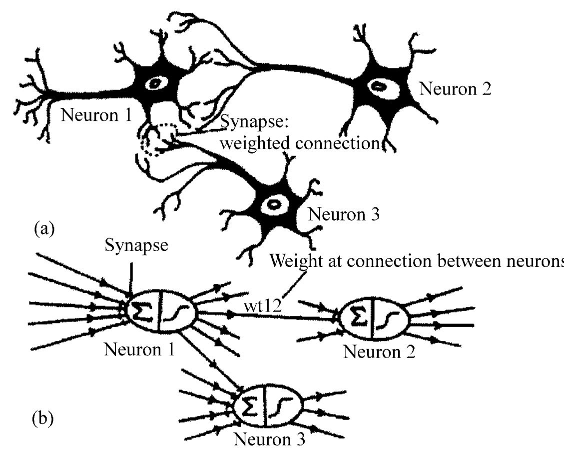

Fig. 2: Interconnections between (a) biological neurons and (b) modelled neurons

path influences the nature of adaptivity and trainability. If the selection of weights and thresholds in an artificial neural net is automated then, this could be thought of as a learning mechanism (Rahman and Bhatnagar, 1988; Sakaguchi and Matsumoto, 1983; Santosa and Tan (1990). This learning capability of neural nets distinguishes them from conventional computer software.

At present, neural nets show good potential for ever-improving performance through dynamic learning. A single neuron in a neural net can be shown in Fig. 1 where Xi is the input, Wi is the weight carried by the input Xi and box A represents the linear combination of weighted inputs. The output Y of box A is passed through a non-linear function called the sigmoid or learning function (Fig. 3).

For nerve connections in the representation of Fig. 1 to be a good approximation to an actual neuron, all neurons connected together must form a stable system. The functionality of this system can be found by

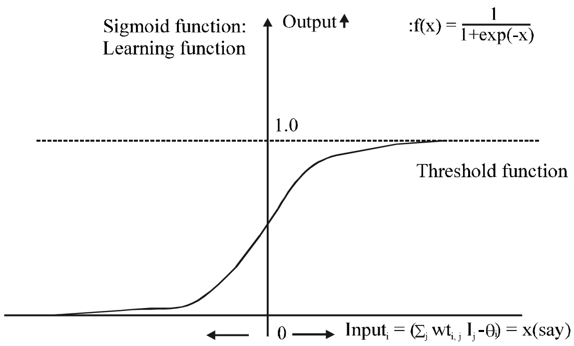

Fig. 3: Threshold function (the non-linear (sigmoidal) function used in every modelled neuron)

modifying the weights. One can think of the combination of neurons and synapses as hardware while weights and thresholds can be treated as software. Neural nets can be viewed as one of the following:

- A set of non-linear differential equations or non-linear difference equations or
- A nonlinear transformation between input and output

Information processing in neural nets is nonalgorithmic so the influence of approximations of mathematical modelling is reduced to a minimum. Further, neural nets are capable of handling uncertainties so that results obtained through trained neural nets, even with partial inputs, may be very close to the results with complete knowledge. Depending on the neuron interaction, neural nets can be classified as feedforward; feedback.

Feedforward neural networks: Here, neurons are arranged in layers like directed graphs. Inputs are applied to the layers and outputs are collected. Stability is not a problem here because these networks are loop free. The computation time in such neural nets is the time required for signals to propagate and the output to settle.

Feedback neural networks: Here, neurons are arranged in the form of undirected graphs. The connections in this case are symmetrical and bidirectional. Feedback neural network models are sequential or asynchronous. Here the system is initialized and then it evolves to a final state in the course of time. The stability of this type of neural net is analyzed with the help of energy functions that can be defined in terms of states or neurons, weights and thresholds. Under certain conditions, the energy function decreases monotonically as the network moves from one state to another. The stability and convergence can be analyzed by studying the descent of scalar energy functions instead of the transition of states.

