Neural Processing Letters **00:** 00–00, 2004.

Visual – Motor Coordination using a Quantum Clustering based Neural Control Scheme

NIMIT KUMAR and LAXMIDHAR BEHERA
Department of Electrical Engineering, Indian Institute of Technology, Kanpur, UP 208016, India. e-mails: nimit.kumar@ieee.org, lberhera@iitk.ac.in

Abstract. Visual – Motor Coordination is a problem considered analogous to the hand-eye coordination in biological systems. In this work we propose a novel approach to this problem using Quantum Clustering and an extended Kohonen's Self-Organizing Feature Map (K-SOFM). This facilities the use of the method im varying workspaces by considering the joint angles of the robot arm. Unlike previous work, where a fixed topology for the input space is considered, the proposed approach determines a topology as the workspace varies. Quantum Clustering is a method which constructs a scale-space probability function and uses the Schroedinger equation and its lowest eigenstate to obtain a potential whose minimum gives the cluster centers. It transforms the input space into a Hilbert space, where it searches for its minimum. The motivation of this work is to identify the implicit relationship existing between the end-effector positions and the joint angles through Quantum Clustering and Neural Network methods to fine-tune the system to correctly identify the mapping.

Key words. Hilbert spaces, quantum clustering, scale space clustering, schrodinger equation, visual – motor coordination

1. Introduction

This work introduces a new method based on Quantum Clustering for the visual – motor coordination of a Robotic Arm. Previous works in this area [4–6, 8, 11] gave a 'black-box' like system with high training time and fixed topological space. Walter and Schulten [11] extended the Self-Organizing Feature Map Algorithm to give an improved result and added flexibility in the robot parameters. However, the fixed topology of the feature map and strong dependence on the training parameters could not be avoided. Behera and Kirubanandan [1] introduced the idea of a hybrid approach, based on Kohonen's Self-Organizing Maps with an online training scheme and fixed topology. To solve the issue of fixed topology and to involve the implicit relationship between the input space and the output space, we propose a novel technique using Quantum Clustering (QC). The QC was originally proposed by Horn and Gottlieb [3] and is used to determine the necessary topology of the network for the workspace.

2. Quantum Clustering

The QC is a non-parametric clustering technique [7] based on the scale-space clustering algorithm [9]. It uses the Gaussian kernel to generate the Parzen-window

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

estimator [2]. Given a set of N data points, $\mathbf{x}_1, \ldots, \mathbf{x}_N$, the task is to estimate the probability density function $\Psi(\mathbf{x})$ using the Parzen-window estimator. The estimator is constructed by associating with each of the N data points a Gaussian defined as in Equation (1). In this sense it belongs to a more general class of non-parametric method – the one using Kernels, here a Gaussian kernel.

$$\Psi(\mathbf{x}) = \sum_{i=1}^{N} e^{-(\mathbf{x}-\mathbf{x}_i)/2\sigma^2}. \tag{1}$$

The maxima of the function $\Psi(\mathbf{x})$ have been shown to occur at the cluster centers [9], where $1/2\sigma^2$ is the scale of the probability estimator. The function $\Psi(\mathbf{x})$ is the Schroedinger Wave Function and performs a nonlinear transformation of the input space into a Hilbert space. It can be seen that as $\Psi(\mathbf{x})$ is a Gaussian kernel, it is positive definite and hence a standard inner product defined in the transformed space is also positive definite. That is, by Mercer's theorem [10], the nonlinear transformation $\Psi(\mathbf{x})$ defines a Hilbert space.

2.1. IMPLEMENTING QUANTUM CLUSTERING

The motivation is to search for a Hamiltonian for which Ψ is an eigenstate and a ground state of the operator H.

$$H\Psi = \left(-\frac{1}{2\sigma^2}\nabla^2 + V(\mathbf{x})\right)\Psi = E\Psi. \tag{2}$$

Equation (2) is a rescaled form of the Schrodinger Equation in quantum mechanics, with Ψ denoting the eigenstate, H the Hamiltonian, V the potential energy and E is the energy eigenvalue. It must be noted that σ is the only parameter in the equation. σ deduces the correct clustering of the space as it controls the width of the Parzen-window [9]. The parameter σ can be controlled in a way such that the technique yields the relevant number of clusters.

