

DOTTORATO DI RICERCA IN BIOINGEGNERIA  
UNIVERSITÀ DEGLI STUDI DI BOLOGNA  
XIX CICLO



***PHD THESIS:***  
**“MODELS FOR THE MOTOR CONTROL OF THE UPPER LIMB”**

**IVAN BERNABUCCI**

*Supervisore:* Prof. Tommaso D’Alessio  
*Università degli Studi di Roma TRE*

*Correlatore:* Prof. Mauro Ursino  
*Università degli Studi di Bologna*

*Controrelatore:* Prof. Angelo Cappello  
*Università degli Studi di Bologna*

*Coordinatore:* Prof. Angelo Cappello  
*Università degli Studi di Bologna*

*Host institution:* *Università degli Studi Roma TRE*



*To my lab mates ...thanks to you this long journey has been a fun one.*

*To all my real friends who always supported and helped me.*

*To my family...*

<b>CHAPTER 1</b>	<b>INTRODUCTION</b> .....	<b>5</b>
	THE MOTIVATION .....	11
	STATE OF THE ART.....	12
	ORGANIZATION OF THE WORK .....	15
<b>CHAPTER 2</b>	<b>NEURAL CONTROLLER OF A BIOMECHANICAL ARM MODEL: MATERIALS AND METHODS</b> .....	<b>16</b>
	SUMMARY .....	17
	THE PROPOSED MODEL.....	17
	NEURAL NETWORK CONTROLLER .....	19
	<i>Introduction</i> .....	19
	<i>Historical Background</i> .....	21
	STRUCTURE OF THE NEURAL CONTROLLER.....	24
	THE PULSE GENERATOR.....	28
	IMPLEMENTATION OF THE BIOMECHANICAL ARM MODEL .....	29
	<i>Skeletal Structure of the Model</i> .....	30
	<i>Muscular Structure of the Model</i> .....	34
	STUDY OF VARIATIONS OF THE HILL'S PARAMETERS .....	40
<b>CHAPTER 3</b>	<b>LEARNING PARADIGM: IMPLEMENTATION</b> .....	<b>43</b>
	<b>HIERARCHICAL NEURAL CONTROLLER</b> .....	<b>43</b>
	SUMMARY .....	44
	NEURAL NETWORK TRAINING MECHANISMS .....	44
	LEARNING PARADIGM: DYNAMICS OF THE REACHING TASKS .....	47
	CONSTRUCTION OF THE INTERNAL MODEL : BIOLOGICAL LEARNING PARADIGM .....	48
	SIMULATING THE INTERNAL MODEL: THE TRAINING PHASE .....	51
	HIERARCHICAL NEURAL CONTROLLER: A COARSE TO FINE APPROACH .....	54
	SIMULATING THE INTERNAL MODEL: TESTING THE PERFORMANCE OF THE MODEL .....	57
	SIMULATING THE INTERNAL MODEL: TESTING THE PERFORMANCE OF THE MODEL FACING EXTERNAL FORCES .....	60
<b>CHAPTER 4</b>	<b>NEURAL CONTROLLER IN NORMAL AND DISTORTED ENVIRONMENT: RESULTS AND OBSERVATIONS</b> .....	<b>62</b>
	NEURAL CONTROLLER IN A NORMAL ENVIRONMENT .....	63
	ADAPTATION OF THE NEURAL CONTROLLER TO FORCE FIELDS: RESULTS AND OBSERVATIONS .....	75
	HIERARCHICAL MODEL: RESULTS AND OBSERVATIONS.....	83
<b>CHAPTER 5</b>	<b>NEURAL CONTROLLER: APPLICATION</b> .....	<b>88</b>
	SUMMARY .....	89
	MATERIALS AND METHODS.....	92

<i>The markerless motion estimation method</i> .....	92
<i>The proposed neural controller of the upper limb model</i> .....	98
EXPERIMENTAL TRIALS .....	98
RESULTS.....	102
DISCUSSION.....	103
<b>CHAPTER 7      CONCLUSION</b> .....	<b>105</b>
<b>APPENDIX A</b> .....	<b>108</b>
<i>Back-Propagation</i> .....	108
<i>Self-Organising Networks</i> .....	112
<i>Winner Selection: Dot Product</i> .....	112
<b>REFERENCES</b> .....	<b>115</b>

# **Chapter 1**

## *Introduction*

Understanding the development and the functioning of the human motor control undoubtedly represents a great challenge among all the scientific studies. Its great importance is linked to the fact that motor control allows to completely interface ourselves with the environment, exploiting the ability of transforming thought into action. The evolution of the investigation on this problem has been made possible thanks to a deep integration among several disciplines ranging from cognitive psychology to theoretical physics, from neurophysiology to control systems theory. The interest grown over the years by these research fields allows to face the motor control theory from different approaches: a neurobiologist may be interested on the neuroanatomical pathways and on the segmentation of the brain thus focusing the analysis on the purpose of the different areas or on the biochemical mechanisms of neural firing, a biomechanist would rather turn the interest on the musculoskeletal system while a control systems researcher would highlight the computational principles of biological feedback control and learning. Nevertheless, the stimulus on this extraordinary fascinating researching field has to continually cope the intrinsic complexity related both on the not exhaustive knowledge of the functional structures of the central nervous system and on the biomechanical architecture of the human body: apparently simple gestures are, as a matter of fact, the culmination of highly organized processes which include perception schemes, anticipative planning, feedback corrections, muscular synergies and other internal elaboration systems. The big effort, aimed at finding out and comprehending the relations between the controller (the Central Nervous System) and the controlled object (the body), is a reflex of the important implications of this subject both from a physiological and from a clinical rehabilitation viewpoint.

The analysis of a biological motor control can be characterized as a problem of controlling nonlinear, unreliable systems within a dynamic environment and whose states are monitored with slow and sometimes low-quality sensors. It's extremely important to

emphasize the role of two main aspects. The first one is the environment, conceived not only as a merely reference system but, in a Gibsonian way, a “provider of affordances”, which are specific information accessible during the execution of a whichever action and that are exploited in order to optimize the perception-action cycle. The second aspect is the presence of the “sensors” which are appointed to gather all the affordances, both from the “inside system” and the “outside world” and to supply these signals to the Central Nervous System; in this way the sensors comprise the perception, that is a mechanism aimed to forecast the sensorial consequences of an action [1]. Thanks to the integration of all the information concerning the effector which is being used and the relations between the environment and the effector itself, the human motor controller is able to manage movements involving the coordination of a dozen or more degrees of mechanical freedom, furthermore adding the often stringent demand on the precision of the movements, in terms of position (handwriting, eating), timing (typing) and force (kicking a ball, playing an instrument) [2]. A general attempt to elaborate an architectural and functional mapping of the Central Nervous System in relation to all these various aspects of the motor control, outlines three main structures: the spinal cord, the brainstem and the cerebral cortex, which are able to interact in a hierarchical and parallel way in order to define and realize the whole set of movements repertoire [3], from the reflexes to highly accurate goal directed tasks.