NEURAL NETS IN MAUFACTURING SYSTEMS

Manufacturing systems may be viewed as a combination of the following activities:

Planning: To operate any manufacturing system efficiently and effectively, proper combination of man, machine, money and material are required. Most of these studies are performed off-line.

Design: Having planned the system now the resources must be adequately positioned within the production floor.

Control: Closed-loop and open-loop controls have to be used for machining control, material flow control in ventory control and quality control.

Operation: Quality fault diagnosis, break down analysis, production state estimations, etc. must be performed on a short-duration basis. These studies are done in real time/on-line.

Traditionally, a manufacturing system operates with large uncertainties. Retrieval of complete knowledge of the interconnections between variables to be controlled is very complex because of the size and non-linear behavior of the system. In the existing system, a decision taken in the case of changes in operating conditions involves the following:

- Use of a large data bank
- Conversion of data to usable information
- Use of input from past experience
- Implementation of control tasks

The last approach is considered to be the most effective. The decision-making procedure can be thought of as a pattern recognition problem.

PROBLEMS WHICH CAN BE SOLVED USING NEURAL NETS

In manufacturing systems, quality assessment and production monitoring problems can be solved by using pattern matching techniques. Depending upon the class of pattern, an appropriate action can be taken. Neural net based decision procedures are capable of recognizing patterns and responding in a real-time manner and these problems are therefore, very good candidates for their application. Another class of problems is production forecasting, state estimation, quality control and bad-data detection. These are solved by using difference equations, differential equations or recursive curve fitting. Since, neural nets themselves are defined in these terms, they are very good tools for solving such equations. Further, data may be misleading in complete or widely varying in nature. Neural nets therefore, provide a particularly effective solution as they are capable of handling uncertainties (Fukui and Kawakami, 1986; Liu and Tomsovic, 1986; Santosa and Tan, 1990).

A third class of problems includes material flow analysis, contingency analysis, prediction of material requirement and emergency control actions. These problems are difficult to solve because the mathematical equations involved and the optimization of their solutions is time consuming. By applying neural nets to these problems and training them extensively, it may be possible to reduce the size of data banks and knowledge bases with the added advantage of real-time solutions. Designers of these controllers could encounter the following problems:

- Redundancy in data used for system modelling
- Limitations in implementation of control laws based on heuristics or complicated mathematics (fuzzy theory or expert system based approaches, too, face limitations in terms of flexibility)
- Restrictions in measurements of system parameters

A standard control system tries to minimize the error between the reference and measured signals. In control systems associated with manufacturing systems this minimization procedure is rendered more complex because of the non-linearities involved and because of the multivariable or Multiple Input and Multiple Output (MIMO) nature of the system. To achieve fast, reliable and optimal performance of control systems, control loops can be implemented through parallel computer architecture.

The problem of non-linearities can be partially solved by using intelligent control systems. Complexities such as time dependence can be solved only through large knowledge based expert systems. These however have inherent drawbacks as learning procedures are neither fully understood nor possible to implement completely. Further, search processes may be intensive hence, time consuming (Sobajic and Pao, 1989; Talukdar *et al*., 1986; Tomsovic *et al*., 1986). Neural nets, on the other hand can handle situations of incomplete information, corrupt data and large data volumes, thus providing control systems with close to ideal performance. While the training period may be lengthy, once trained, neural nets can be used for real-time applications as they do not employ any search techniques.

ILLUSTRATION: MINIMIZING MUSCULOSKELETAL DISORDERS USING NEURAL NETS

A neural network model has been designed and tested for minimizing musculoskeletal disorders within grinding machining workers. The details of the neural network model and that of the training algorithm used to achieve this goal are described further.

Network structure (Fig. 4): The Artificial Neural Net (ANN) used in this problem is a two-layer feedforward net. The two layers are the hidden layer and the output layer. Input values are fed directly to the input layer which simply distributes them among different hidden layer connections. An auxiliary node has been inserted to decide the threshold of the nodes in the network and the input to these nodes is a biased value (1.0 in the present implementation).