Given Ψ, one can solve Equation (2) for V:

$$V(\mathbf{x}) = E + \frac{(1/2\sigma^2)\nabla^2\Psi}{\Psi}. \tag{3}$$

If V is positive definite, that is $V \geq 0$, E may be defined as in Equation (4).

$$E = -\min\frac{(1/2\sigma^2)\nabla^2\Psi}{\Psi}. \tag{4}$$

As Ψ is positive definite, it follows that it has no node and hence E is the minimal eigenvalue of V. The lowest possible eigenvalue occurs for the harmonic potential in which case $E = \mathrm{d}/2$. This leads to the inequality (5).

$$0 < E \leq \frac{1}{2}d. \tag{5}$$

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

QC, thus works by determining the cluster centers, with the data points assigned to a cluster based on the Euclidean distance. In the problem of visual-motor coordination, we are interested in locating the cluster centers and hence the use of QC is preferred.

2.2. WHY USE QUANTUM CLUSTERING?

In QC, the cluster centers are obtained while searching for the minima of the potential functions. In statistical clustering methods, the distribution function is maximized. In QC the wave function is interpreted as the probability density function itself. However, determining the potential minima is more robust than locating the maxima of the distribution or wave function. It has been observed that the minima of the potential is more well-behaved than the maxima of the wave function. We illustrate clustering using QC method on the historic IRIS dataset. IRIS is a three class 4-D problem, with each class denoting a ower type – Virgnica, Versicolor and Setosa. While Setosa is linearly separable from the other two, the remaining two classes are mutually non-separable. In Figure 1 plot the data using the first two principal components of the dataset.

Figure 2 shows the plot of normalized potential function, V/E, for the IRIS Dataset, with the threshold on deciding the minima as a dotted line. The plot of the distribution function is similarly shown in Figure 3. It is evident that the maxima of the distribution function is difficult to find and does not occur in all the three regions. However, the minima of the potential function can be found in every cluster. Thus minima of potential is easily obtained for every cluster than the maxima of the wave function.

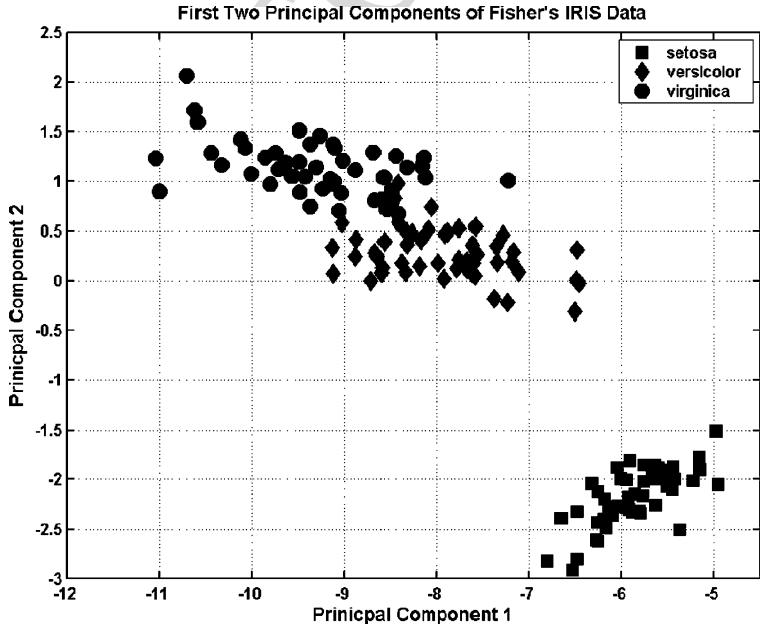

Figure 1. Plot of the first two principal components of the IRIS data.

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

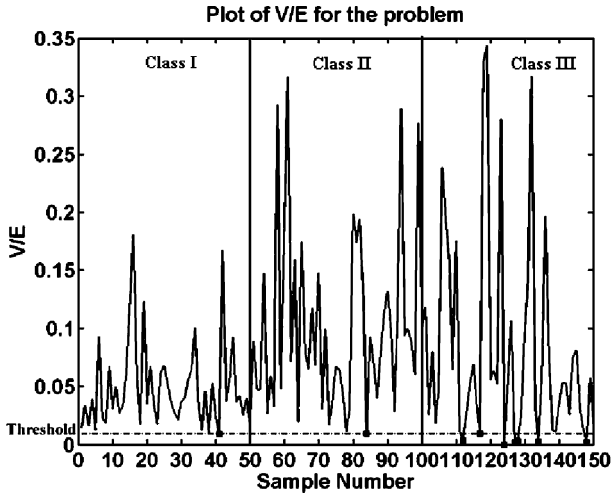

Figure 2. Plot of the potentials at the datapoints of the IRIS data show that all clusters have at least one point as the minima.