What is more astonishing about the human skill to control all these kind of movements is the learning capacity related to an high neuronal plasticity, which reveal itself since the birth and that becomes refined during the life cycle; the acquisition of cognitive abilities is a fundamental point in the investigation about motor control. The mechanism underling the neurophysiological development of a defined motor control has been described by several studies [4] [5] [6] [7]: what is commonly highlighted is the separation of the process in different phases; from the exploration of the outer space in order to acquire sensory-motor patterns to a circular-reaction behaviour that is defined by an automatic association of an action generation to a proper stimulus related to the same action. From an early age infants are found to have rather powerful adaptability and learning ability [8] together with a limited knowledge about the sensory-motor mapping of their bodies (i.e. moments of inertia, viscosity, stiffness of their arm segments) thus

showing a movements repertoire based primarily on reflexes and basal synergies [9]. Moreover there is a lack of a fully efficient visual structure and this limits the ability to generate a movement in order to reach something. The absence of these two characteristics prevents the newborns to execute a proper reaching task: what is necessary for having centrally planned and complex actions is a complete interaction between the controller and its “plants”. On this basis the infants develop more complex behaviours and motor mechanisms [10]. Appropriate head and trunk righting reactions begin to emerge 2 -3 months after birth. Despite all these limitations, babies as early as one week of age will attempt small arm movements directed towards the target, and are capable of orienting towards and tracking a moving object by means of coordination of head and eyes. A few days after birth infants are also able to perform anticipatory arm movements when trying to intercept a moving target [11]. Two and three month old infants’ movements appear to be pre-programmed, in a way that emphasizes the initial learning of the limbs dynamics in relation with a finalized action. About 3 months after the onset of reaching, infants reach consistently for objects in their surround and rarely miss their target. By the same time infants reveal improvements in their manipulative skills (e.g., precision grip). Kinematically, their hand paths become straighter and they seem to exploit the presence of the external forces acting on their body; the gravity force is the first force they learn to face and to use, in the sense that they do not initiate elbow extension by means of muscular activation but let the gravity act on the arm [12]. As a consequence of this learning process, infant tends to activate muscles only when it’s needed, reaching, however, an adult-like skill economy not before 24-36 months of age. But the evolutionary process of the human motor control is not the only extraordinary feature. Another significant aspect is the adaptability. The human motor control is able to change its activation signals depending on the variations of the parameters that generate them. A little perturbation on the visual system or on the tactile system or the presence of external loads acting on the plants, drive the Central Nervous System to modify the neural connections in order to achieve a normal motor behaviour. The adaptability allows the system to intrinsically take into account the external perturbations. Sensorimotor adaptation has been studied by introducing visual [13] [14] or mechanical distortions [15]

[<sup>16</sup>] or examining stable [<sup>17</sup>] [<sup>18</sup>] and unstable [<sup>19</sup>] [<sup>20</sup>] interactions produced by haptic interfaces.

In the last years the interest has been directed towards the study of these features of the human motor control and this has led to the hypothesis of the existence of the so-called “internal models”: models which are supposed to involve or consist of neural mechanisms that are capable of establishing a relation between input and output signals of the sensorimotor system in a feedforward fashion [<sup>21</sup>]. These structures are considered the keystone upon which the motor control takes shape and dynamically evolves. Internal models intrinsically contain information about biomechanical properties of the human body in relation both to the environment and the subject’s experience. An example of the role of the internal models can be pointed out while observing the control of the forces and the torques applied to the upper limb: when it accelerates the movement of the single joint causes inertial coupling upon all the other joints. Recent studies [<sup>22</sup>][<sup>23</sup>] pointed out that the compensation role of these forces is carried out by the cerebellum, proposed as the hypothetical site of the internal models. These are gradually built through practice and experience [<sup>24</sup>] and the proof of this construction derives from psychophysics studies, i.e. changing in the Electromyography registrations during the learning of a specific task [<sup>25</sup>][<sup>26</sup>][<sup>27</sup>]; internal models depend on task and limb structure [<sup>28</sup>][<sup>29</sup>].

There are two kinds of Internal Models (IM) [<sup>30</sup>]: the direct internal models (DIM) and the inverse internal models (IIM). The former are able to forecast the sensorial consequences of an action, assessing the limb future state (i.e. position and velocity) from the knowledge of the actual state and the motor commands. On the contrary the IIM can produce the motor commands which bring to a desired modification of the state. This natural differentiation concerns the discussion about: i) the existence of an anticipative control of the movements, ii) the role of a central planning actuated before the trigger of the task connected to corrective processes based on the feedback system. The presence of feedforward mechanisms is fundamental since sensorimotor control needs a significant and highly variable amount of time (150–250ms) to elaborate a motor reaction to a simple sensory feedback stimulus [<sup>21</sup>]. Many of the traditional human motor control models include both the control structures: “open-loop” control and “closed-loop” control [<sup>31</sup>][<sup>32</sup>]; however some of the recent studies on this subject have minimized the

importance or entirely neglected every contribution by the close-loop circuit [33]. Indeed there are experimental proof which reveal how the visual information is effectively used in order to execute fast adjustments on the trajectory [34], but a feedback control alone is not able to explain how “de-afferented” subjects are able to move an upper limb towards a target without visual and somatosensorial information (Willingham 2004). Anyway it has been demonstrated that adaptation to a new environment proceeds through the construction of the ‘internal models’ of body and environment, which is specific to the motor task in question [35][29][36][30].

Together with the internal model hypothesis, the equilibrium control point hypothesis has been presented. Following this theory muscles and peripheral reflex loops have spring-like properties that pull joints back to their equilibrium positions by generating a restoring force against external perturbations. In this way the trajectory becomes a series of equilibrium points. Due to the fact that this viscoelasticity can be regarded as peripheral feedback control gain, adjustable by regulating the associated muscle co-contraction level and reflex gain, exploiting it, the brain can control the limbs simply by commanding a series of stable equilibrium positions aligned along the desired movement trajectory, without the necessity to pre-program the muscular activation for the fast movements in order to avoid high delay in somatosensorial signals [37][38]. The drawback of this theory is that “..viscoelastic forces increase as the movement speeds up because the dynamic forces acting on the multijoint links grow in rough proportion to the square of the velocity..” [21]. The controversy is thus related to two different hypothesis of motor control behaviour; one relies on the idea of a high value of the viscoelastic forces and the other one which is based on the internal model. Recent observations of low stiffness during well-trained movements have suggested the hypothesis of internal models as the plausible theory [39][40].

## The motivation

The mechanisms underlying the generation and organization of the internal models are still object of controversy [<sup>41</sup>]. However, since these structures are believed to have a distributed neural-like internal structure, modern studies try to describe them by means of the use of the Artificial Neural Network (ANN), that is through parallel elaboration systems inspired by the structure and the physiology of the brain. The interest in the use of ANN depends on their capabilities to adapt and to generalise to new situations.

Following this perspective, the aim of the present work is to implement a software model based on artificial neural network that can control a synthesised human arm in order to learn ballistic movements in a specified workspace; more specifically the neural network has to simulate the behaviour of a specific controller which through a developmental process has to be lead to the generation of the internal model of the biomechanical arm.

In order to link the neural learning/adaptation processes to their artificial replica, ANN have been used in some studies regarding neurophysiologic simulations.

In most of these studies a connectionist model is designed, the input and the output patterns needed for the learning phase are prepared and the network is trained: this methodology, commonly implemented on forward multilayer networks with retrospective learning (back propagation, see Appendix A), is efficient from an operative standpoint, but not completely plausible as a biologically inspired learning model of motor control, at least for two principal reasons:

- the presence of a teacher who is pre-existent to the organization of the system.
- the fact that it is not possible to hypothesize a single homogeneous net responsible for the complete motor control when it's well known that the Central Nervous System is a highly complicated system composed by different nervous cells which define subnet, maps and subsystems.

In order to overcome these drawbacks, both a system based on a novel learning paradigm which neglects the presence of an external teacher and an evolved system structured in modules with a hierarchical organization are presented in this work.

The learning algorithm mimics the scheme generally considered for the development of reaching movements for infants in the earlier months of life: that is, the exploratory behaviour is not dependent on the target, which is not directly used to “goal-correct” the movement [42]. The neural system, which will simulate the behaviour of the Central Nervous System, and therefore the internal model, will be modelled as a generator of modified motor patterns, that is an Artificial Neural Network that generates the control signals which have to be sent to the biomechanical arm model. The hierarchical structure is based on a self-organizing net (see Appendix A) which uses the proprioceptive information to chose a specific subnet to activate to finalize a motor task.

The reasons that motivate the study of the human motor control and therefore this work are:

- The base cognitive research: to study and to try to comprehend the device that in nature support the behaviour and the intelligence by means of the modelling of artificial intelligent system that try to reproduce these devices. The interest is related to the possibility to observe from a different point of view and exploiting mathematics tools, the phenomena of the complex dynamics system which can explain the functioning of the human motor control.
- The research of a specified artificial intelligent system which could be a help for the functional recovery of the stroked patients.