Training algorithm and parameters: A standard back-propagation algorithm (Talukdar *et al*., 1986) was used to train the feedforward type of neural network. The algorithm is summarized in the steps. The mathematical formulations A, B, C, D, E and T are furthee described.

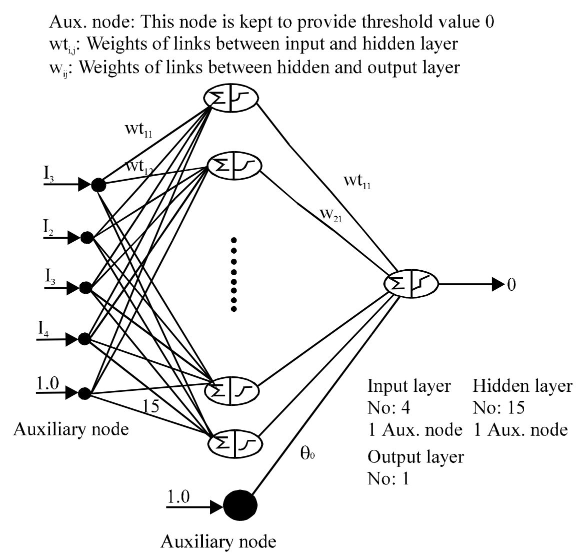

Fig. 4: Structure of the ANN used for short-term load forecating. Showing the input nodes, the auxiliary node, the hidden layer and the output node

Step 1: Set the initial weights and thresholds for all inputs using random numbers between 0.0 and 1.0 and set the number of iterations N-0.

Step 2: Read inputs and desired output in appropriate form (A).

Step 3: Find the output of each layer and then the final output (B) using the threshold function (T) and increment the number of iterations.

Step 4: Find the error E that is the deviation (C) from the desired output. If $E > E_0$ or the number of iterations $N > N_0$, stop processing and repeat the loop a fixed number of times to ensure better learning; else go to step 5 (here E_0 is the maximum permissible error and N_0 is the maximum number of iterations permitted).

Step 5: Calculate error functions δ_k and δ_j for the output layer and other hidden layers using (D).

Step 6: Modify the weights and thresholds using (E). Go to step 3.

These operations are to be performed for every set of input data over which we want to train the neural network. When the training with the first set of inputs is over, we present the new set and desired output for that input set and so on. After all the input sets are entered, the training is repeated with less iteration to improve the performance and the fault tolerance of the neural network model. The mathematical formulations used above are as follows. (T) Threshold function: sigmoid function:

$$f(x) = 1/\{1+\exp[-(x-x_0)/\theta]\}$$

Where:

- x = The variable value
- x_0 = The threshold
- θ = Slope of the sigmoid function

For finite x_0:

- f (x) = 0 as $x \rightarrow$ negative infinity
- = as $x \rightarrow$ positive infinity
- = Some value between 0 and 1 for any other x. In the MSD forecasting problem we used: $\theta = 1$
- x_0 = Obtained by training

The inputs are taken in normalized form between 0.1 and 0.9 using the equation:

$$€ = (x - MIN)/(MAX - MIN) * 0.8 + 1$$

MAX and MIN are the global maximun and minimum, respectively in the input set. The output of any layer j is defined as:

$$O_j = f(\textstyle\sum_i W_{ij}\, x_i - x_0)$$

Where:

- f ( ) = The threshold function described above
- W_{ij} = The weight between node i of the input and node j the next layer
- x_i = The input to node i
- x_0 = The threshold set by the auxiliary node connection between the auxiliary node and the node in which we are interested

The mean error criterion is used to measure the deviation of the neural network's output from the desired output:

$$Error = 0.5\textstyle\sum_I (Output_i - desired\ output_i)^2$$

Error functions

The error for the output layer: For the kth node in the output layer the error function used is δ_k which is represented as:

$$\delta_k = (t_k - O_k)\, O_k\, (1-O_k)$$

Where:

- O_k = The actual output
- t_k = The desired output

The error for the other layers: Since, the output of the hidden layer is not known, a different error function is used to evaluate the error for each hidden layer node as given:

$$\delta_j = O_j\, (1-O_j) \textstyle\sum_k W_{jk}\delta_k$$

Where:

- O_j = The output of node j
- δ_k = The error of node k in the output layer
- W_{jk} = Weight between node j and node k

The weights of the neural net were modified using the error functions and the weight change as follows:

$$W_{ij}\,(new) = W_{ij}\,(old) + \eta\, \delta_j\, i_i + \alpha\, [W_{ij}\,(old) - W_{ij}\,(*)]$$

Where:

- η and α = Acceleration functions
- W_{ij} (*) = The value of the weight two steps before the current one

Application to MSD forecasting: In this application a neural network is trained with the collected data to forecast MSD's using a feedforward network trained for forecasting with the back-propagation algorithm. For the SUBJECT data was collected with four inputs:

- Age of the SUBJECT
- Weight lifted
- Frequency of lift
- Lifting height

Input set: Experimentation was done on the four SUBJECTS i.e., A, B, C and D of age groups between 25-35 years and their EEG was recorded. SUBJECTS were asked to lift multi cylinder crank shaft weighing 24 kg. Based on the lifting sessions their EEG was recorded and the value of the musculoskeletal disorder was taken based on the graphs variation from the mean. The values of the four inputs were normalized and were taken as the input data. EEG of the subject is shown in Fig. 5 and Table 1.

Training: The following parameters were selected for training the neural network model for MSD forecasting. The slope of the threshold function θ was taken as unity for training. The accelerations and momenta were as follows: for the initial training:

$$\eta = 0.9 \text{ and } \alpha = 0.05$$

For the later part due to large weight variations:

$$\eta = 0.3 \text{ and } \alpha = 0.8$$

Table 2 shows the weights that were generated after the training of the neural net.

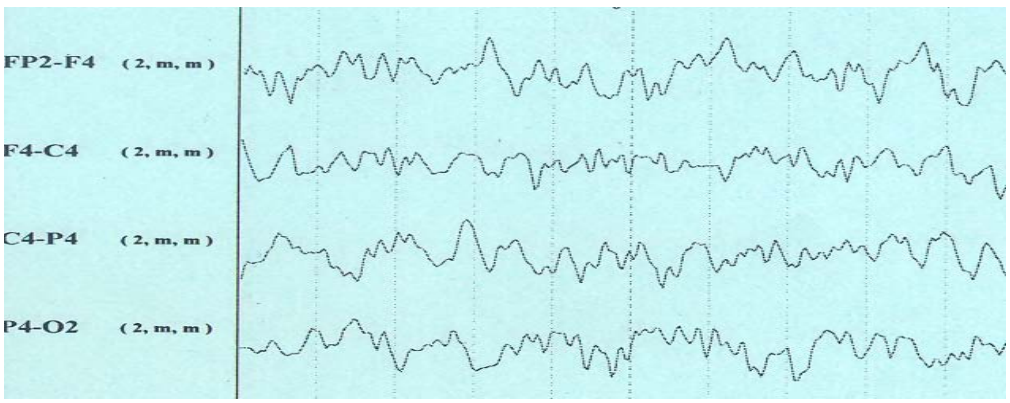

Fig. 5: Electroencephalography of the SUBJECT

Table 1: Input set used to train the ANN in the MSD forecasting application

Inputs for subject			
A	B	C	D
Inputs for the neural network			
INP 1 = 1.7000000000E-01	INP 1 = 1.6000000000E-01	INP 1 = 1.8000000000E-01	INP 1 = 1.7000000000E-01
INP 2 = 1.2000000000E-01	INP 2 = 1.3000000000E-01	INP 2 = 1.4000000000E-01	INP 2 = 1.3000000000E-01
INP 3 = 1.1000000000E-01	INP 3 = 1.2000000000E-01	INP 3 = 1.3000000000E-01	INP 3 = 1.2000000000E-01
INP 4 = 1.2400000000E-01	INP 4 = 1.1200000000E-01	INP 4 = 1.4000000000E-01	INP 4 = 1.2400000000E-01
MSD = 1.4000000000E-01	MSD = 1.3000000000E-01	MSD = 1.6000000000E-01	MSD = 1.4000000000E-01