3. Visual-Motor Coordination Control

3.1. THE PROBLEM

A pair of 2-D 'retinal' coordinates u_1, u_2 are obtained from the two cameras, respectively, as shown in Figure 4. The objective is to identify the transformation $\Theta(\mathbf{u}_{\text{target}})$ from the retinal coordinates to the joint angles θ of the three-joint arm. The vector $\mathbf{u}_{\text{target}}$ is a 4-D vector obtained by grouping the retinal coordinates $\mathbf{u}_1$and $\mathbf{u}_2$. In

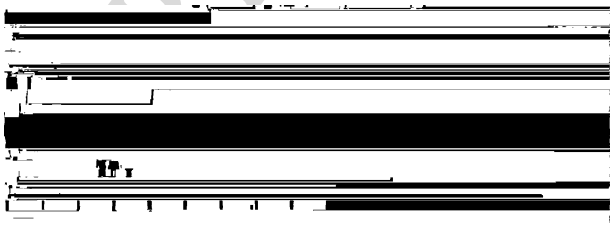

Figure 3. Plot of the wave functions at the datapoints of the IRIS data show that the maxima are not well defined for all the clusters.

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

this work, we consider $\mathbf{u}_{\text{target}}$ as a 3-D vector by taking into account the actual 3-D location of the end-effector in the workspace rather than through the transformed camera-plane. That is, we do not consider the camera transformation and work on the original end-effector position in the workspace. The first-order Taylor expansion of $\Theta\mathbf{u}_{\text{target}}$ is given by Equation (6).

$$\Theta(\mathbf{u}_{\text{target}}) = \theta_s + \mathbf{A}_s(\mathbf{u}_{\text{target}} - \mathbf{w}_s). \tag{6}$$

The idea is to discretize the workspace $\mathbf{U}$, into non-overlapping regions $\mathbf{F}_r$, such that $\mathbf{w}_r \in \mathbf{F}_r$ is the reference or weight vector, θ_r is the zero-order term and $\mathbf{A}_r$ is the Jacobian Matrix, which determines the first order term. Equation (6) gives the local approximation of each neuron for the joint angle values of the target point $\mathbf{u}_{\text{target}}$

3.2. A JOINT INPUT – OUTPUT SPACE PARTITIONING SCHEME

We simulate the workspace using the inverse kinematics relationship between the end-effector position in the 3-D world space and the joint angles. This inverse kinematics relationship is easily obtained using the geometry of the robot manipulator. A typical manipulator is shown in Figure 5. Given a set of end-effector positions $\mathbf{u}$ and their corresponding joint-angles θ from the workspace, we consider a 6-D vector $\mathbf{z}$ by grouping $\mathbf{u}$ and θ. The motivation for the proposed approach can be under-stood if we observe from Figure 6 that the output space formed by the joint angles is clustered into four well-defined regions. Thus, the vector $\mathbf{z}$ also forms a clustered space. This implicit relationship between the workspace and the joint-angle space when considered when initializing and training the network, greatly improves the capability of the network while decreasing the complexity of the network topology. It also follows from the kinematics of the manipulator that not all joint-angle values will be allowed when it is expected to work in a given workspace due to dimensional constraints.

Using QC, we obtain the cluster centers in the space represented by $\mathbf{z}$. The initial values for $\mathbf{w}_r$ and θ_r are then extracted back from the 6-D cluster centers. It should be noted that though a given joint angle determines a unique end-effector position, it is not true otherwise. An end-effector position maybe realized by more than one joint-angle values. This implicit information must be accounted for while preparing the

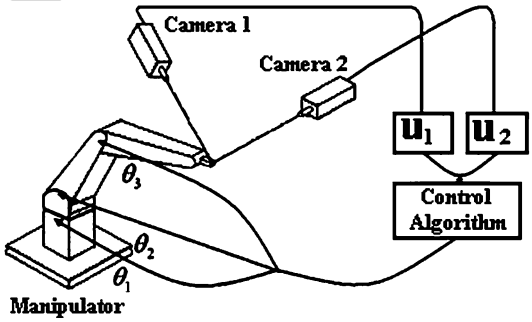

Figure 4. Schematic diagram of the visuo-motor setup.