## **State of the art**

The movements studied in this work are denoted as ballistic [43], and they have been extensively studied for over a century, even if a unique theory regarding the planning of these movements is still absent. The quarrel is on the relative importance of

sensory feedback for online corrections during fast and goal-directed movements [<sup>44</sup>][<sup>45</sup>]. While Plamondon (1995) stressed the absence of feedback contribution during the movements, Elliott's experiments (1999) stated that motor commands can be adjusted online without the necessity to involve a conscious decision process, and thus outrunning delays specifications. It is, however, commonly agreed that, especially in absence of environmental changes, this contribution is minor with respect to the pre-planned control. In order to optimize movement capabilities and extend the possibilities of motor learning, nature provided the human arm with a redundant number of degrees of freedom. As a result, the same motor task can be executed in many different ways. This means that, each time a movement is produced, the sensorimotor control must have selected one of the countless possible strategies to achieve that motor goal [<sup>46</sup>]. Nevertheless it is possible to observe not only intra-subjective but also inter-subjective invariants in fast reaching arm movements, e.g., paths roughly straight and bell-shaped hand speed profiles [<sup>47</sup>][<sup>48</sup>]; moreover speed profiles are also invariant with regard to the spatial extent or amplitude of the movement [<sup>49</sup>]. Among all the implications that this characteristic, defined as "scale effect", highlights, it is likely that the hand trajectory planning could be unconcerned with respect to the acceleration of the movement; this could be a mechanism used by the CNS in order to simplify the elaboration of the motor commands. The movement of the hand tend to follow a roughly straight line. Another invariant aspect is that the planar ballistic movements are practically without discontinuity.

. Some authors [<sup>50</sup>][<sup>51</sup>] tried to provide a mathematical explanation of these kinematic invariants suggesting the hypothesis that the central nervous system aims at maximizing the smoothness of the movement. the end-effector velocity in ballistic movements is typically bell-shaped.

For what concerns the biomechanical model there are many examples in literature of artificial upper limbs that have been used in order connect a plant to the specific controller presented. The one on which this work is based is the model presented in [<sup>52</sup>]: it includes a 2DOF manipulator driven by three muscle couples.

A lot of research has been done on using feed-forward neural networks as the adaptive component in a learning controller [<sup>53</sup>]. The network weights can be adjusted using the backpropagation algorithm, genetic algorithms [<sup>54</sup>], or various stochastic search

algorithms (for example, statistical gradient following [<sup>55</sup>]). Supervised training is usually performed using error signals derived from the system's performance error, although other approaches which transfer expert information from a rule base are common.

Several control approaches have been developed which perform training on the system with its controlling neural network unfolded over discrete time. Backpropagation through time [<sup>56</sup>] propagates error information backwards through time. Such algorithms can also train recurrent neural network controllers that have their own dynamical properties. These algorithms have been generalized to continuous systems [<sup>57</sup>]. Miller [<sup>58</sup>] has extended the backpropagation through time approach so that error information is also propagated through a custom-built central pattern generator (CPG). Judicious choice of the CPG circuit can improve the performance and stability of learning simple motor tasks. Although theoretically elegant, forward and backward propagation approaches are ill suited to practical on-line control. Others have used a more successful analytical control-theory approach to train a neural network so that it becomes an inverse (in some sense) of the system being controlled [<sup>59</sup>] [<sup>60</sup>] [<sup>61</sup>]. Anyway most of the models present in literature are based on learning algorithms which need the use of training example or in which the controller is directly connect to the arm model and whose output are the torque values to drive it.

A comprehensive neural-based model of the human arm has been implemented by Karniel and Inbar (1997). It includes a 2DOF manipulator driven by three muscle couples, for the biomechanical arm modelling, and an ANN and a Pulse Generator (transforming the neural outputs into representative motor commands) for the CNS functionality synthesis. The results obtained are consistent with physiology although the movements are restricted to a tiny region of the entire workspace and the learning algorithm is not biologically plausible as much as the model. The authors stressed that the model could be improved by optimising both the learning scheme and the number of neural outputs.

## **Organization of the work**

The rest of this dissertation is organized as follows: in Chapter 2 the base structure of the system is presented. The neural controller, the pulse generator and the biomechanical arm model. Each module is described highlighting its features and functionalities in the perception-action process of the movements.

Chapter 3 firstly introduces the novel learning paradigm. The development process of the neural controller is explained showing the single steps composing the exploration phase. In the second part a more complicated system structured as a hierarchical controller is presented. Finally the test that have been carried out on both the controller (the simple one and the hierarchical one) are presented: a specific part is dedicated to the test of the adaptability of the neural controller to the presence of external forces acting on the end-effector.

Chapter 4 shows all the results of the test performed. The results are compared with the data extracted from the literature and related to similar tests carried out by human subjects.

In Chapter 5 is presented a first application of the neural controller. An application with rehabilitative aims based on a FES system driven by an intelligent connectionist model.

In Chapter 6 the conclusions are reported.

# **Chapter 2**

## ***Neural Controller of a Biomechanical Arm Model: Materials and Methods***

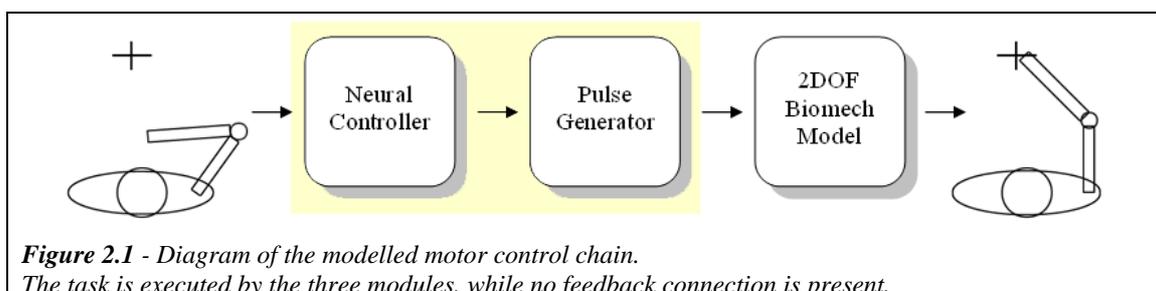
## Summary

In this section the mechanical model of a human upper limb which has been implemented in the work is briefly introduced, and the project and the development of the Artificial Neural Network used as controller of the effector is presented. Moreover the design of three main neural structures used to face the problem of simulating a biological controller with respect to different analysis are presented:

- Analysis of point-to-point reaching movements.
- Analysis of motor control in presence of environmental distortions
- A hierarchical structure: from an exploration learning approach to a coarse to fine learning approach.

## The proposed Model

The general scheme of the proposed model is shown in figure 2.1. The entire model can be divided into three main modules, each one with a specific functionality in the transformation process from perception to motor action, that is: the perception task, the elaboration of data and the motor activation. The first two computational blocks represent the plant for the motor control of the upper limb, while the third block is responsible for the modelling of the actuator (i.e. in this implementation a biomechanical arm model).



The first module is devoted to process the spatial information in order to solve the inverse dynamics problem, that is answering to the question "which neural signals, that is which forces, have to be generated to reach a specific point in the environment?". The strategy can be mastered after a series of synaptic modifications that represent the construction of the internal model both in architectural and in functional ways. The whole process that simulates the generation of the internal models by means of synaptic modifications is called learning.

The second module, called Pulse Generator, generates the motor signals necessary for the muscle activation and consequently for the generation of the movements of the arm model.

