GENERATION OF LIMB TRAJECTORIES WITH A SEQUENTIAL NETWORK

Lina Massone and Emilio Bizzi

Dept. of Brain and Cognitive Sciences
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

This paper focuses on the representation and generation of unconstrained aiming movements of a limb by means of a neural network architecture. The goal is that of producing a time trajectory of a limb from a starting posture toward a target specified as a sensory stimulus. The velocity profile along the trajectory is imposed to be bell-shaped as in most movements performed by biological systems. The generalization capabilities of the network are investigated as well as its internal organization. Two experiments are performed on the trained network to test its robustness to noise and its dynamical properties.

Introduction

This paper focuses on the representation and generation of ***unconstrained aiming movements*** of a limb by means of a neural network architecture. Aiming movements are present, in biological systems, at different levels of complexity, from accurately planned movements to reflexes [1]. The class of aiming movements addressed in this work is that of unconstrained limb movements elicited by sensory stimulation. They are meant to mimic, for example, the wiping movements of spinal frogs (i.e. leg movements which occur when the frog skin is stimulated by an irritant) or the scratch reflexes of spinal cats [2,3,4]. Recording of frogs wiping movements [2] shows that the motor strategy remains basically the same - with minor variations - in both intact and spinal animals, suggesting that the basic motor programs for this particular task are generated at the spinal cord level and not explicitly planned by higher brain structures. On the basis of this observation, we adopted a non-hierarchical, purely executional neural network to represent such movements.

The motor task described in this paper is that of generating a trajectory of a limb from a fixed starting posture toward a target specified in terms of sensory stimulus. Hence the network performs a *sensory-motor transformation*. Movements are assumed to be planar, but there is no theoretical limitation to the dimensionality the network could deal with. The velocity profile along the trajectories is imposed to be *bell-shaped* - as in most movements performed by biological systems; in fact, it has been shown [5,6,7,8] that the limb's end-point motion is smooth when the trajectory's velocity is bell-shaped. The duration of movements under consideration is assumed to be constant. As a consequence, when the neural network is asked to generalize the task, it is required not only to generate the correct trajectory of the limb, but also to adjust the velocity profiles accordingly.

A major goal of this research is to investigate the generalization capabilities of the network as well as its internal organization. In addition, two experiments are performed on the trained network to test its robustness to noise and its dynamical properties.

Architecture

Neural networks have been frequently applied to the robotic and motor control fields [9,10,11,12,13,14,15,16]. The architecture which we chose to produce time sequences is called a ***sequential network*** and it was proposed by Jordan [17]. Figure 1 shows the basic structure of such network. This type of network is able to produce sequences of output signals, due to the recurrent connections from *output units* to *state units*. These connections cause the activation of the state units to change, thereby providing a time-varying input to the layered network which learns the sequences. The self-connections to the state units make the next state of the network a function of the whole past history. The other input to the layered network derives from the *plan units*, which are activated by the external stimuli. The activation of the plan units remains constant within a given sequence, but varies between sequences to allow different sequences to be learned by the same network. The network is a dynamical system in which both output functions and state functions change as the weights in the forward path of the network are modified through learning. The reminder of this section shows how the aiming task can be formulated in terms of sequential networks.

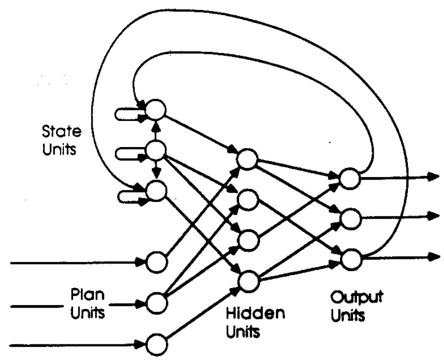

Figure 1. The basic architecture of Jordan's sequential networks.