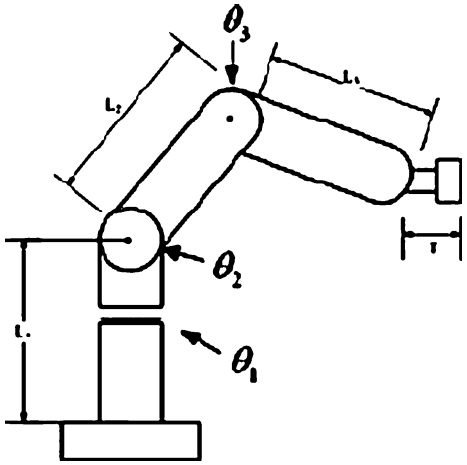

Figure 5. Schematic diagram of the manipulator which can be used to generate the inverse kinematics relationship.

Figure 6. Plot of the reference centers of different neurons and corresponding joint-angles obtained using QC-based method.

network. This forms one of the motivation for resorting to the proposed clustering method. It has been observed that such a selection needs a small fine-tuning of the cluster centers and hence has a very fast learning time. In addition, by an adaptive selection of clusters from the workspace, we have reduced the number of neurons significantly.

3.3. QUANTUM CLUSTERING VS KOHONEN'S SELF-ORGANIZING MAPS FOR VISUAL MOTOR COORDINATION

In the Kohonen's Self-organizing Feature Map based method, a fixed topology for the space is used by creating a 3-D lattice of neurons. These neurons then organize themselves homogeneously so that the workspace is well spanned and Equation (6) can be used to identify the joint-angles for a given end-effector position. This has two major disadvantages:

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

– the number of neurons have to be pre-decided and fixed, thus allowing little flexibility;
– the topology of the workspace is fixed once the lattice parameters are specified.

A fixed topology of the workspace is not necessary because that constraints the neurons to occur uniformly. The receptive field dimensions of all neurons in the visual system is not same. The proposed method has the same efficiency using almost half the neurons used in the K-SOM based approach and a flexible topology. Figure 6 shows the distribution of points in a workspace and the corresponding points in the joint-angle space.

4. Visual-Motor Coordination using Quantum Clustering based Neural Control Scheme

4.1. THE NEURAL LEARNING ALGORITHM

The Learning Algorithm adopted in the present work is motivated by the extended Kohonen's Self-Organizing Feature Maps introduced by Walter and Schulten in [11]. Given an end-effector position target $\mathbf{u}_{\text{target}}$, a winner neuron μ is selected, based on the Euclidean distance metric in the workspace. The neuron whose reference vector is closest to the target is declared winner. The arm is given a coarse movement, θ_0^{out} by the initial output of the network determined by Equation (7) resulting in the end-effector to move to a position $\mathbf{v}_0$. This if followed by a fine movement determined by θ_1^{out}, using Equation (8), giving a correcting movement to the arm and the end-effector reaching the final position of $\mathbf{v}_1$. It has been shown that a series of similar corrective actions will converge to the correct end-effector position. However, we use only one corrective fine movement and achieve a considerable amount of accuracy.

The collective averaged output is evaluated using Equation (7).

$$\theta_0^{\text{out}} = s^{-1} \sum_k h_k^{\text{mix}} (\theta_k + \mathbf{A}_k(\mathbf{u}_{\text{target}} - \mathbf{w}_k)), \tag{7}$$

where $s = \sum_k h_k^{\text{mix}}$ and $h_k^{\text{mix}} = \exp((\|\mu - k\|)/2\sigma_{\text{mix}}^2)$, with μ representing the winner neuron such that $\mathbf{w}_\mu$ is closest to the vector $\mathbf{u}_{\text{target}}$. θ_0^{out} gives a coarse movement to the arm such that the end-effector reaches a position $\mathbf{v}_0$. The correcting fine movement is evaluated using Equation (8) resulting in a final movement of the end-effector to $\mathbf{v}_1$.

$$\theta_1^{\text{out}} = \theta_0^{\text{out}} + s^{-1} \sum_k h_k^{\text{mix}} \mathbf{A}_k(\mathbf{u}_{\text{target}} - \mathbf{v}_0). \tag{8}$$