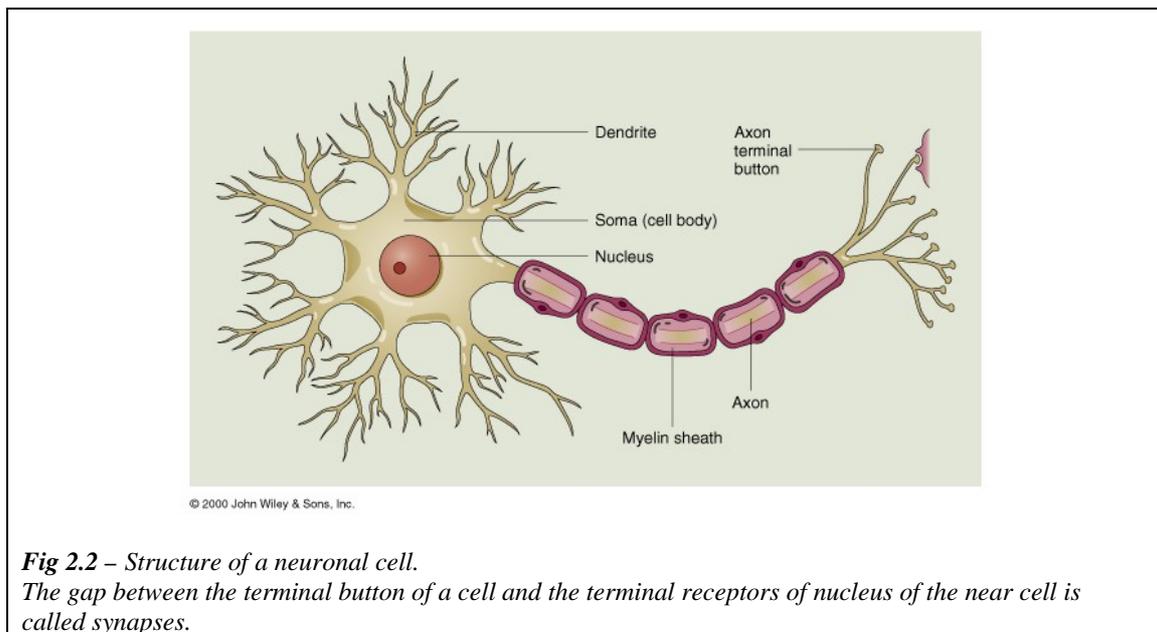
The third module which includes the scheme of the control flow, simulates a simplified version of the biomechanical arm model. In fact, the human arm presents a high number of degrees of freedom and a redundancy due to the difference of dimensions between muscular activations space and working plane space (that is the whole set of the points attainable by the arm model), so that the set of available ways to accomplish a specific task is not unique. In the proposed model, two mono-articular pairs of muscles for each joint (elbow and shoulder) and a bi-articular pair of muscles connecting the two joints were considered as relevant to the execution of planar movement, and thus taken into account.

It must be emphasized that, since the main purpose of the present work is to characterize a model simulating the generation and the actuation of ballistic movements, no feedback signal on the position error is present in the scheme. As a matter of fact, the model deals with a process where the learning scheme modifies the neural features in order to map the working space and reach the desired targets. Even if the learning scheme can be considered as a functionality of the Neural System, the Chapter 3 has been devoted to the explanation of the learning process in order to outline the adopted processing scheme.

# Neural Network Controller

## Introduction

Two are the basic characteristics of the human brain: the “plasticity” of the neural connections which can be modified by means of the interactions with the environment and through the experience, and the ability to break down the acquired information. Each neuron is connected to thousands of other units establishing a connection from the soma through specific connections (axons) to the dendrites (see figure 2.2).



The whole system of “communication” and “activation” of the neurons is based on electrochemical processes that involve the difference in electrical charge of the membranes [62]. The variation of the potential between inside and outside the nucleus can be transmitted along the axon to the next neuron; anyway, while propagating through the axon, this potential difference can possibly become smaller, following the law:

$$\frac{\partial V}{\partial t} = \frac{\partial^2 V}{\partial x^2} \quad (1)$$

This means that the potential decays exponentially having the value  $V_{(x)} = V_0 e^{-x}$ . If the total potential difference (i.e. the sum of all electrochemical signals deriving from other units) that reaches the dendrites of a neuron, is large enough to exceed a set threshold (activation level), a new pulse can be generated. Each kind of neuron has its own activation level and it's just this value which determines the dynamics of the reinforcement or the weakening of the synapses, thus influencing the process and the memorizing process.

The real neuronal units show several benefits:

- Real neurons show a slow activation time; the processing time of a modern processor is greatly smaller than the processing time of a neuronal cell. Nevertheless, the brain is able to solve extremely complex vision and language problems in less than 500ms; this is due to the high interconnectivity of all the neurons, which can perform a real parallel distributed elaboration of the data.
- The thermal energy dissipated by a neuron in a elementary calculus operation is about  $3 \times 10^{-3}$  erg that is about ten order of magnitude lower than a transistor.
- The high redundancy of the neural structures allows the brain to be an highly flexible system and to overcome local failures without a significant loss of performance.

The Artificial Neural Networks (ANN) are computational models whose purpose is to design the way in which the brain performs a particular task. Indeed, a neural network is composed by a number of linked units through weighted connections, just as the human neuronal structures. The development of the ANN<sup>1</sup> derived from the attempt to simulate the nervous structures of the brain tissue; the original idea undoubtedly

---

<sup>1</sup> The majority of this paragraph is taken by (Ben Krose, Patrick van der Smagt “An introduction to Neural Networks”, 1996)

derives from the studies on the central nervous system, and still today most of the research activity follows that direction. It is possible to assert that the ANN can be characterised as ‘computational models’ with particular properties such as the ability to adapt or learn, to generalise, or to cluster or organise data, and which operation is based on parallel processing.

## **Historical Background**

The birth of this idea dates back to 1943 when McCulloch and Pitts, combining elements of neurophysiology and mathematics, modelled the neuron as a binary discrete-time element. During the end of the fifties two different kinds of artificial neural network were introduced. These ones would have had a great impact on the development of the actual neural structures in the following decades. In 1958 Rosenblatt introduced the Perceptron, using a linear function activation, while in 1960s a second structure, called Madaline (Widrow-Hoff, 1960), was implemented with a continuous activation function. Before the end of the sixties the interest for these new mathematical instruments decreased, due to their structural design limitation. In fact, it was demonstrated that ANN could solve problems concerning only linearly separable data (Minsky – Papert, 1969). In 1985 there has been a renewed interest on the artificial neural networks, when it was demonstrated that the previous limitation could be overtaken by using learning schemes for multilayer structures, thus improving the interest to apply them on different scientific and economic fields.

An important stimulus derived from the work of Rumelhart (1986), who introduced the Generalized Delta Rule, whose implementation simplicity is based on two main steps:

- During the first phase of elaboration, the input spreads forward to the output units.

- During the second phase, an error signal spreads backward through the network and it is used to refresh the weights of the internal connections. From this second phase the definition of the Error Back Propagation (EBP) algorithm came (See Appendix A for details).

Nowadays it is widely acknowledged that the artificial neural networks are powerful tools especially for pattern recognition problems (given an input, the net is able to analyze it and to provide an output corresponding to a specific and significant classification), non-linear control and data processing.

The neural networks represent an alternative computational paradigm to the conventional computational methods, but the basic concepts of the neural networks can be understood following a pure abstract approach starting from the information processing.

A feedforward neural network can be seen as a mathematical function which transform a set of input variables to a set of output variables. The exact form of the transformation is defined by a set of parameters called weights (which are the artificial proxies of the synapses), whose values can be estimated on the basis of the examples of the linear or non-linear function that has to be modelled. The evaluation process of the weight values is defined *learning* or *training* and it is the most computationally onerous part. Once the weights are fixed, the new data can be processed very fast. The main drawback of the artificial neural networks is the necessity to be fed by a set of uniformly distributed data in the solution space, otherwise both the extrapolation, and the interpolation of values in the output space can bring to a not perfect computational efficiency. One of the best characteristics of the neural network is the capacity of generalization, that is the ability to classify patterns never analyzed previously. This is a great advantage with respect to a simple associative memory; the real training should allow the network to predict answers [<sup>63</sup>].

The advantages of the artificial neural networks are:

- Adaptive Learning: the ability to learn how to achieve particular tasks based on the training data.
- Self-Organization: an ANN can create its own organization or representation of the information it receives during the learning phase.
- Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.
- Fault tolerance via redundant information coding: partial destruction of network leads to the corresponding degradation of performance. However, some network capabilities retained even with major network damage.
- Non-linearity: a neuron is basically a nonlinear device and an artificial neural network can approximate non-linear mapping; with respect to other techniques they need less parameters, and since they allow a multiple input-output architecture, they can be implemented in multi-variable systems control.
- Neurobiological analogy. Neurobiologists look to Artificial Neural Networks as a research tool for the interpretation of neurobiological phenomena. Engineers look to neurobiology for new ideas to solve problems more complex than those based on conventional hard-wired design techniques.

