

Myoelectric signals pattern recognition for intelligent functional operation of upper-limb prosthesis

N Chaiyaratana¹, A M S Zalzal² and D Datta³

^{1,2}Robotics Research Group, Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK

³Disablement Services Centre, Northern General Hospital, Herries Road, Sheffield, S5 7AT, UK

^{1,2}*rrg@sheffield.ac.uk*

ABSTRACT

This paper presents a comparative study of the classification accuracy of myoelectric signals using multilayer perceptron with back-propagation algorithm and radial-basis functions networks. The myoelectric signals considered are used to classify four upper-limb movements which are elbow bending, elbow extension, wrist pronation and wrist supination. The network structure for multilayer perceptron is a fully connected one, while the structures used in radial-basis functions network are both fully connected and partially connected. Two learning strategies are used for training radial-basis networks, namely supervised selection of centres and fixed centres selected at random. The results suggest that radial-basis function network with fixed centres can generalise better than the others without requiring extra computational effort.

Keywords: multilayer perceptron, myoelectric signal, pattern recognition, radial-basis function network, upper-limb prosthesis

1. INTRODUCTION

Myoelectric signals are the signals which are generated by muscles when they contract. They have been used in various aspects of medical and biomedical engineering, for example, the diagnosis of neuromuscular disease such as polymyositis (Kumaravel and Kavitha, 1994). One of the most common uses of myoelectric signals is for controlling prosthesis manipulators (Scott and Parker, 1988). Myoelectric control prostheses have received widespread use as devices for individuals with amputations or congenitally deficient upper limbs (Scott and Parker, 1988), and many systems are now available commercially to control a single device such as a hand, elbow and wrist. Each myoelectric signal, generated by the muscle in performing different task, has a unique pattern which contains the information about the direction of movement and the speed of action. To be able to control the prosthesis successfully, the microprocessor, which is part of the prosthesis, must be able to classify these patterns accurately; this results in a pattern recognition problem. The myoelectric signal is essentially a one-dimensional pattern, and the methods and algorithms developed for one-dimensional pattern recognition can be applied to this analysis. The information extracted from the myoelectric signal, represented in a feature vector, is chosen to minimise the control error. To achieve this, a feature set which maximally separates the desired output classes must be chosen. The need for fast response of the prosthesis limits the period over which these features can be extracted.

A number of researchers have discussed the possibility of using neural network in solving myoelectric signal pattern recognition problem. Most research had been carried out by using a multilayer perceptron which contains one hidden layer in conjunction with back-propagation algorithm, except the work by Costa and Gander (1993) which had been carried out by using a multilayer perceptron with two hidden layers. Myoelectric signals can be drawn from various locations on subject's body, an application dependent criterion. For example, the signals from flexor digitorum superficialis are used in classification of finger movements (Hiraiwa et al., 1989) or the signals from biceps and triceps brachii are used to determine the arm movements (Hudgins et al., 1993; Ito et al., 1991; Kuruganti et al., 1995). The control signal can be derived from a single myoelectric channel (Costa and Gander, 1993; Hudgins et al., 1993; Karlik et al., 1994; Kelly et al., 1990), or from multichannel such as two channels (Kuruganti et al., 1995; Yeh et al., 1993), four channels (Ito et al., 1991) or five channels (Doerschuk et al., 1983). Using a single channel myoelectric signal would result in less complexity in neural network structure. However, using multichannel signals makes the positions

of electrodes become less critical to the experiment and increase the classification accuracy (Kuruganti et al., 1995). Most of the previous research is concerned with the arm movements classification task. Usually, the experiments are carried out in two possible ways: by exposing the subject's arm movements to a weight constraint (Costa and Gander, 1993) or by allowing the subject to perform the movement naturally (Hudgins et al., 1993; Kuruganti et al., 1995).