The learning scheme used in the present work can be grouped as under:

$$\Delta \mathbf{v} = \mathbf{v}_1 - \mathbf{v}_0, \tag{9}$$

$$\Delta \theta^{\text{out}} = \theta_1^{\text{out}} - \theta_0^{\text{out}}, \tag{10}$$

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

$$\Delta\theta_k = \theta_0^{\text{out}} - \theta_k^{\text{out}} - \mathbf{A}_k(\mathbf{v}_0 - \mathbf{w}_k), \tag{11}$$

$$\Delta\mathbf{A}_\mu = \|\Delta\mathbf{v}\|^{-2}(\Delta\theta^{\text{out}} - \mathbf{A}_\mu \cdot \Delta\mathbf{v})\Delta\mathbf{v}^{\text{T}}, \tag{12}$$

$$\mathbf{w}_k \leftarrow \mathbf{w}_k + \epsilon\, h'_{\mu k}(\mathbf{u}_{\text{target}} - \mathbf{w}_k), \tag{13}$$

$$\theta_k \leftarrow \theta_k + \epsilon'\, h'_{\mu k}\Delta\theta_k, \tag{14}$$

$$\mathbf{A}_k \leftarrow \mathbf{A}_k + \epsilon'\, h'_{\mu k}\,\Delta\mathbf{A}_k. \tag{15}$$

The functions $h_{\mu k}$ and $h'_{\mu k}$ are defined as

$$h_{\mu k} = \exp\left(-\frac{\|\mu - k\|}{2\sigma^2}\right), \tag{16}$$

$$h'_{\mu k} = \exp\left(-\frac{\|\mu - k\|}{2\sigma'^2}\right), \tag{17}$$

with μ and k representing the 3-D indices associated with the corresponding neuron. The parameters $\epsilon, \epsilon', \sigma, \sigma'$ and σ_{mix} vary during the training time depending on the current iteration and can be expressed using the general expression as below

$$\eta = \eta_{\text{initial}}\left(\frac{\eta\text{final}}{\eta\text{initial}}\right)^{(t/t_{\max})}. \tag{18}$$

In Equation (18), $\eta \in \{\epsilon, \epsilon', \sigma, \sigma', \sigma_{\text{mix}}\}$, t is the current iteration and $t_{\max}$ is the total number of iterations to be performed by the network.

4.2. INDEXING SCHEME FOR THE ADAPTIVE TOPOLOGY OF EXTENDED KSOM

In this work, we use an adaptive scheme for determining the topology of the workspace using QC of the joint space formed by grouping the end-effector positions and the corresponding joint-angles. After the clustering step, a set of points, $\mathbf{w}_1, \ldots, \mathbf{w}_N$, in the workspace are obtained, which are the potential cluster centers and hence denote the reference or weight vector for the neurons. The index of each neuron is then obtained using a normalized and scaled version of the corresponding weight vector. The normalization is carried out by dividing each dimension by the corresponding linear dimension of the workspace. It should be noted that in the proposed approach, the index values are real values and not necessarily integers, unlike the method in [6] and [11]. The motivation for the proposed indexing scheme is that it represents both the actual and the relative location of neurons in the workspace.

5. Implementation and Results

5.1. PARAMETERS AND INITIALIZATION

The workspace is defined by a region 200 mm × 300 mm × 200mm. The arm lengths are all equal to 254 mm and the manipulator has a rigid wrist portion having a