## Structure of the Neural Controller

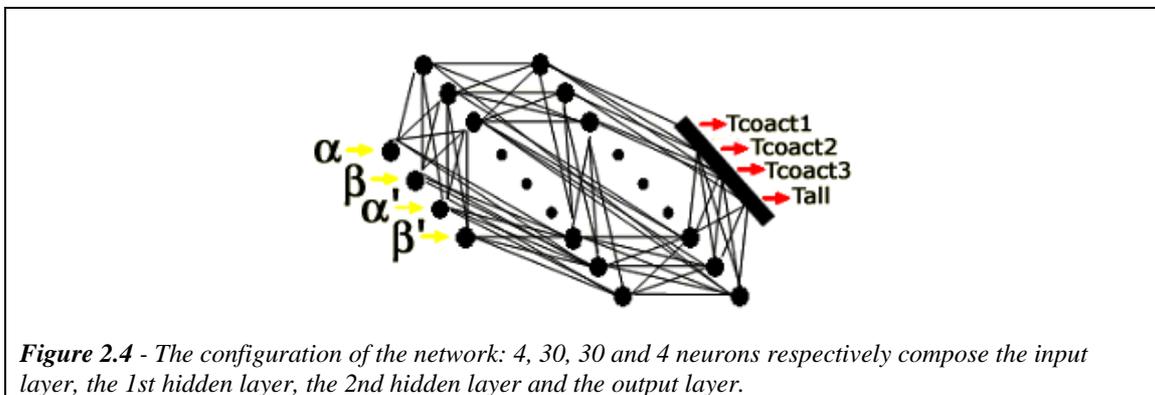
The first module of the implemented system (see Figure 2.1) has been structured as a Multi Layer Perceptron with an architecture composed by 4 layers.

The *design process* of the neural network used for this study is based on the analysis of the behaviour of various neural structures when fed by the same training and testing sets. In the early phase of this study an algorithm able to generate a set of about 300 associations of starting/ending points of a planar trajectory, together with the respective parameters which allow the biomechanical arm model to follow the trajectory, has been developed. The generated data set contained examples of almost all the kind of movements inside the working plane. Subsequently, in order to choose the most adequate structure, different types of neural networks have been considered and trained: a first ensemble composed of ANNs with only one hidden layer (varying the number of neurons), and a second group composed of ANNs with two hidden layer (varying the number of neurons in different combinations for each layer). Experimental results considering errors with respect to the training set and to the testing set as cross-validation (in order to avoid over-fitting problems) led to choose an ANN design with two hidden layer of 30 neurons each.

In fact, one of the deficiencies of the back-propagation algorithm is connected to the number of the hidden units. Two neural network with a different number of hidden units can approximate the same function, but they could behave in a different way. A network with too many units can fit exactly with the learning samples, but because of the large number of the hidden units with respect to the problem, the representation of the function could differ from the original one: this effect is called overtraining. In case of learning samples containing a certain amount of noise, the network will most probably “fit the noise” of the learning samples instead of making a smooth approximation. The example shows that a large number of hidden units leads to a small error on the training set but not necessarily to a small error on the test set. Adding hidden units always leads to a reduction of the learning error, but at the same time it is possible that the error on the testing set could increase.

Nevertheless the variation of the number of the units for each layer within a certain range shows less influence on the resolution of the problem, but shows a great effect on the computational cost and, therefore, on the learning speed.

The input layer of the implemented neural controller is therefore defined by 4 input units, which correspond to the coordinates of the joint angles of the arm both at the starting and at the final points of the movement, and represent the set of the proprioceptive information on the position of the arm within the working plane (see figure 2.4).



In particular, the first 2 units are related to the information on the angles of shoulder and elbow joints in the initial and final position of the trajectory, while the other 2 units are related to the same information in the desired final position.

The output layer has 4 units, according to the following principle: the neural network generates one value of timing for each of the muscular pairs related to shoulder and elbow and the one connecting both of the joints, plus one value shared by all the muscular pairs, that is  $T_{coact-shoulder}=T_{coact1}$ ,  $T_{coact-elbow}=T_{coact2}$ ,  $T_{coact-biarticular}=T_{coact3}$ ,  $T_{all}$ .

More specifically:

- for the shoulder, when the agonist muscle is activated, the movement starts. After a time interval defined by the ANN, the antagonist is

activated, so that the time interval  $T_{\text{coact-shoulder}}$  characterizes the co-activations of the agonist and antagonist monoarticular muscles of the shoulder joint (i.e. simultaneous presence of the neural inputs for shoulder muscles); its sign defines which muscle (i.e. agonist or antagonist) is activated first;

- for the elbow,  $T_{\text{coact-elbow}}$  has the same function of  $T_{\text{coact-shoulder}}$ ;
- for the muscle pair connecting the two joints,  $T_{\text{coact-biarticular}}$  has the same function of  $T_{\text{coact-shoulder}}$  and  $T_{\text{coact-elbow}}$ ;
- the movement duration is  $T_{\text{all}}$ : it represents the total duration of the neural activation, thus affecting the whole movement of the arm. This output value is constrained in the range 300ms – 1s.; the time range has been chosen in order to let the limb model reach every sector of the environment where it operates, while maintaining the ballistic characteristics of the movement.

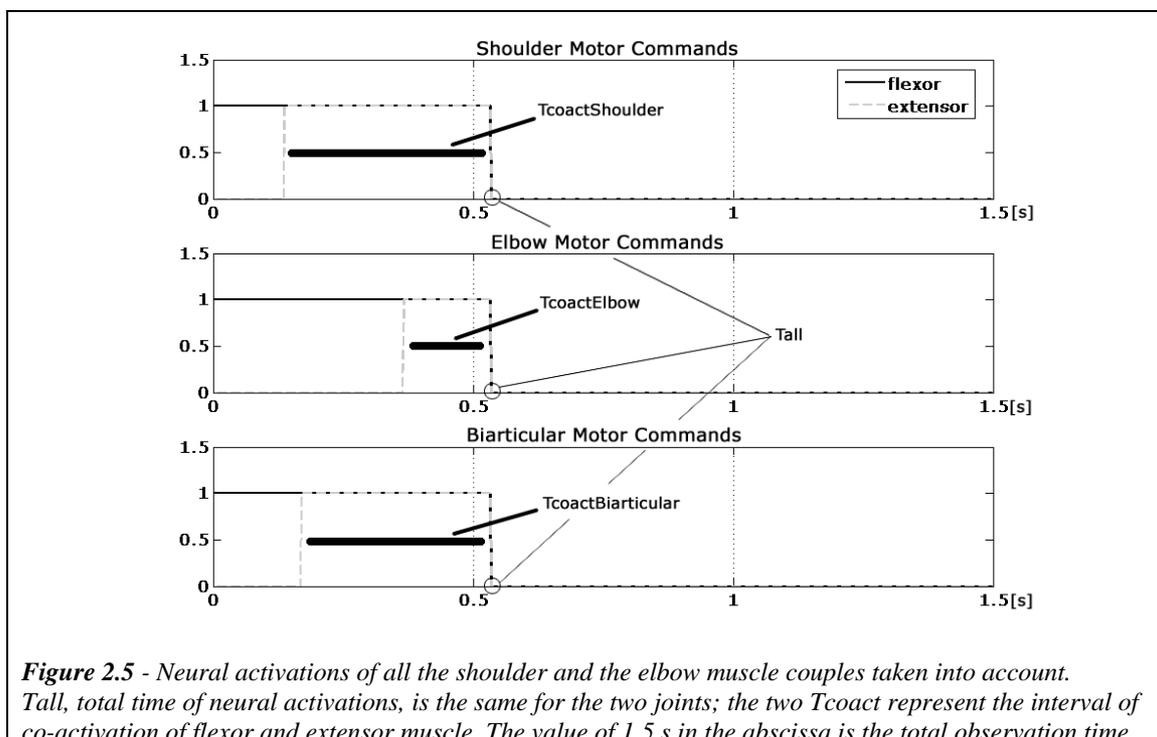
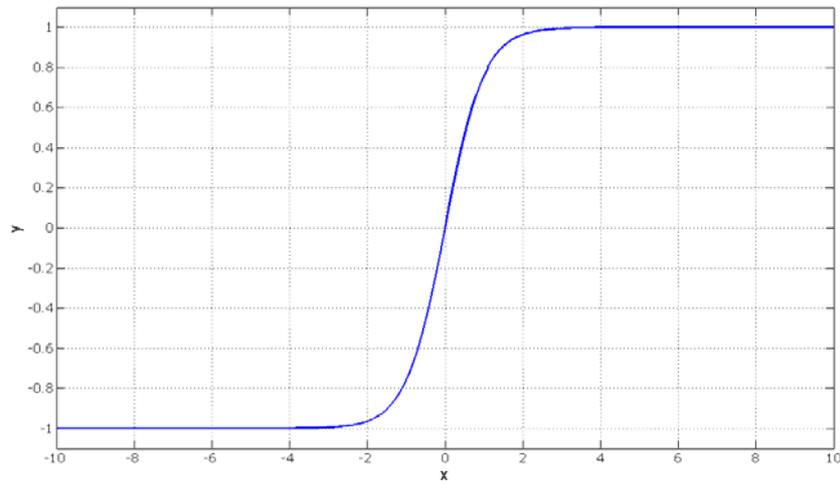