Output units drive a redundant 3-joint limb which moves from the initial posture to the target. The limb is modeled in terms of 4 pairs of antagonist muscles: shoulder flexor and extensor, double joint flexor and extensor, elbow flexor and extensor, wrist flexor and extensor. Muscles are represented as springs according to a model that is described in next section. Each output unit activates a muscle; output units can then be considered as motorneurons. The time sequence generated by the network is hence encoded in muscle space (not in joint or end-point space).

Plan units contain a representation of the sensory stimulus. It is assumed that the limb workspace and the limb itself can be measured in the same body-centered reference frame (as in the aiming phase of the wiping to the back); consequently, the coordinate transformation problem (from world-centered to body-centered coordinates) is not addressed in this work [1]. Part of the limb workspace (see figure 2a) was discretized, as shown in figure 2b, with a 15x15 pixel grid; also figure 2b shows the initial posture of the limb for all aiming movements. The choice of the grid step, i.e. the resolution over the input space, is completely arbitrary, but the same learning and generalization procedures were repeated for a different grid step (see next section) with the purpose of understanding how a change in resolution might affect the performance of the network. The stimulus is encoded as a narrow gaussian distribution centered on one of the 225 pixels; any pixel

[1] It was addressed elsewhere, see for example [10,11,16].

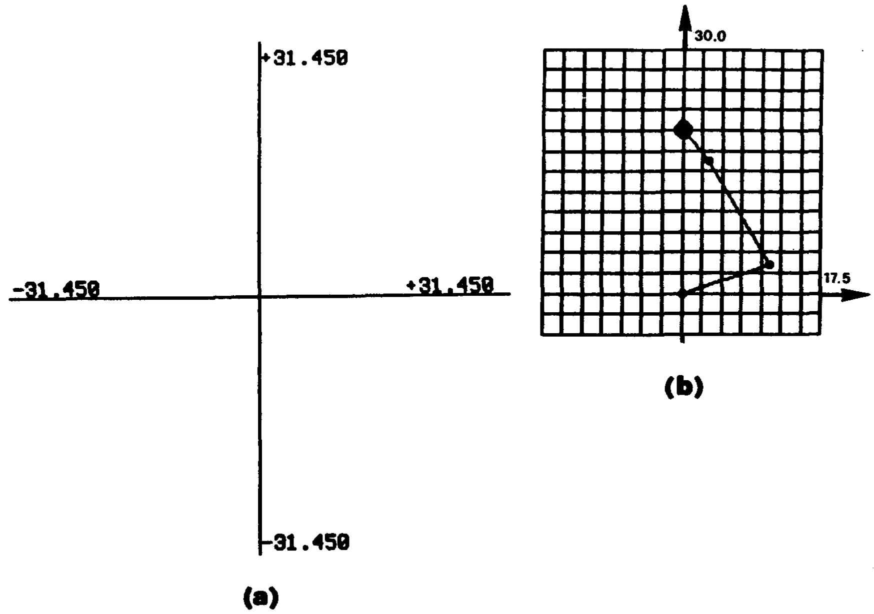

Figure 2. (a) The limb workspace. The origin of the coordinates is in the shoulder; 31.45 is the sum of the joint lengths. (b) A 15x15 pixels grid on a part of the workspace shown in figure 2a; the step is 2.5. The limb is in the starting posture. The lenghts and proportions of the three joints are those of a monkey arm.

may become the target of the aiming movement. The values of pixels were translated into activation of plan units by means of *coarse coding* [18]. Each plan unit has a *receptive field* over the limb workspace which contains 9 pixels; receptive fields are overlapped. In this way the 15x15 array of pixels is represented by means of a 7x7 array of plan units; the activation of each plan unit is the sum of the values of the pixels which belong to its receptive field. The resulting value is then properly normalized.

The number of *hidden units* was empirically fixed to 10. This number resulted from a trade-off between two conflicting needs: i) to provide enough units so that the network can be able to compute the input/output transformation, ii) to keep the number of hidden units as low as possible to achieve good generalization (an excess number of hidden units would make the network act as a look-up table). A graphic interface built on top of the simulation software made it relatively easy to investigate the behavior of the hidden layer during the learning phase; we have found experimentally that when the number of hidden units is greater than 10 learning is not improved and generalization properties are negatively affected.