To control n functions in the prosthesis requires n unique patterns of muscle contraction. The control schemes have been based almost entirely on the discriminant approach to pattern recognition, in which each pattern is described by a set of features. Features set can be obtained by using various methods. For example, the parameters of some stochastic models such as an autoregressive (AR) model or autoregressive moving average (ARMA) model can be used as features set. A number of research has been done by utilising AR model (Doerschuk et al., 1983; Graupe et al., 1985; Karlik et al., 1994; Kelly et al., 1990; Kiryu et al., 1994; Latwesen and Patterson, 1993; Merletti and Lo Conte, 1995; Zardoshti-Kermani et al., 1995). All of these works are based upon the research by Graupe and Kline (1975) which involves modelling myoelectric signals as ARMA models. The later research has shown that the use of AR model is sufficient for modelling myoelectric signals. Graupe et al. (1985) have proved that a myoelectric signal within a period of 0.2-0.3 second can be model as a 4th order AR model.

Various methods have been used to obtained the parameters of AR model. One of the most common methods which has been used is recursive least squares (RLS) or sequential least squares (SLS) (Graupe et al., 1985; Kiryu et al., 1994; Latwesen and Patterson, 1993; Zardoshti-Kermani et al., 1995). This method has been proved to be very reliable and has capability in dealing with noisy myoelectric signal. The extension of recursive least squares algorithm to accommodate multivariable autoregressive model is proposed by Doerschuk et al. (1983). This extension is done such that all parameters of the AR models from different channels of the signals can be computed simultaneously. Other methods which have been used to obtained the parameters of AR model are the application of discrete Hopfield network (Kelly et al., 1990) and the PARCOR algorithm (Karlik et al., 1994). Unlike recursive least squares algorithm which is based on the principle of minimising the error between the estimate value and the actual value of each signal in each iteration, discrete Hopfield network is used under the principle of multivariable optimisation. One advantage in using discrete Hopfield network is that the convergence rate of computation by discrete Hopfield network is higher than that of the recursive least squares algorithms (Kelly et al., 1990).

Some other characteristics of myoelectric signal can also be used as features for the neural network input. For example, the time domain characteristics of the signal such as mean absolute value, mean absolute value slope, zero crossings, slope sign changes and waveform length have been used by Hudgins et al. (1993) and Kuruganti et al. (1995). Other time domain characteristics such as the turns and mean amplitude and normalised time average have been utilised by Yeh et al. (1993) and Ito et al. (1991), respectively. An advantage which can be gained by using time domain characteristics instead of parameters of stochastic models is the complexity reduction in feature extraction process. One drawback is the increase in input layer of the network. Parameters from fast Fourier transform (FFT) can also be used as the features set (Hiraiwa et al., 1989). Del Boca and Park (1994) have extracted the signal features through Fourier analysis. Unlike any other previous research, the features are then unsupervised clustered by using fuzzy c-mean algorithm before they are presented to the neural network for pattern recognition. This method has been proved to be very efficient for real-time operation.

The assumption that myoelectric signal is a stochastic signal has always been made when the parameters of AR model, time domain characteristics or results from Fourier analysis are used as features. Costa and Gander (1993) have proposed a new assumption that myoelectric signal should be treated as a chaotic signal. This assumption leads to a totally different type of features that can be used as input to the neural network. They use the Poincaré sections of the chaotic myoelectric signal as the features set. They also suggest the use of their result in automated myopathy diagnosis.

This paper is concerned with the use of neural networks in myoelectric control system. The myoelectric data used in the experiments are the data obtained from the journal article by Graupe et al. (1985). These data are the autoregressive (AR) model parameters of the single-site myoelectric signals obtained by measuring the signals from the location between biceps and triceps with sampling period of 2 ms. The location of the electrode is expected to be the place where the maximum cross-talk between signals occurs. Four contraction types are concerned in the experiments. These contraction types are elbow bending (EB) or elbow flexion, elbow extension (EE), wrist pronation (WP) and wrist supination (WS) (Graupe et al., 1985). The AR model parameters are used as feature for the feature sets. These feature sets are divided into two groups, one for training the networks, another for testing. The feature sets are used as input to two types of neural network. They are a multilayer perceptron with back-propagation algorithm network and a radial-basis function network. Comparative study on myoelectric signal classification accuracy performance between these two types of network is the major objective in this paper. This objective rises from the fact that nearly all research in this area had been carried out by using multilayer perceptron which contains one hidden layer in conjunction with back-propagation algorithm. The experiments which had been done can be divided into two sections. They are determination

on multilayer perceptron structure and determination on learning strategies and structure topologies in radial-basis function network.