*The values have been normalized between 0.0 and 1.0

Table 2: Weight matrix obtained after training the ANN: A, between the inputs and the hidden layer; B, between the hidden layer and the output node

A weight matrix for forecast of msd's of subject				
8.9495471863E-01	4.2145860041E-01	1.7711596644E-01	3.2515800330E-01	5.7675013988E-01
7.2296291209E-01	4.9274067765E-01	7.2423366243E-01	3.3912477629E-01	1.7139321365E-01
7.5899705391E-01	8.3196410431E-01	6.5369018413E-01	9.6243140767E-01	7.1389375716E-02
4.5346028759E-01	4.0724331484E-01	5.4722603637E-01	4.4929114683E-01	6.1705591370E 01
9.5671931907E-01	4.8756074933E-01	2.4759020824E-01	7.6533068647E-01	5.4772932110E-01
6.6985348756E-04	2.8159258087E-01	8.7767749808E-01	6.5233359699E-01	4.7497879660E-01
6.3394811437E-01	2.9645921845E-01	2.2413569799E-01	7.4948989563E-02	4.1709592652E-01
4.4033730322E-01	4.5499720895E-01	7.3375416164E-01	2.5863196905E-02	2.5038262624E-01
6.5898919291E-01	1.9208509749E-01	5.8461062097E-02	5.0665933031E-01	6.8347373600E-01
2.0081629000E-02	3.4489177685E-01	2.7319310983E-01	4.1948613316E-01	1.5679098472E-01
2.2652629849E-01	4.9304637579E-01	9.5642652748E-01	7.7106146827E-01	2.6070724991E-01
9.27410756230E-01	7.5830948900E-02	1.4425399716E-01	2.5299009470E-01	8.0100443500E-01
5.5375621774E-01	6.2200050374E-01	3.2993908891E-01	5.8013437764E-01	6.7520808174E-01
7.5071593461E-02	2.1816695323E-01	7.1642117564E-01	6.7004404199E-01	8.8289042050E-01

Table 2: Continue

8.5404316213E-01	4.0867673590E-01	6.9889262255E-01	1.4765973862E-01	9.4730516868E-03
5.5639865337E-01	9.1261324453E-01	1.6673227288E-02	2.8984888797E-01	1.2354309396E-01
5.0491647970E-01	2.8002949065E-01	4.2554848800E-01	4.4534552013E-01	6.0348858122E-01
-2.1973029047E-02	5.4828298896E-01	9.7549404170E-01	-2.1578816794E-02	2.4995045649E-01
8.4922251023E-01	3.9019107184E-01	-1.7947125233E-01	9.7042213979E-01	5.0602813558E-01
3.9348115156E-01	8.2432056933E-01	6.8600658140E-01	4.9168308937E-01	8.3436293343E-01
9.7124016718E-01	1.3002167301E-02	4.8414272302E-01	9.9849633347E-01	2.4115970804E-01
7.4983601895E-02	9.2048044453E-02	9.3755407574E-01	4.8754755409E 01	9.5793937101E-01
6.3456414000E-02	8.2831805270E-01	5.3496806076E-01	4.8075504786E-01	3.7465089376E-01
5.8978635662E-01	9.4269747909E-01	5.2623732577E-02	9.7853371554E-03	4.7583000230E-01
2.3599251634E-01	3.5050321442E-01	7.1805481363E-01	5.6458836351E-02	1.7156749722E-01
6.9757427838E-01	6.7738954315E-01	8.7858041277E-01	8.7042405604E-01	7.7794371132E-02
Weights for links between hidden layer and output layer				
-1.1760996639E-01	-1.8639870613E-01	6.2276096823E-02	-1.0813871042E-01	-3.0106709006E-01
-4.9696132808E-01	3.00411162482E-01	-5.9169606517E-02	-2.4126275419E-01	-5.5944435967E-01
3.2882769900E-01	1.1472194008E-01	9.4075003719E-02	-2.5531086270E-01	2.6722486602E-01
-6.1223439529E-01	-6.3045542694E-01	-4.1726611219E-01	-6.3669579427E-01	5.3502814224E-02
-1.9212661746E-01	2.4975474009E-01	-1.9996350131E-03	-2.6950394976E-01	5.1984532190E-02