All units in the network have continuous activation functions. Output functions of plan, hidden and output units are of logistic type (sigmoids) between a minimum and a maximum value, namely: between -1 and 1 for output units and between 0 and 1 for hidden and plan units. This choice was dictated by the nature of the problem: in this particular implementation (see next section) muscle activations range between negative and positive values which were normalized to the interval [-1,1]; the stimulus representation ranges between 0 (no stimulus) and a maximum positive value (activation of the unit which contains the gaussian peak), normalized to the interval [0,1]. The activation function for state units is linear.

Training and Generalization

The network was trained by means of supervised learning (see [19] for a review on supervised learning). This class of learning algorithms requires providing a set of input - output pattern pairs, in our case {sensory stimulus - time trajectory in muscle space}. Computing the sequence of muscle activations which corresponds to a trajectory of the limb from the initial to the final posture is not straightforward. For this purpose, we used a model which represents a redundant motor system in the form of a network of constraints expressing the geometrical relations among component elements and their steady-state mechanical behavior [20,21]. This model is based on experimental investigations that stressed the role of muscle mechanical properties in motor control, suggesting that a muscle is mechanically analogous to a *tunable* spring [22,23,24,25]. The model makes it possible to compute i) the xy coordinates of the limb tip given the muscle activations, which is a well-posed problem and ii) the muscle activations given the xy coordinates of the tip, which is a ill-posed problem. Muscle elastic properties are exploited as a natural representation for motor redundancy; the redundancy present in ii) is solved by representing the set of muscles as a chain of spring-like elements and by observing that the chain would naturally settle into a configuration of *minimum potential energy* when perturbed by an external force. The minumum potential energy criterion makes the solution to the inverse problem unique. The model can then compute a trajectory in muscle space given the initial and final tip positions; also, it makes it possible to specify the velocity profile along the trajectory (bell-shaped in our case). The result of the computation is a set of 8-dimensional vectors of muscle activations, each corresponding to an equilibrium configuration [2]; such positions are equispaced in time but not in space because of the bell-shaped velocity profile. The inverse transformation was used to compute the output patterns to train the network, while the direct transformation (from muscle activation to tip position) was used during the testing phase.

The particular algorithm used to train the network is a standard back-propagation algorithm which makes use of a momentum term; the learning rate was interactively lowered during the training sessions to allow learning of coarser and finer movements. All trajectories used during the training phase have a duration of six time steps: initial posture, target posture and four intermediate postures.

A major concern of the training phase was how many and which sequences the network must learn to generalize correctly the task. The goal was that of achieving a generalization capability such that the error on the tip position [3] for each point of the trajectory did not exceed the grid step.

[2] It has been proposed [26,27] that arm movements are represented and generated by the central nervous system as smooth transitions in posture along *virtual trajectories* given as time sequences of *equilibrium configurations*. An equilibrium configuration is defined for a given value of muscle activation as that position at which the forces of opposing muscles generate equal and opposite torques about the joints.

[3] Errors were measured, for each tip position, as the euclidean distance between the tip position produced by the network and the expected tip position produced by Mussa Ivaldi's model.

This is equivalent to requiring that the network must behave well at the resolution imposed by the discretization of the limb workspace. This result was achieved after teaching the network 14 sequences; the corresponding stimulus positions are shown in figure 3. Figure 4 shows some generalized sequences. The tip position is correct along the whole trajectory and the velocity profile is properly adjusted. In addition, it is worth pointing out that the network could produce joint reversals when necessary; moreover, the net generated patterns of muscular activation which correspond to equilibrium positions of the limb. Two further learning experiments were performed. First, the learning procedure was repeated by making use of local coding instead of coarse coding (1 plan unit for each pixel, 225 plan units). After learning the same 14 sequences the network was not able to generalize and behaved as a look-up table. Second, the learning procedure was repeated for a lower resolution on the workspace, obtained by doubling the grid step. This led to a 7x7 array of pixels coarse coded by a 3x3 array of plan units. In this case the number of sequences to be taught to the network to produce errors lower than the grid step decreased from 14 to 8.