2. DETERMINATION ON MULTILAYER PERCEPTRON STRUCTURE

The cost function of multilayer perceptron is given in Eq. (1),

$$\varepsilon(n) = \frac{1}{2} \sum_{k=1}^{N_o} e_k^2(n) \quad (1)$$

where $\varepsilon(n)$ is the instantaneous cost function at iteration n ,
 $e_k(n)$ is the error from output node k at iteration n and
 N_o is the number of output nodes.

The error from each output node can be defined as follows,

$$e_k(n) = d_k(n) - y_k(n) \quad (2)$$

where $d_k(n)$ is the desired response of output node k at iteration n and
 $y_k(n)$ is the output of output node k at iteration n .

Haykin (1994) gives summary on back-propagation algorithm as follows.

1. *Initialisation.* Set all the weights and threshold levels of the network to small random numbers that are uniformly distributed.
2. *Forward Computation.* Let a training example be denoted by $[\mathbf{x}(n), \mathbf{d}(n)]$, with the input vector $\mathbf{x}(n)$ applied to the input layer and the desired response vector $\mathbf{d}(n)$ presented to the output layer. The net internal activity level $v_j^{(l)}(n)$ for neuron j in layer l is given by

$$v_j^{(l)}(n) = \sum_{i=0}^p w_{ji}^{(l)}(n) y_i^{(l-1)}(n) \quad (3)$$

where $y_i^{(l-1)}(n)$ is the signal from neuron i in the previous layer $l-1$ at iteration n and
 $w_{ji}^{(l)}(n)$ is the weight of neuron j in layer l that is connected to neuron i in layer $l-1$ at iteration n .

For $i = 0$, we have

$$y_0^{(l-1)} = -1 \quad (4)$$

and

$$w_{j0}^{(l)}(n) = \theta_j^{(l)}(n) \quad (5)$$

where $\theta_j^{(l)}(n)$ is the threshold applied to neuron j in layer l .

With the use of a logistic function for the sigmoidal non-linearity, the output of neuron j in layer l is given by

$$y_j^{(l)}(n) = \frac{1}{1 + \exp(-v_j^{(l)}(n))} \quad (6)$$

If neuron j is in the first hidden layer (i.e., $l = 1$), set

$$y_j^{(0)}(n) = x_j(n) \quad (7)$$

where $x_j(n)$ is the j th element of input vector $\mathbf{x}(n)$. If neuron j is in the output layer (i.e., $l = L$), set

$$y_j^{(L)}(n) = o_j(n) \quad (8)$$

The error can be computed as follows,

$$e_j(n) = d_j(n) - o_j(n) \quad (9)$$

where $d_j(n)$ is the j th element of the desired response vector $\mathbf{d}(n)$.

4. *Backward Computation.* Compute the local gradients (δ) of the network by progressing backward, layer by layer. For neuron j in output layer L , the local gradient is given by

$$\delta_j^{(L)}(n) = e_j^{(L)}(n) o_j(n) [1 - o_j(n)]. \quad (10)$$

For neuron j in hidden layer l , the local gradient is given by

$$\delta_j^{(l)}(n) = y_j^{(l)}(n) [1 - y_j^{(l)}(n)] \sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n). \quad (11)$$

The weight of the network in layer l can be adjusted according to the generalised delta rule as follows,

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha [w_{ji}^{(l)}(n) - w_{ji}^{(l)}(n-1)] + \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n) \quad (12)$$

where η is the learning rate parameter and
 α is the momentum constant.

In this study, there are three parameters which are needed to be considered. They are learning rate parameter, momentum constant and number of hidden nodes. Previous research has shown that only one hidden layer is sufficient for this application. The ranges of value for both learning rate parameter and momentum constant are typically lied between 0 and 1. These two parameters cannot be chosen independently. Three observations on choices of learning rate parameter and momentum constant are given by Haykin (1994) as follows.

- A smaller learning rate parameter leads to slower convergence. The search with smaller learning rate parameter can cover more error surface than the search with larger learning rate parameter.
- For learning rate parameter approaching zero, the use of momentum constant with the value near one will increase the speed of convergence. On the other hand, for learning rate parameter approaching one, the use of momentum constant with the value near zero will ensure the stability of learning.
- The use of large value of learning rate parameter in conjunction with large momentum constant can leads to oscillations in the mean-squared error during learning process and a high value of final mean-squared error (Haykin, 1994).