Table 3: Forecast obtained using the trained network

Forecast for subject			
A	B	C	D
MSD forecast for 3 different subjects			
Input1 = 1.6700000000E-01	Input1 = 1.7200000000E-01	Input1 = 1.6600000000E-01	Input1 = 1.7500000000E-01
Input2 = 1.3200000000E-01	Input2 = 1.2800000000E-01	Input2 = 1.2800000000E-01	Input2 = 1.2800000000E-01
Input3 = 1.2600000000E-01	Input3 = 1.2900000000E-01	Input3 = 1.2400000000E-01	Input3 = 1.2200000000E-01
input4 =1.2100000000E-01	Input4 = 1.2400000000E-01	Input4 = 1.2800000000E-01	Input4 = 1.2700000000E-01
Output forecast = 1.3400000000E-01	Output forecast = 1.3500000000E-01	Output forecast = 1.2600000000E-01	Output forecast = 1.2800000000E-01

Outputs: Table 3 shows the results obtained for the trained network. The output has been predicted for six different SUBJECTS taken within the age group of 25-35 years. If the value of the respective inputs is given to the neural network then the prediction of musculoskeletal disorder developing within the SUBJECT body is known.

CONCLUSION

In this particular application, the neural network predicted the musculoskeletal disorders for 25-35 years age group grinding machining workers. This neural network as such does not model other machine workers engaged in performing different tasks but they can be used to forecast with a separate network of a similar kind but with a different training. To make this neural net more general and foolproof, extra parameters such as temperature, humidity, etc. can be helpful while training and during prediction.

This study brings to the reader the background and thinking needed by production system engineers for application of neural nets to system operation and control. In the opinion of the reseachers, neural nets will have a great impact on production system operation by providing fault tolerance and massive parallel and distributed processing.

IMPLEMENTATION

Future of neural nets: Neural nets can be applied to a wide variety of problems in production system operation and control but more thought needs to be given to some aspects such as:

- Improvement of the efficiency of the learning procedures as the back-propagation approach is slow
- Maintenance of software and hardware in the control rooms of substations
- Comparison of various neural net structures
- The design of new threshold function structures which simulate more accurately the nonlinearities involved

Knowledge based systems can be used together with neural nets to eliminate the shifts between declarative and reflexive mechanisms.

Research problems in the domain of integration of AI based systems and neural nets may address the following:

- The possibility of exchange of knowledge between an AI based system and neural nets
- A knowledge based system learning from neural net performance

- Division of knowledge between neural nets and knowledge based systems
- Use of symbolic computation in AI for neural nets
- Creation of learning rules for AI based systems from neural nets and vice versa

REFERENCES

Andersen, J.H., J.P. Haahr and P. Frost, 2007. Risk factors for more severe regional musculoskeletal symptoms: A two-year prospective study of a general working population. Int. J. Arthritis Rheum., 56: 1355-1364.

Antonopoulou, M.D., A.K. Alegakis, A.G. Hadjipavlou and C.D. Lionis, 2009. Studying the association between musculoskeletal disorders, quality of life and mental health. A primary care pilot study in rural Crete. BMC Musculoskeletal Disord., 20: 30-38.