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

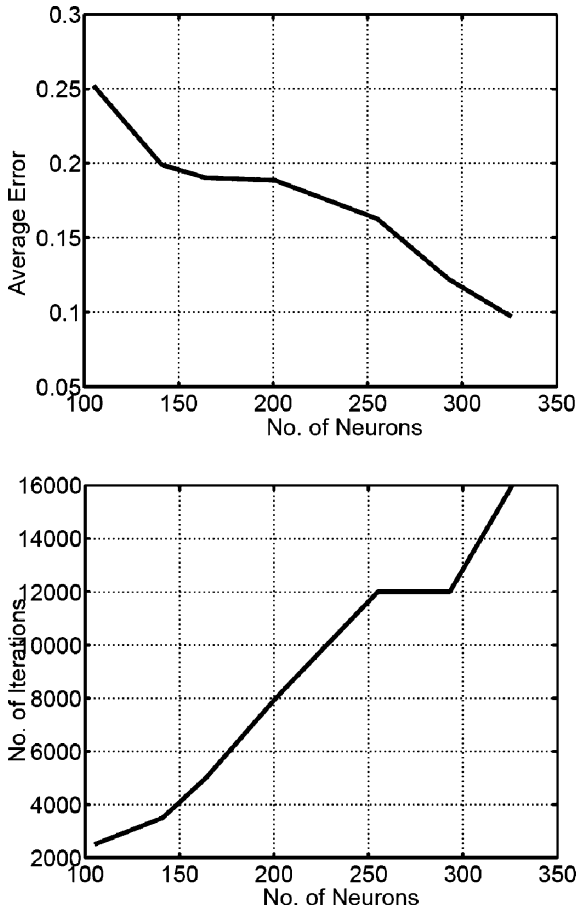

Figure 7. Plot of the average error and no. of iterations required as the number of neurons varies.

Table I. Performance (in terms of the mean normed error) of extended K-SOM based method when 5000 iterations were used to train the system.

SOM topology	No. of neurons	200 Random points	Circle	Sphere
12×7×4	336	0.1438	0.0722	0.0711
8×7×5	280	0.1551	0.0750	0.0715
6×6×5	180	0.1931	0.1162	0.1133
6×5×5	150	0.2122	0.1001	0.0924

length equal to 50 mm as used in [1]. The initialization of parameters was similar to that in [1]. The values were chosen as $\epsilon_i = 1.0$, $\epsilon_f = 0.05$, $\epsilon'_i = 0.9$, $\epsilon'_i = 0.9$, $\sigma_i = 2.5$, $\sigma_f = 0.01$, $\sigma'_i = \sigma_{mix_i} = 2.5$, $\sigma'_f = \sigma_{mix_f} = 0.01$. The subscript i and f denote the initial and final values of the corresponding parameters, respectively.

Journal : NEPL SPS Article No : 123R1 Dispatch : 9-6-2004 Pages : 12
PIPS No. : 5381183 ☐ LE ☑ TYPESET
MS Code : NEPL 123R1 ☑ CP ☐ DISK

Table II. Performance (in terms of the mean normed error) of QC based method when 5000 iterations are used.

$q = 1/2\sigma^2$	No. of neurons	200 Random points	Circle	Sphere
0.14	255	0.1481	0.0803	0.0968
0.125	181	0.1804	0.10	0.11
0.12	164	0.1814	0.09	0.13
0.11	141	0.2291	0.1088	0.1334

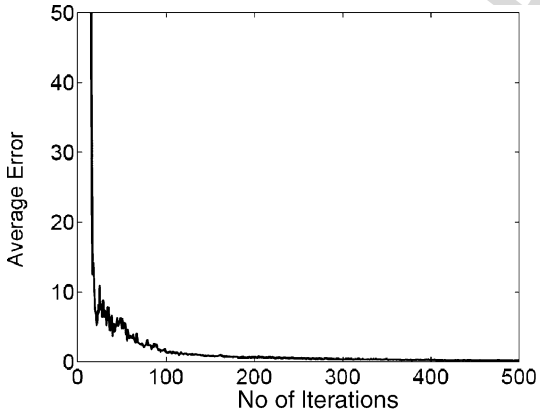

Figure 8. Plot of the average error with the number of iterations for 164 neurons.

5.2. PERFORMANCE AND RESULTS

The topology of the workspace is determined by the number of clusters obtained. Unlike the Kohonen-SOM based approach where the workspace topology in the form of a 3-D map has to be pre-specified after a heuristic approach mostly based on hi-and-trial, the QC-based approach has only one free parameters in form of the cluster width parameter $q = 1/2\sigma^2$, which controls the number of clusters generated. Each cluster denotes the receptive field of a neuron.