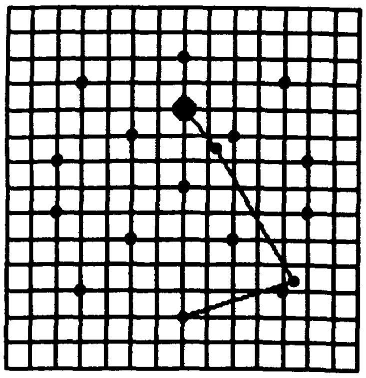

Figure 3. Position of the stimuli for the 14 sequences which were taught to the network.

Connections

The connection matrix was randomly initialized in the range {0.0 - 0.5}. We could observe that, after learning, the connections were organized into inhibitory and excitatory zones. Interesting patterns were found in the connections from hidden units to output units; Table 1 shows the values of such connections. By grouping muscles into flexor-extensor pairs, it can be observed that, for every hidden unit, whenever one hidden unit sends an excitatory connection to a flexor, the same unit sends an inhibitory connection to the corresponding extensor and vice-versa (***negative correlation***). The network has represented in the connectivity pattern the rule of reciprocal inhibition of agonist-antagonist pairs. Inhibition and excitation are more marked for shoulder, elbow and double joint muscles than for wrist muscles. This result agrees with the experimental data of Georgopoulos [1] which show that aiming movements involve wrist joint only in a very marginal way. Moreover (see again Table 1), it can be observed that

- units #3 and #10 exhibit a total ***positive correlation*** between all flexors and between all extensors;
- all other units exhibit a total ***positive correlation*** between
  - shoulder flexor - double joint flexor;
  - shoulder extensor - double joint extensor;
  - elbow flexor - wrist flexor;
  - elbow extensor - wrist extensor;

  except for hidden unit #2 for which shoulder and double joint exhibit a negative correlation.

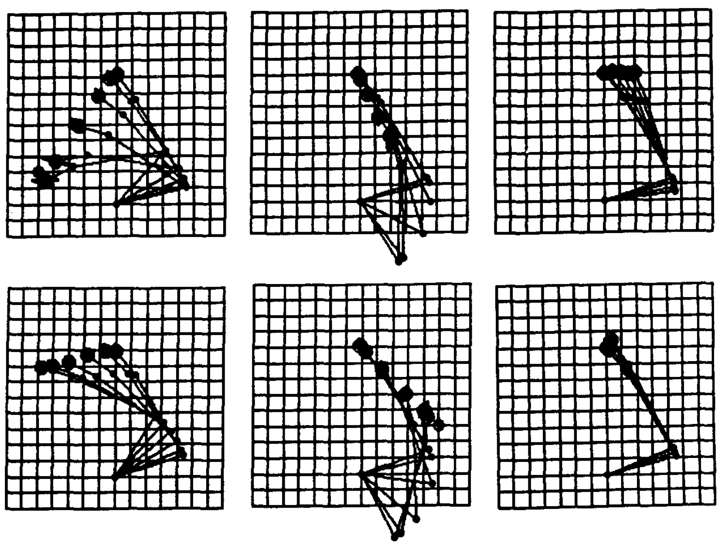

Figure 4. Generalization capability after learning 14 trajectories. The top-left trajectory contains a generalization of the joint reversal on the shoulder; the bottom right trajectory is a particular case of generalization in which the stimulus was positioned right on the limb tip; although the network was not explicitly taught about the initial posture, it has "understood" how the limb is positioned at the beginning of each trajectory.