Figure 2.5 depicts the profile of the neural activations, as it will be built by the Pulse Generator, having rectangular shapes, and shows the duration of the entire voluntary task ranging in the interval 300ms – 1s. The transfer function chosen for every unit is the hyperbolic tangent (graphical representation in figure 2.6): the output  $n_i^m$  of the  $i_{th}$  neuron at the  $m_{th}$  layer is obtained from the weighted outputs of the  $(m - 1)_{th}$  layer, according to equation 2:

$$n_i^m = \frac{2}{1 + e^{-\sum_{j=0}^{N_m} w_j^{m-1} \cdot n_j^{m-1}}} - 1 \quad (2)$$



**Figure 2.6** – Graphic representation of tansigmoid function (tansig.m MATLABR2006©). The  $y$  is the output of the single neuronal unit  $n_i^m$ , while the  $x$  corresponds to the sum of the inputs coming from the

units connected to this one:  $\sum_{j=0}^{N_m} w_j^{m-1} n_j^{m-1}$

The values generated by the output layer, from now on indicated as neural outputs  $p$ , are limited in the range  $[-1,1]$ , and are used by the Pulse Generator module.

## The Pulse Generator

The Pulse Generator would model the role of the motor-neurons connected to the muscle fibres. It transforms the efferent signals from the “brain neurons” (the neural controller) to specific activation commands for the muscular units

From the study of the envelope of specific surface EMG acquisitions and from the data of literature [<sup>64</sup>], it has been possible to observe that the burst activations related to the contraction of the skeletal muscles of the arm during fast planar reaching movements show a characteristic and recurrent scheme: the agonist muscle shows a first activation whose amplitude tends to decrease while the action of the antagonist places upon it. Other authors [<sup>65</sup>] have verified that it is possible to have a last activation of the agonist in the end of the movement, with a low intensity. The last kind of muscular activation pattern is defined ABC, in relation with the three phases: the activation, the braking and the clamping in opposite to the first hypothesis of a AB pattern.

The system, in the present version, allows having biphasic activation patterns for each muscle pair. Thus, the interval delimited by the initial point of the pattern and the Tcoact-Shoulder, the Tcoact-Elbow and the Tcoact-Biarticular values represents the Action Pulse, i.e. the time in which the neural activations of the agonist muscle determine an activation in the sEMG signal, while the one going from this value till the end of the pattern, i.e. the time at which the neural activations of the agonist muscle determine an activation of the sEMG signal, while the one going from this value till the end of the pattern, i.e. the time at which the neural co-activations of the antagonist muscle correspond to a braking burst in the sEMG signal, is to the Braking Command. The final coactivation of the agonist and antagonist muscles of each joint determines the limb stability.

The range of these intervals, including the coactivation time of the shoulder and the elbow muscles, together with the entire duration of the activations, establishes the direction, the length and the curvature of the movements. The choice of using only four

parameters as input data for the Pulse Generator module depends upon the requirement of limiting the computational complexity of the neural network by reducing the number of output units. In any case, it has been shown that this setting, even if seemingly oversimplified, is fairly adequate to deal with the motor control because it allows to represent movements which correspond to a complete coverage of the working plane during the exploration phase while, at the same time, maintaining accuracy in the movement and speeding-up the neural network training process.

## **Implementation of the Biomechanical Arm Model**

In order to analyze specific motor tasks performed by the arm, as planar ballistic movements, it is necessary to give a brief description of the characteristics of the limb. It has to be considered as an active element whose mechanical properties allow a simple elaboration of the data for the central nervous system [66]. The arm under the control of the efferent signals of the brain moves and changes its pose. The generation of the voluntary movements by the central nervous system derives from the elaboration of a complex informative process divided in three levels: 1) the determination of the desired trajectory; 2) the transformation from the visual coordinates to the body coordinates; 3) the generation of the motor commands which carry out the trajectory. [48] [3].

The first two levels are related to the kinematics of the movement: position, velocity, and acceleration. The creation of the motor commands patterns, with the corresponding forces and torques applied to the joints, refers to the dynamic aspects of motor control. Kinematics and dynamics of this system was widely explored in robotics on the basis of mechanical theory of rigid body. Since the aim of the present work is directed to the motor control of the upper limb, it is necessary to take into account the basic anatomical structure of the musculoskeletal system in order to achieve an optimal model for the specific tasks that have to be analyzed.

## Skeletal Structure of the Model

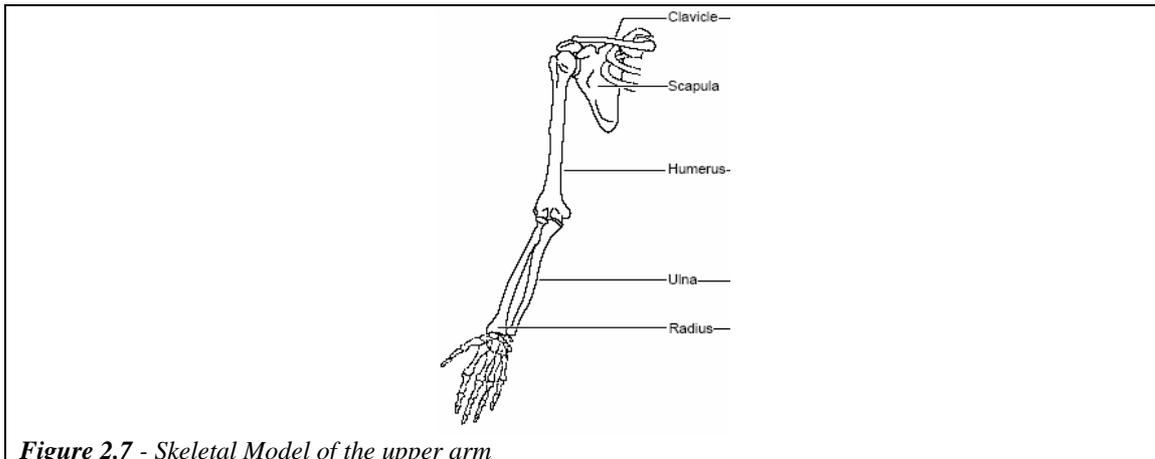
The human skeleton system is composed of bones and joints organized as an articulate structure. It defines the general shape of the body and of the single parts composing it. The bones of the skeleton are connected to the articulations which allow the respective movement. The association of the chained mechanisms including the shoulder, the elbow and the wrist allows a wide range of combined motion conferring to the human arm an extreme mobility [67].

The upper limb is composed of 4 sections (see figure 2.7):

- the shoulder, in which the clavicle and the scapula are posed;
- the arm in which only one long bone, the humerus, is present;
- the forearm, which is composed by two long bones: the ulna and the radius;
- the hand, composed by a more little bones, divided in carpus, metacarpus and phalanxes.

Since the simulation of fast reaching point to point tasks is the principal aim of this study, the hand was simply considered as the end point of the forearm, thus not taking into account any movement of the wrist joint.