From these three observations, learning rate parameter are chosen to be 0.1 and momentum constant is chosen to be 0.9. It will lead to a good coverage of error surface and fast convergence. The number of hidden nodes is determined via experiment. The testing range is chosen to be from 10 to 15. Classification results are shown in Fig. 1.

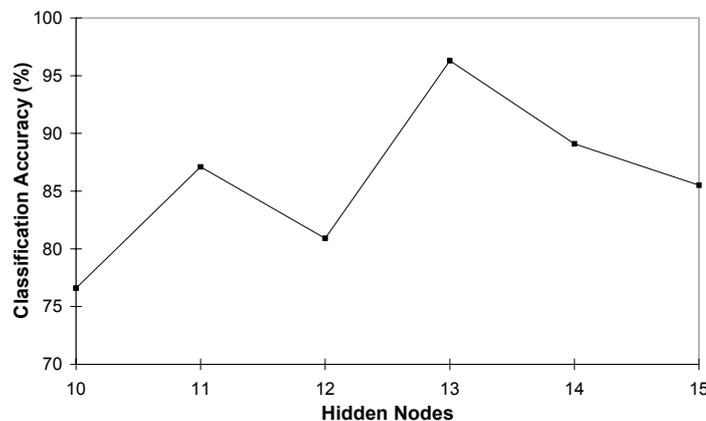


Figure 1. Classification results of multilayer perceptron with back-propagation algorithm.

Experiment results show that the optimum number of hidden nodes is 13 with the highest classification accuracy of 96.3 %.

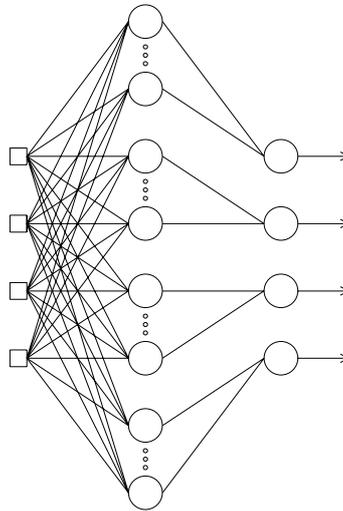
3. DETERMINATION ON LEARNING STRATEGIES AND STRUCTURE TOPOLOGIES IN RADIAL-BASIS FUNCTION NETWORK

Two learning strategies are used in this study. These learning strategies are supervised selection of centres of radial-basis function and fixed centres of radial-basis function selected at random.

3.1 Supervised Selection of Centres

In this learning strategy, all free parameters in the network undergo a supervised learning process. These free parameters are the weights connecting the hidden layer to the output layer, the centres of radial-basis function in the hidden layer and the inverse covariance matrices of the radial-basis function. This is the most generalised learning strategy for radial-basis function network. The algorithm used in the learning process is derived from the method of steepest descent. In this study, all learning rate parameters are set at 0.1. Two topologies of the network are used under this learning strategy. These topologies are partially connected network and fully connected network.

3.1.1 Partially Connected Network. In this topology, hidden nodes and output nodes are partially connected. Since there are four patterns in concerned, the number of output nodes in this case is four. Each hidden node will be connected to only one output node. Output from the first output node will be used to determine whether input pattern belongs to class 1 or not. Other output nodes work in similar fashion. Combination of result from all output nodes will be used to determine the class of input pattern. The topology of the network is shown in Fig. 2.



Input layer Hidden layer Output layer

Figure 2. Partially connected radial-basis function network.

This network topology can be viewed as four radial-basis function networks with one output node, sharing the same input working in co-operation. In this study, each part of network that is fully connected to a single output node is called a sub-network. In this case, there will be four sub-networks.

The cost function for each sub-network is defined as follows,

$$\varepsilon_k(n) = \frac{1}{2} e_k^2(n), \quad k = 1, 2, 3, 4 \quad (13)$$

where $\varepsilon_k(n)$ is the instantaneous cost function of sub-network k at iteration n and $e_k(n)$ is the error from output node k at iteration n .