Shoulder Fl.	−1.596938	−0.459756	0.238360	0.464764	−1.130529	1.225434	1.249900	−0.822495	0.946107	−1.160787
Shoulder Ex.	1.597220	0.458936	−0.238326	−0.464758	1.130295	−1.225155	−1.249560	0.822554	−0.946128	1.161238
Elbow Fl.	2.205403	1.201014	0.990554	−0.725229	1.035587	−0.788169	−1.198095	0.046117	−1.464604	−0.762154
Elbow Ex.	−4.299550	−0.824471	−2.072892	1.214777	−1.433692	0.294094	1.581221	0.024668	2.398601	0.365059
Double J. Fl.	−0.822154	1.664527	0.419754	0.069341	−0.511715	0.004101	0.556992	−0.617517	0.068661	−1.571512
Double J. Ex.	0.580872	−1.538881	−0.525151	−0.019188	0.490678	−0.050527	−0.561579	0.623930	0.010338	1.459015
Wrist Fl.	0.260377	0.220753	0.122209	−0.092430	0.129453	−0.155435	−0.139240	−0.012050	−0.188938	−0.090522
Wrist Ex.	−0.260409	−0.220748	−0.122226	0.092426	−0.129474	0.155452	0.139244	0.012054	0.188934	0.090537

Table 1. Hidden to output connections. Each row contains the connections from all hidden units to one particular hidden unit.

These observations indicate the presence of a number of synergies between all hidden units, which is the necessary condition for the network to exhibit good generalization properties. Furthermore, the network seems to have represented in the connectivity pattern the main features of the set of patterns which was used as training set. In fact, the sign of muscle activations in the training sequences was ***always the same*** for elbow-wrist flexors and elbow-wrist extensors and ***almost always the same*** for shoulder-double joint flexors and shoulder-double joint extensors. That "almost" has been encoded by means of a negative correlation at unit #2. Finally, the network has devoted two hidden units, #3 and #10, to encode the synergies between all flexors and between all extensors.

Experiments

Two experiments were performed on the trained network.

The first experiment aimed at testing the robustness of the system with respect to the sensory stimulus. The network was trained with stimuli coded as gaussian distributions centered on the target with a certain standard deviation d_0; the value of the standard deviation was modified during testing as follows:

$$d_1 = d_0 + 0.1 * d_0$$

$$d_2 = d_0 + 0.2 * d_0$$

Both cases correspond to a stimulus which is flatter and more spread over the workspace. In the first case the average distance in end-point space of the trajectories from the corresponding trajectories generated by a gaussian with standard deviation d_0 is lower than 0.4; in the second case the average distance is higher (around 0.7), which results in trajectories somewhat "noisy", but still acceptable. This experiment showed that the architecture is reasonably robust to slight changes in the stimulus representation.

The second experiment was concerned with the duration of the trajectories. Pineda [28] showed that arbitrary networks of logistic units typically have many point attractors, i.e. these networks naturally exhibit certain dynamic properties. In our case, the network was instructed, during training, to produce certain output patterns for six time steps; no instructions were given on what to do after the 6th time step. We tested the network for 15 time steps and we observed that in about 80 percent of the cases (i.e. in about 80 percent of the limb workspace) the limb remains steady at the final posture corresponding to the location of the sensory stimulus; in other words, in 80 percent of the cases the final posture of the limb acts as a point attractor. The portion of workspace in which the limb is unstable after the 6th time step changes with different learning sessions, i.e. it depends on which solution the network settles into; there were also cases in which the entire workspace was steady. Interestingly, adding a 7th time step to the training sequences which repeats the final position made the whole workspace steady independently of the particular training session.

Conclusions

In this paper a model for limb trajectory formation was presented, based on a non hierarchical neural network architecture. The task under consideration is that of reaching a target defined in terms of a sensory stimulus, with a bell-shaped velocity profile. The task is performed at the reflex level; no planning activity occurs. The network produces trajectories in muscle space, which are translated into end-point space by means of a model which takes into account the elastic properties of muscles [21]. The same model was used to generate the training sequences as described in section 3. The particular architecture used for producing time trajectories is that proposed by Jordan [17]. We have shown that the task can be learned and generalized (in terms of both trajectory and velocity profile) by a three layer sequential network trained by a standard back-propagation procedure. Moreover, we found that connections from hidden to output units exhibit a number of positive and negative correlations which encode the main features of the training set. The robustness of the model to noise on the input signals was successfully tested and some attractor dynamics properties were found.