The proposed algorithm is tested on 200 Random Points in the workspace, a circle in a 2-D plane and a Sphere. The error measured is reported using the Mean Normalized Error (MNE). MNE for a set of dataset with N points in the workspace is defined by Equation (19).

$$\text{MNE} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{u}_{\text{target}} - \mathbf{u}_{\text{predicted}}\|. \tag{19}$$

We begin by evaluating the number of neurons which are necessary for representing the workspace appropriately. We observe that larger number of neurons give

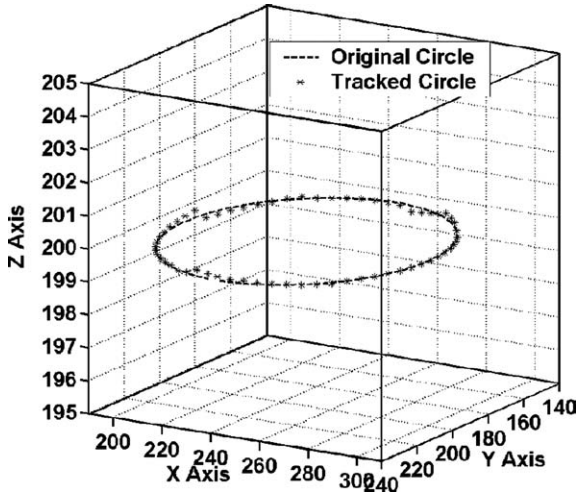

Figure 9. Plot of the tracked circle using 164 neurons.

better results, but also require more training. This is evident from Figure 7. Figure 7 shows that increasing the number of neurons shows a decline in the MNE. However, for faster training, it is important to have only a sufficient number of neurons.

Our simulations lead us to the conclusion that the adaptive toplogy based scheme proposed in this work needs only 164 neurons for the workspace in use. This number can be compared with the work in [1], where $12\times7\times4$ ($= 336$) neurons are used. The result for three situations using different SOM topology and also with different cluster widths is demonstrated in Tables I and II. They show that the QC based method achieves better results with fewer neuron units, mainly because of the fact that, the topology of the workspace is not fixed but is determined through a joint clustering process.

The error vs. iterations graph is shown in Figure 8 for an adaptive scheme of 164 neurons. Figure 9 shows a circle being tracked in the workspace with an average error of 0.09 mm. These results are comparable to that obtained using KSOM-based method where a much higher number of neurons were used.

6. Conclusion

In this work, a new method for the visual-motor coordination Control using an adaptive scheme and a flexible topology for the workspace is proposed. The novelty of the method lies in a joint input–output partitioning scheme. This leads to fewer neurons being used with the receptive field of every neuron having a non-uniform distribution in space. The motor control is carried out through a collection of neurons, all in a neighborhood, firing and determining the precise movement, instead of one neuron acting independently. QC used in this work, determines the local

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK

maxima of the potential and identifies them as the cluster centers. It should be noted that the method still adopts a classical computation methodology.

References

1. Behera, L. and Kirubanandan, N.: A hybrid neural control scheme for visual-motor coordination. *IEEE Control Systems Magazine*, **19**(1999), 34–41.
2. Duda, R. Hart, P. and Stork, D.: *Pattern Classification*. John Wiley & Sons, 2001.
3. Horn, D.: Clustering via hilbert space. *Physica A*, **302**(2001), 70–79.
4. Kuperstein, M.: Adaptive visual – motor coordination in multijoint robots using parallel architectures. In *Proceedings of the IEEE International Automation and Robotics*, pp. 1595–1602, 1990.
5. Kuperstein, M.: Neural model of adaptive hand-eye coordination for single postures. *Science*, **239**(1998), 1308–1311.
6. Martinetz, T. Ritter, H. and Schulten. K.: Three dimensional neural network for learning visuomotor coordination of a robot arm. *IEEE Transactions on Neural Networks*, **1**(1990), 131–136.
7. Ripley, B.: *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996.
8. Ritter, H. Martinetz, T. and Schulten, K.: Topology conserving maps for learning visuomotor coordination. *Neural Networks*, **2**(1998), 159–168.
9. Roberts. S.: Non-parametric unsupervised cluster analysis. *Pattern Recognition*, **30**(1997), 261–272.
10. Vapnik, V.: *Statistical Learning Theory*. John Wiley & Sons, 1998.
11. Walter, J. and Schulten, K.: Implementation of self-organizing neural networks for visuo-motor control of an industrial robot. *IEEE Transactions on Neural Networks*, **4**(1993), 86–95.

Journal : NEPL SPS Article No : 123R1 | Dispatch : 9-6-2004 | Pages : 12
PIPS No. : 5381183 | ☐ LE | ☑ TYPESET
MS Code : NEPL 123R1 | ☑ CP | ☐ DISK