Barrero, L.H., J.N. Katz and J.T. Dennerlein, 2009. Validity of self-reported mechanical demands for occupational epidemiologic research of musculoskeletal disorders. Scand. J. Work Environ. Health, 35: 245-260.

Brewer, S., D. van Eerd, B.C. Amick 3rd, E. Irvin and K.M. Daum *et al*., 2006. Workplace interventions to prevent musculoskeletal and visual symptoms and disorders among computer users: A systematic review. J. Occup. Rehabil., 16: 325-358.

Caban-Martinez, A.J., M.P.H.D. Epidemiology and O.M. Student, 2010. Musculoskeletal pain and measures of depression. Gen. Hosp. Psychiatry, 32: 113-118.

Cecchini, M., A. Colantoni, R. Massantini and D. Monarca, 2010. The risk of musculoskeletal disorders for workers due to repetitive movements during tomato harvesting. J. Agric. Saf. Health, 16: 47-53.

Choi, S.D., 2010. Ergonomic assessment of musculoskeletal discomfort of iron workers in highway construction. Work, 36: 47-53.

D'Angelo, R., L. Attaianese, E. Attaianese and P. Mura, 2006. Musculo-skeletal disorders and diseases in the complex metal-mechanical industry. Med. Lav. Ergon., 28: 182-184.

Deeney, C. and L. O'Sullivan, 2009. Work related psychosocial risks and musculoskeletal disorders: Potential risk factors, causation and evaluation methods. Work, 34: 239-248.

Ebron, S., D.L. Lubkeman and M. White, 1990. A neural network approach to the detection of incipient faults on power distribution feeders. IEEE Trans., 5: 905-914.

Fujiwara, R., T. Sakaguchi, Y. Kohno and H. Suzuki, 1986. An intelligent load flow engine for power system planning. IEEE Trans., 1: 302-307.

Fukui, C. and J. Kawakami, 1986. An expert system for fault section estimation using information from protective relays and circuit breakers. IEEE Trans., 1: 83-90.

Jansen, J.J. and H.B. Puttgen, 1987. ASDEP: An expert system for electric power plant design. IEEE Expert, 2: 56-66.

Liu, C.C. and K. Tomsovic, 1986. An expert system assisting decision-making of reactive power/voltage control. IEEE Trans., 1: 195-201.

Rahman, S. and S. Bhatnagar, 1988. An expert system based algorithm for short term load forecast. IEEE Trans., 3: 392-398.

Sakaguchi, T. and K. Matsumoto, 1983. Development of a knowledge based system for power system restoration. IEEE Trans. Power Apparatus Syst., 102: 320-329.

Santosa, N.I. and O.T. Tan, 1990. Neural net based real time control of capacitors installed on distribution systems. IEEE Trans., 5: 266-271.

Sobajic, D.J. and Y.H. Pao, 1989. Artificial neural net based dynamic security assessment for systems. IEEE Trans., 4: 220-226.

Talukdar, N.S., E. Cardozo and T. Perry, 1986. The operator's assistant an intelligent, expandable program for power system trouble analysis. IEEE Trans., 1: 182-187.

Tomsovic, K., C.C. Liu, P. Ackerman and S. Pope, 1986. An expert system as a dispatcher's aid for the isolation of line section faults. Proceedings of IEEE PES Transmission and Distribution Conference, Nov. 12, University of Washington Seattle USA., pp: 736-743.

Wang, P.C., D. Rempel, R. Harrison, J. Chan and B. Ritz, 2007. Work-organizational and personal factors associated with upper body musculoskeletal disorders among sewing machine operators. Occup. Environ. Med., 64: 806-813.

Wiesinger, B., H. Malker, E. Englund and A. Wenman, 2007. Back pain in relation to musculoskeletal disorders in the jaw-face: A matched case-control study. Pain, 131: 311-319.

Wollenberg, F.B., 1986. Feasibility study for an energy management system intelligent alarm processor. IEEE Trans., 1: 241-247.