Our model is different from that proposed by Kawato; he has studied voluntary movements and proposed a hierarchical, structured model for generating motor commands (torques) from a desired trajectory expressed in body centered coordinates [9]. Moreover, he has studied the coordinate transformation problem and proposed an iterative control learning algorithm [10]. Our research deals with a sensory-motor transformation based on a non-hierarchical layered architecture which translates a sensory stimulus directly into time-varying patterns of muscular activation which correspond to minimum jerk trajectories. We did not face the coordinate transformation problem as we made the hypothesis that both target and movement are already expressed in the same body-centered reference frame. We did address the problem of trajectory formation based on a constant sensory stimulus, rather than a reference trajectory. Issues related to trajectory formation were also investigated by Bullock and Grossberg [14] who have presented a model called VITE which produces arm trajectories from a target position command (TPC) and a GO command which defines the movement's speed. Although VITE has nice generalization properties, it is worth pointing out that trajectories are generalized in joint space, while our model can generalize trajectories in muscle space and then in end-point space through Mussa Ivaldi's model [21]. Moreover, VITE cannot be easily applied to multi-joint movements and does not address learning.

The work described in this paper is relevant to the robotics research as it could suggest some basic principles for designing artificial limbs whose structure is inspired by natural systems [29]. However, the relevance of our research to the understanding of the organization of biological motor systems is an open problem and will be the object of further investigations.

References

[1]. A. Georgopoulos, "On Reaching", Annual Review of Neuroscience, 9, pp. 147-170, 1986

[2]. M.B. Berkinblitt, A.G. Feldman, O.I. Fulkson, "Adaptability of Innate Motor Patterns and Motor Control Mechanisms", Behavioral and Brain Sci., 9, pp. 585-638, 1986

[3]. S. Giszter, J. McIntyre, E. Bizzi, "Kinematic Strategies and Sensorimotor Transformations in the Wiping Movements of Frogs", J. of Neurophysiol., in press.

[4]. R. Shadmer, G.D. Lindquist, "A Neural Model for Modifiable Pattern Generation in the Scratch Reflex", in Proc. 1st INNS Meeting, Boston (MA), pp. 359, 1988

[5]. P. Morasso, "Spatial Control of Arm Movements", Experimental Brain Research, 42, pp. 223-227, 1981

[6]. W. Abend, E. Bizzi, P. Morasso, "Human Arm Trajectory Formation", Brain, 105, pp. 331-348, 1982

[7]. C.G. Atkeson, J.M. Hollerbach, "Kinematic Features of Unrestrained Arm Movements", J. of Neurosci., 5(9), pp. 2318-2330, 1985

[8]. T. Flash, N. Hogan, "The Coordination of Arm Movements: an Experimentally Confirmed Mathematical Model", J. of Neurosci., 5(7), pp. 1688-1703, 1985

[9]. M. Kawato, M. Furukawa, R. Suzuki, "A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movements", Biol. Cybern. 57, pp. 169-185, 1987

[10]. M. Kawato, M. Isobe, Y. Maeda, R. Suzuki, "Coordinates Transformation and Learning Control for Visually Guided Voluntary Movements with Iteration: a Newton-like Method in Function Space", Biol. Cybern., 59, pp. 161-177, 1988

[11]. M.I. Jordan, "Supervised Learning and Systems with Excess Degrees of Freedom", COINS Tech. Rep., MIT. Cambridge (Ma), 1988

[12]. H. Miyamoto, M. Kawato, T. Setoyama, R. Suzuki, "Feedback Error Learning Neural Network for Trajectory Control of Robotic Manipulator", Neural Networks, vol. 1, pp. 251-265, 1988