The error from each output node is defined as follows,

$$e_k(n) = d_k(n) - \sum_{i=1}^{M_k} w_{ki}(n) G(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}), \quad k = 1, 2, 3, 4 \quad (14)$$

where $d_k(n)$ is the desired output of output node k at iteration n ,

$w_{ki}(n)$ is the weight of sub-network k which connects to the i th Green's function in the same sub-network,

$G(\cdot)$ is the Green's function and
 M_k is the number of Green's function in sub-network k .

Define Green's function in radial-basis function network as follows,

$$G(\|\mathbf{x} - \mathbf{t}_i\|_{C_i}) = G((\mathbf{x} - \mathbf{t}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \mathbf{t}_i)) \quad (15)$$

where \mathbf{x} is the input pattern,
 \mathbf{t}_i is the i th centre of radial-basis function network and
 $\boldsymbol{\Sigma}_i^{-1}$ is the i th inverse covariance matrix.

In this study, Green's function is chosen to be Gaussian function which can be shown in Eq. (16),

$$G(\|\mathbf{x} - \mathbf{t}_i\|_{C_i}) = \exp(-(\mathbf{x} - \mathbf{t}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \mathbf{t}_i)). \quad (16)$$

Steepest descent algorithm is used to determine algorithm for adapting the value of weights, centres and inverse covariance matrices. This leads to the use of partial derivatives of the cost function with respect to these free parameters. The adapting formula would be in the form shown in Eq. (17),

$$p(n+1) = p(n) - \eta \frac{\partial \varepsilon(n)}{\partial p(n)} \quad (17)$$

where $p(n)$ is free parameter at iteration n and
 η is learning rate parameter.

These partial derivatives are shown in Eqs (18-20),

$$\frac{\partial \varepsilon_k(n)}{\partial w_{ki}(n)} = -e_k(n) G(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}), \quad k = 1, 2, 3, 4; i = 1, 2, 3, \dots, M_k \quad (18)$$

$$\frac{\partial \varepsilon_k(n)}{\partial \alpha_i(n)} = 2w_{ki}(n)e_k(n)G'(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i})\boldsymbol{\Sigma}_i^{-1}(n)[\mathbf{x}(n) - \mathbf{t}_i(n)], \quad k = 1, 2, 3, 4; i = 1, 2, 3, \dots, M_k \quad (19)$$

$$\frac{\partial \varepsilon_k(n)}{\partial \boldsymbol{\Sigma}_i^{-1}(n)} = -w_{ki}(n)e_k(n)G'(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i})\mathbf{Q}_i(n), \quad k = 1, 2, 3, 4; i = 1, 2, 3, \dots, M_k \quad (20)$$

where $G'(\cdot)$ is the first derivative of the Green's function $G(\cdot)$ with respect to its argument and

$$\mathbf{Q}_i(n) = [\mathbf{x}(n) - \mathbf{t}_i(n)][\mathbf{x}(n) - \mathbf{t}_i(n)]^T. \quad (21)$$

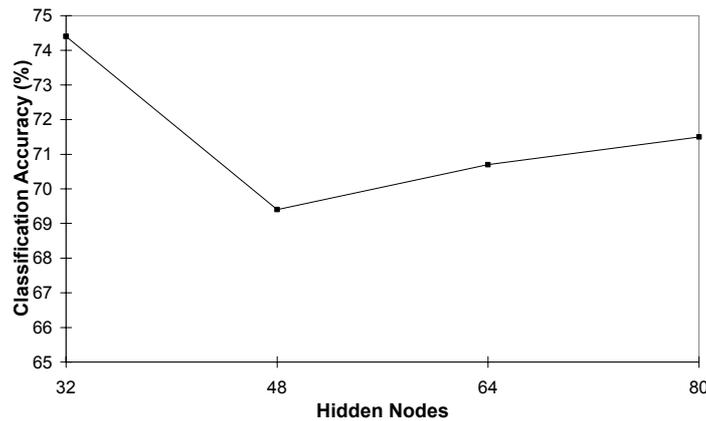


Figure 3. Classification results of partially connected radial-basis function network with supervised selection of centres (centres in each sub-network are selected from all input classes).

Two sets of experiments are conducted under this structure type of network. In the first set of experiment, the centres in each sub-network are chosen from all four pattern classes with equal number of centres from each pattern. The number of hidden nodes in each sub-network is chosen to be 8, 12, 16 and 20. In other words, the total number of hidden nodes

in the entire network in each case would be 32, 48, 64 and 80, respectively. The classification results are shown in Fig. 3.