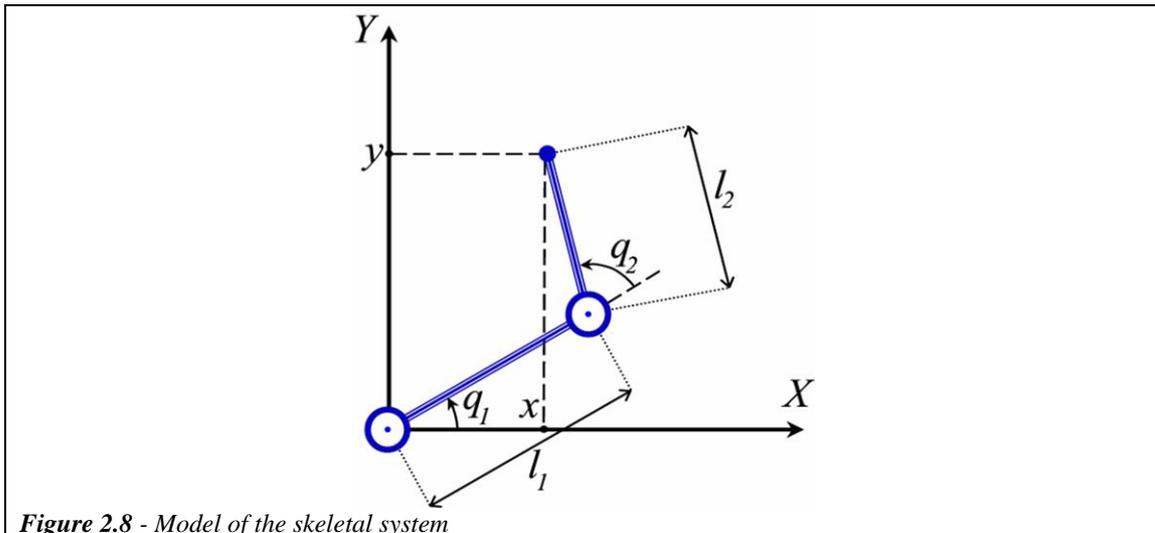
Among the seven joints that can be considered as a base architecture of the human arm, the sterno-clavicular joint, the acromio-clavicular joint, the scapulo-thoracic joint and the humero-radial joints haven't been taken into account. Instead, for a proper model of planar arm movements, the gleno-humeral joint and the ulno-humeral and humero-radial joints are fundamental, and both have been modelled as hinges. The first one allows the humeral head to rotate in the glenoid fossa of the scapula [68], while the other two articulate both ulna and radius on the distal end of the humerus. In this way, the possible movements of the simplified forearm and upper arm model were: shoulder flexion and extension, and elbow flexion and extension.



**Figure 2.7** - Skeletal Model of the upper arm

The articular groups of the ligaments and of the muscles allow the humerus to achieve adduction and abduction movements. The upper extremity of the humerus is connected to the shoulder joint and the lower extremity defines, together with the radius and the ulna, the elbow joint. Considering the motor control of movements on the transversal plane, it has been possible to develop and implement a schematic model of the upper limb.

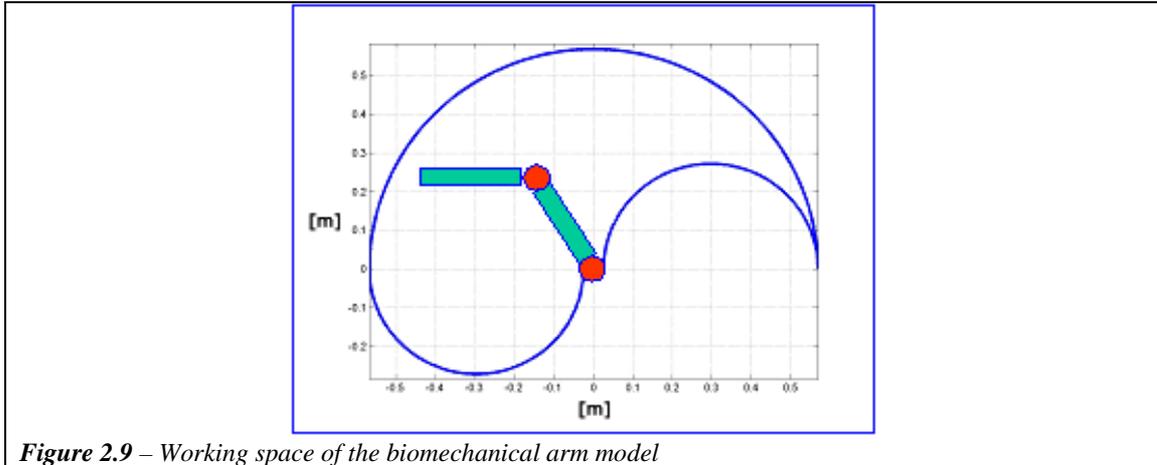
As generally simplified in motor control upper arm modelling [52] [69], the skeletal model has been modelled by means of a plant structure composed of two segments approximated by rigid cylinders, with lengths  $L_1$  and  $L_2$ , which represent the forearm and the upper arm respectively, connected with two joints assumed as ideal hinges (see figure 2.8).



A structure like this shows problem of redundancy: mathematically a point within the working space can be reached by two different configurations of the linked system. Since one of the purposes of the project is to simulate the human perception-action chain by using biologically plausible models, the problem of the implicit redundancy in a 2DOF robotic manipulator has been avoided by imposing the joints to move in the range  $[0,\pi]$  interval, as the human joints do. These values uniquely identify the Cartesian coordinates of the free end in the working plane by means of direct kinematics transformation (equation 3).

It has then been possible to define a working space where the model could operate from this kind of manipulator scheme and from the combination of all the possible values assumed by the angles  $q1$  and  $q2$ ; figure 2.9 depicts the “working area”.

The origin of the axes of the graph represent the fixed position of the shoulder joint.



The central nervous system is responsible of solving the kinematics inverse problem that is of predicting the particular muscle lengths and joint angles corresponding to a specific hand position in the space. Even if the transformation could be included as an implicit transformation calculated by the neural controller, by following the studies of [70], the proprioceptive input is given as the starting and final position in joint coordinates.

$$\begin{aligned}
 x &= l_1 \cdot \cos(q1) + l_2 \cdot \cos(q1 + q2) \\
 y &= l_1 \cdot \sin(q1) + l_2 \cdot \sin(q1 + q2)
 \end{aligned}
 \tag{3}$$

Body segment anthropometrics and inertias of both upper arm and forearm are obtained from the scientific literature [71], taking into account different body heights and weights.

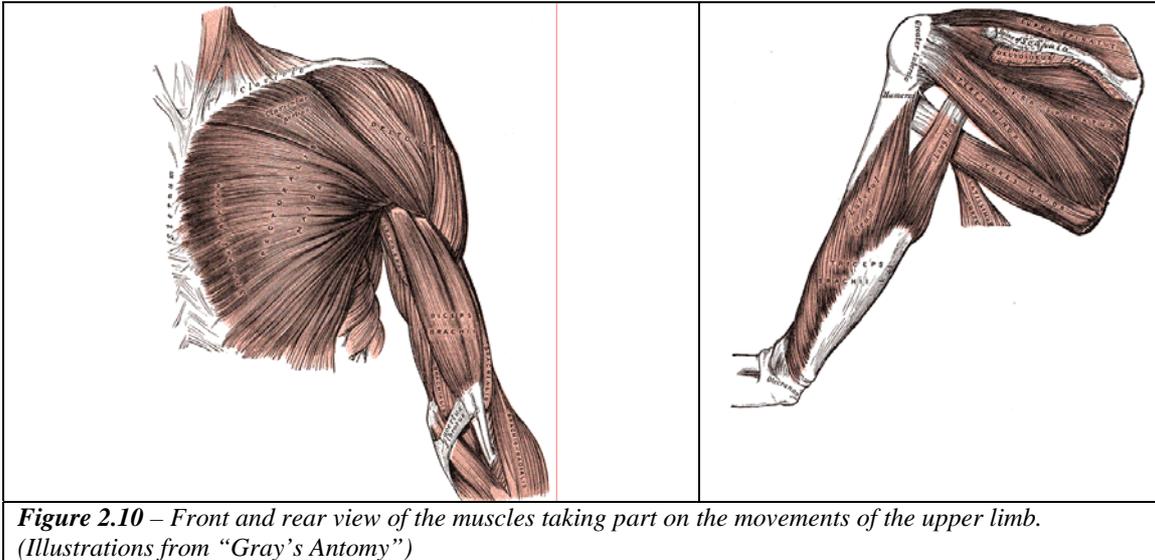
Table 2.1 shows the values of the inertias of the muscular-skeletal system.