[13]. R. Eckmiller, "Concept of a 4-Joint Machine with Neural Network Control for the Generation cf Two-Dimensional Trajectories", Proc. 1st Annual INNS Meeting, Boston (MA), pp 334, 1988

[14]. G. Schoner, J.A.S. Kelso, "Dynamic Pattern Generation in Behavioral and Neural Systems", Science, 239, pp. 1513-1520, 1988

[15]. D. Bullock, S. Grossberg, "Neural Dynamics of Planned Arm Movements: Emergent Invariants and Speed Accuracy Properties During Trajectory Formation, in Neural Networks and Natural Intelligence, S. Grossberg Ed., pp. 553-622, MIT Press, 1988

[16]. M. Kuperstein, "Neural Model of Adaptive Hand-Eye Coordination for Single Postures", Science, Vol. 239, pp. 1308-1311, 1988

[17]. M.I. Jordan, "Attractor Dynamics and Parallelism in a Connectionist Sequential Machine", in Proc. 8th Annual Conf. of the Cognitive Science Society, Hillsdale, N.J.: Erlbaum, 1986

[18]. G.E. Hinton, J.L. McClelland, D.E. Rumelhart, "Distributed Representations", in Parallel Distributed Processing, McClelland JL, Rumelhart DE Eds., MIT Press, Cambridge (MA), 1986

[19]. G.E. Hinton, "Connectionist Learning Procedures", Tech. Rep. CMU-CS-87-115, 1987

[20]. F.A. Mussa Ivaldi, J. McIntyre, E. Bizzi, "Theoretical and Experimental Perspectives on Arm Trajectory Formation: A Distributed Model for Motor Redundancy", in Biological and Artificial Intelligence Systems, E. Clementi and S. Chin Eds.,pp. 563-577, Escom, 1988

[21]. F. A. Mussa Ivaldi, P. Morasso, R. Zaccaria, "Kinematic Networks - A Distributed Model for Representing and Regularizing Motor Redundancy", Biol. Cybern., 60, pp. 1-16, 1988

[22]. A.G. Feldman, "Once More on the Equilibrium Point Hypothesis (λ model) for Motor Control", Journal of Motor Behavior, 18, pp. 17-54, 1986

[23]. P.M.H. Rack, D.R. Westbury, "The Effects of Length and Stimulus Rate on Tension in Isometric Cat Soleus Muscle", J. Physiol. , 204(443), 1969

[24]. E. Bizzi, A. Polit, P. Morasso, "Mechanisms Underlying Achievement of Final Head Position", J. of Neurophysiology, 39, 435, 1976

[25]. T.R. Nichols, J.C. Houk, "Improvement in Linearity and Regulation of Stiffness that Results from Action of Stretch Reflex", J. Neurophysiol., 29 (119), 1976

[26]. E. Bizzi, N. Accornero, W. Chaple, N. Hogan, "Posture Control and Trajectory Formation during Arm Movement", Journal of Neuroscience, 4(1: 2738-2744, 1984

[27]. N. Hogan, "The Mechanic of Multi-Joint Posture and Movement Control", Biol. Cybern., 53 (1), 1985

[28]. F.J. Pineda, "Generalization of Backpropagation to Recurrent and Higher Order Neural Networks", in Proc. IEEE Conf. on Neural Networks and Information Processing Systems, Denver (CO), 1987

[29]. D. De Rossi, G. Casalino, P. Morasso, "Skin and Muscles: the Basic Wetware for Sensory-Motor Control, in Proc. 1st Annual INNS Meeting, Boston (MA), pp.332, 1988

ACKNOWLEDGEMENTS

The authors wish to thank Michael Jordan for his constant support and valuable suggestions, and for making available the basic software which implements sequential networks; and also F.A. Mussa Ivaldi, J. McIntyre and S. Giszter. This work has been supported by the Office of Naval Research Grant N00014/88/k/0372 and the Sloan Foundation.