Experiment results show that the optimum number of hidden nodes in this case is 32 with the highest classification accuracy of 74.4 %.

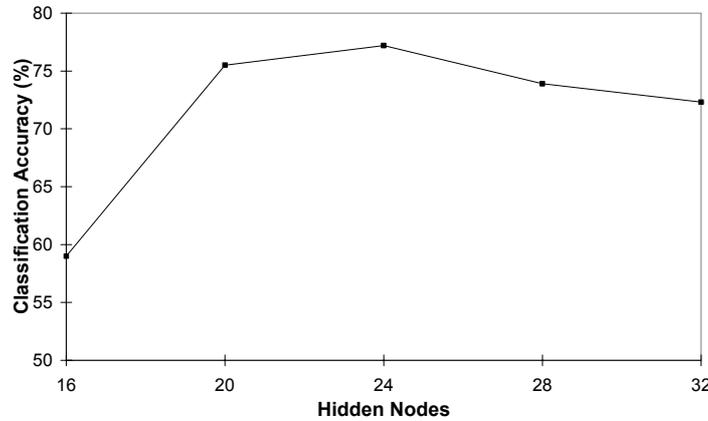


Figure 4. Classification results of partially connected radial-basis function network with supervised selection of centres. (centres in each sub-network are selected from one input class only).

In the second set of experiment, the centres in each sub-network are chosen from only one pattern class. The number of hidden nodes in each sub-network is chosen to be in the range of 4 to 8. This is corresponding to the total hidden nodes of 16, 20, 24, 28, and 32. The classification result is shown in Fig. 4. Experiment results show that the optimum number of hidden nodes in this case is 24 with the highest classification accuracy of 77.2 %.

3.1.2 Fully Connected Network. In this topology, hidden layer is fully connected to output layer. The cost function in this case is given in Eq. (22),

$$\varepsilon(n) = \frac{1}{2} \sum_{k=1}^{N_o} e_k^2(n) \quad (22)$$

where $\varepsilon(n)$ is the instantaneous cost function at iteration n ,
 $e_k(n)$ is the error from output node k at iteration n and
 N_o is the number of output nodes.

The error from each output node is defined as follows,

$$e_k(n) = d_k(n) - \sum_{i=1}^M w_{ki}(n) G(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}), \quad k = 1, 2, 3, 4 \quad (23)$$

where M is the number of Green's function in the network.

Steepest descent algorithm is used to adapt free parameters in the network. Partial derivatives of free parameters are given in Eqs (24-26),

$$\frac{\partial \varepsilon(n)}{\partial w_{ki}(n)} = -e_k(n) G(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}), \quad k = 1, 2, 3, 4; i = 1, 2, 3, \dots, M \quad (24)$$

$$\frac{\partial \varepsilon(n)}{\partial \mathbf{a}_i(n)} = 2 \sum_{k=1}^{N_o} w_{ki}(n) e_k(n) G'(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}) \mathbf{\Sigma}_i^{-1}(n) [\mathbf{x}(n) - \mathbf{t}_i(n)], \quad i = 1, 2, 3, \dots, M \quad (25)$$

$$\frac{\partial \varepsilon(n)}{\partial \mathbf{\Sigma}_i^{-1}(n)} = - \sum_{k=1}^{N_o} w_{ki}(n) e_k(n) G'(\|\mathbf{x}(n) - \mathbf{t}_i(n)\|_{C_i}) \mathbf{Q}_i(n), \quad i = 1, 2, 3, \dots, M. \quad (26)$$

For this structure type of network, all learning rate parameters are set to 0.1 as well. The number of hidden nodes is chosen to be 8, 12, 16 and 20. The centres are selected from all pattern classes with equal number of centres from each pattern. The classification results are shown in Fig. 5.

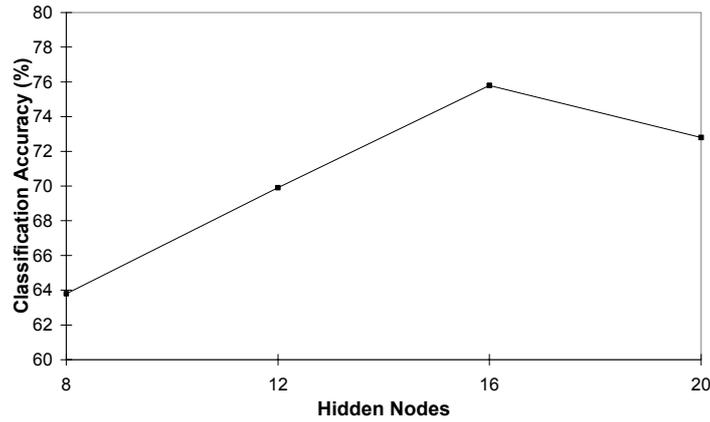


Figure 5. Classification results of fully connected radial-basis function network with supervised selection of centres.

From experiment results, the optimum number of hidden nodes is 16 with the highest classification accuracy of 75.8 %.

3.2 Fixed Centres Selected at Random

In this learning strategy, the centres of radial-basis function are fixed. These centres are selected at random from all pattern classes with equal number of centres from each pattern. In this case, the network is fully connected. The Green's function is defined as shown in Eq. (27),

$$G(\|\mathbf{x} - \mathbf{t}_i\|^2) = \exp\left(-\frac{M}{d^2}\|\mathbf{x} - \mathbf{t}_i\|^2\right) \quad (27)$$

where M is the number of centres and d is the maximum distance between the chosen centres.

The only parameter that undergoes supervised learning is the weight in the network. Haykin (1994) suggests that one straightforward procedure for finding the weight of the network is to use pseudoinverse method. The weight solution is given by Eq. (28),

$$\mathbf{w} = \mathbf{G}^+ \mathbf{d} \quad (28)$$

where \mathbf{d} is the desired response vector in the training set and \mathbf{G}^+ is the pseudoinverse matrix of matrix \mathbf{G} .

The matrix \mathbf{G} is defined in Eqs (29-30),

$$\mathbf{G}^+ = \{g_{ni}\} \quad (29)$$

$$g_{ni} = \exp\left(-\frac{M}{d^2}\|\mathbf{x}(n) - \mathbf{t}_i\|^2\right), \quad n = 1, 2, 3, \dots, N; i = 1, 2, 3, \dots, M \quad (30)$$

where N is the total number of training input pattern.

One way of solving Eq. (28) is to use recursive least square (RLS) algorithm. In this experiment, the number of hidden nodes is chosen to be 8, 12, 16, and 20. The classification results are shown in Fig. 6.

All experiment results in this case are over 99 % classification accuracy. The optimum number of hidden nodes is 8 with classification accuracy of 99.0 %.

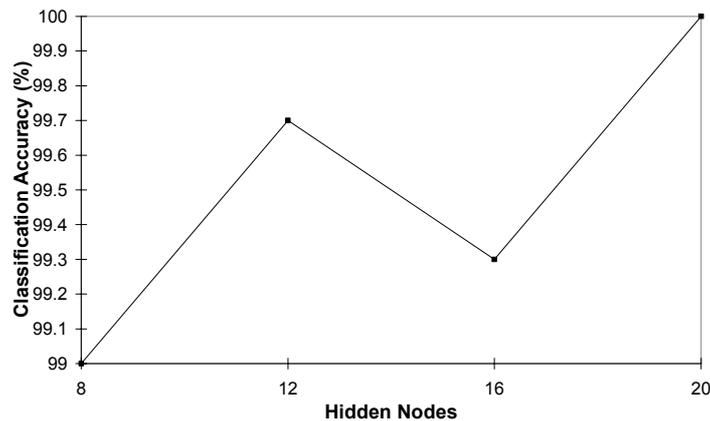


Figure 6. Classification results of radial-basis function network with fixed centres selected at random.

4. CONCLUSIONS

Radial-basis function network with fixed centre selected at random has the highest classification accuracy. This is true even with the number of hidden nodes as small as 8. One possible reason that radial-basis function network with supervised selection of centres cannot generalise as well as the one with fixed centres is that the centres of network are clustered together during training process. These centres cannot distribute themselves to cover the necessary pattern space for correct classification. This is a normal effect which is caused by steepest descent algorithm. Multilayer perceptron still be able to generalise with very high classification accuracy. Compare with radial-basis function network with supervised selection of centres, multilayer perceptron still be a better choice in this application. Since radial-basis function network with fixed centres does not require a bigger network structure or computational effort to outmatch multilayer perceptron, it should be considered for the possibility of microprocessor implementation in prosthesis devices.

5. REFERENCES

- J D Costa and R E Gander (1993), MES classification using artificial neural networks and chaos theory, *Proc. Intl. Joint Conf. Neural Net.*, **3**, Nagoya, Japan, pp. 2243-2246.
- A Del Boca and D C Park (1994), Myoelectric signal recognition using fuzzy clustering and artificial neural networks in real time, *Proc. IEEE Intl. Conf. Neural Net.*, **5**, Orlando, FL, pp. 3098-3103.
- P C Doerschuk, D E Gustafson and A S Willsky (1983), Upper extremity limb function discrimination using EMG signal analysis, *IEEE Trans. Biomed. Eng.*, **30**, *1*, pp. 18-29.
- D Graupe and K W Cline (1975), Functional separation of EMG signal via ARMA identification methods for prosthesis control purposes, *IEEE Trans. Sys. Man Cyber.*, **5**, *2*, pp. 252-259.
- D Graupe, J Salahi and D Zhang (1985), Stochastic analysis of myoelectric temporal signatures for multifunction single-site activation of prostheses and orthoses, *J. Biomed. Eng.*, **7**, *1*, pp. 18-29.
- S Haykin (1994), *Neural Networks: A Comprehensive Foundation*, Macmillan, New York.
- A Hiraiwa, K Shimohara and Y Tokunaga (1989), EMG pattern analysis and classification by neural network, *Proc. IEEE Intl. Conf. Sys. Man Cyber.*, **3**, Cambridge, MA, pp. 1113-1115.
- B Hudgins, P A Parker and R N Scott (1993), A new strategy for multifunction myoelectric control, *IEEE Trans. Biomed. Eng.*, **40**, *1*, pp. 82-94.
- K Ito, T Tsuji, A Kato and M Ito (1991), Limb-function discrimination using EMG signals by neural network and application to prosthetic forearm control, *IEEE Intl. Joint Conf. Neural Net.*, Singapore, pp. 1214-1219.
- B Karlik, H Pastaci and M Korürek (1994), Myoelectric neural networks signal analysis, *Proc. 7th Mediter. Electrotech. Conf.*, **1**, Antalya, Turkey, pp. 262-264.
- M Kelly, P A Parker and R N Scott (1990), The application of neural networks to myoelectric signal analysis: A preliminary study, *IEEE Trans. Biomed. Eng.*, **37**, *3*, pp. 221-227.
- T Kiryu, C J De Luca and Y Saitoh (1994), AR modelling of myoelectric interference signals during a ramp contraction, *IEEE Trans. Biomed. Eng.*, **41**, *11*, pp. 1031-1038.

- N Kumaravel and V Kavitha (1994), Automatic diagnosis of neuro-muscular diseases using neural network, *Biomed. Sci. Instrum.*, **30**, pp. 245-250.
- U Kuruganti, B Hudgins and R N Scott (1995), Two-channel enhancement of a multifunction control system, *IEEE Trans. Biomed. Eng.*, **42**, 1, pp. 109-111.
- A Latwesen and P E Patterson (1994), Identification of lower arm motions using the EMG signals of shoulder muscles, *Med. Eng. Phy.*, **16**, 2, pp. 113-121.
- R Merletti and L R Lo Conte (1995), Advances in processing of surface myoelectric signals: Part 1, *Med. Biolog. Eng. Comp.*, **33**, pp. 362-372.
- R N Scott and P A Parker (1988), Myoelectric prostheses: State of the art, *J. Med. Eng. Tech.*, **12**, pp. 143-151.
- E C Yeh, W P Chung, R C Chan, and C C Tseng (1993), Development of neural network controller for below-elbow prosthesis using single-chip microcontroller, *Biomed. Eng. Appl. Bas. Comm.*, **5**, 3, pp. 340-346.
- M Zardoshti-Kermani, B C Wheeler, K Badie and R M Hashemi (1995), EMG feature selection for movement control of a cybernetic arm, *Cyber. Sys. Intl. J.*, **26**, 2, pp. 189-210.