<b>Parameter</b>	<b>Units</b>
M - Mass of the subject	80 kg
M1 – mass of the upper arm	2.24 kg
M2 – mass of the lower arm	1.92 kg
L - height of he subject	1.70 m
L1 – length of the upper arm	0.297 cm
L2 – length of the lower arm	0.272 m
I1 - inertias of the upper arm	$M1*(0.322*L1)^2$
I2 - inertias of the lower arm	$M2*(0.468*L2)^2$

*Table 2.1 – Numerical values of the parameters of the arm*

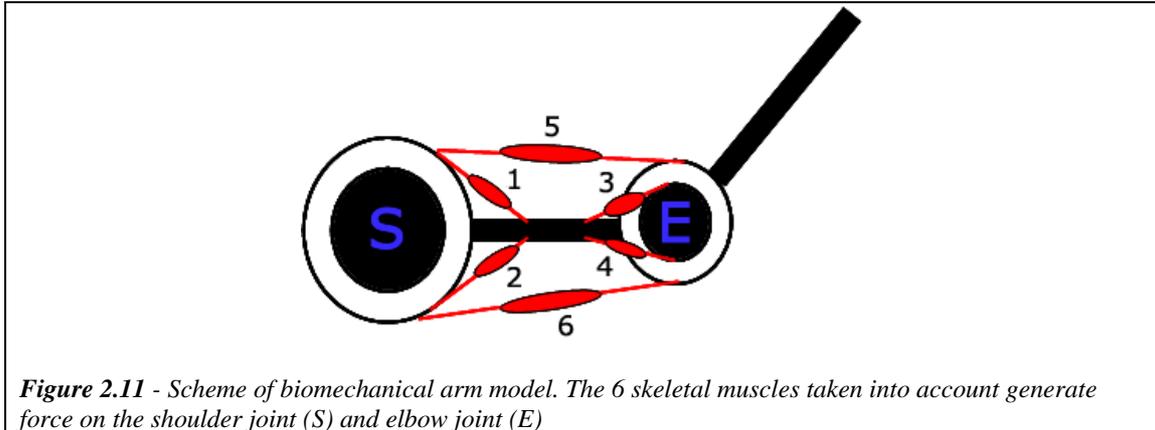
## Muscular Structure of the Model

The muscle model is necessary to calculate the forces acting on the arm joints and to take into account the effect of the dynamics. In the human body it is possible to have many muscles that show their effect on a single articulation; the complexity derived from the redundancy of the musculo-skeletal system and the fact that for the specific type of movements under study a high precision control is not necessary, allows to deal with the modelling of the muscles with some simplifications.



A model taking into account 6 muscle has been analyzed (see figure 2.11):

- a muscle couple acting on the shoulder joint. The flexor muscle is the pectoralis major (pectoralis clavicular head) [1] while the extensor is the posterior deltoid [2] (mono-articular of shoulder) ;
- a muscle couple acting on the elbow joint. The flexor is the biceps brachii long head [3] while the extensor is the triceps brachii lateral head [4] (mono-articular of elbow);
- a couple acting on both the joints. The flexor is the biceps brachii short head [5] and the extensor is the triceps brachii long head [6] (bi-articular of shoulder and elbow);



In the joint space the dynamical model, based on the Lagrange equations, is the following:

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + F_v\dot{q} + f_s(\dot{q}) + g(q) = \tau \quad (4)$$

For the development of a biomechanical arm model simulating rapid planar movements, we can assume that the term corresponding to the gravitational force can be removed; moreover, assuming the rotoidal joints as ideal articulations, it is also possible to neglect the terms deriving from the torque of static friction ( $f_s$ ) and viscous friction ( $F_v$ ).

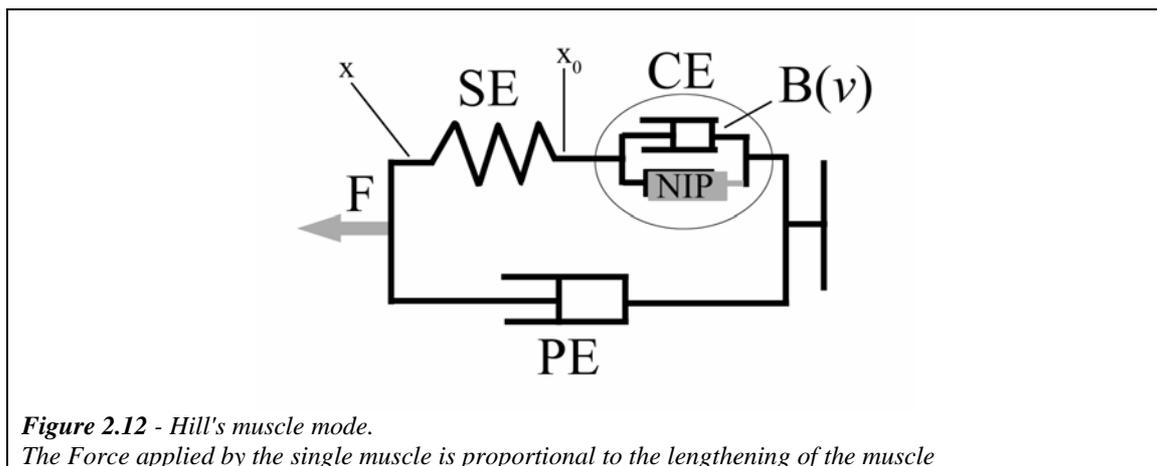
In order to determine the position, the velocity and the acceleration attained by the arm model along a trajectory it is necessary to solve the problem of the direct dynamics, that is to determine the value of  $d\mathbf{q}(t)/dt^2$ ,  $d\mathbf{q}(t)/dt$ ,  $\mathbf{q}(t)$ , from the knowledge of  $\mathbf{q}(t)$ ,  $d\mathbf{q}(t)/dt$ , and  $\boldsymbol{\tau}(t)$  for every  $t > t_0$ .

$$\ddot{q} = B^{-1}q(\tau - C(q, \dot{q})\dot{q}) \quad (5)$$

Therefore the muscular system establishes a dynamic relationship between the position of the arm and the torques acting on each single joint. A key feature of the proposed approach is that an adequate model of the arm of any specific subject can be

obtained and used in the Neural Net. This feature is particularly useful if, for instance, a smart FES system had to be put in place and adapted to the characteristics of a plegic/paretic subject (for further details, see Chapter 6). One crucial problem in a system simulating the motor control of an upper limb is the design of the anthropomorphic model of the arm involving non-linear and time variant properties. Although in a first approximation it's possible not to consider time dependence [69], it's necessary to think of the contribution of the non-linearity. For this reason the Hill's model of the muscle is utilized [72] in order to simulate the six muscle-like actuators on the 2DOF model of an arm.

Following the work of [52], each muscle is synthesized with the non-linear Hill-type lump circuit depicted in figure 2.12



**Figure 2.12** - Hill's muscle mode.  
The Force applied by the single muscle is proportional to the lengthening of the muscle

The rectangular waveforms deriving from the Pulse Generator module serve as inputs for the actuator (*Neural Input Pulses: NIP*), resulting in a time function of the muscle tension  $F$ .

The series elastic element (**SE**) represents the passive elastic muscle property and the parallel contractile element (**PE**) represent the passive viscous tendineous property. Moreover the contractile element (**CE**) includes the non-linear viscosity  $B$  depending on the shortening velocity  $v$ , as in equation 6:

$$B = \begin{cases} (a \cdot T_0) / (b + v) & v \leq 0 \\ a \cdot T_0 & v > 0 \end{cases} \quad a = 4, b = 1 \quad (6)$$

where **a** and **b** are constant parameters, **T<sub>0</sub>** is the value of the torque applied by the single muscular unit derived from the percentage of the maximum isometric force associated to that muscle (equation 7).

$$T_0 = F \cdot F_{\max} \cdot m \quad (7)$$

The equation 6 results in a different behaviour of the contractile element when shortening or lengthening. The viscous muscle element B takes into account the non linearity of the muscle behaviour [69] [52].

Tables 2.2 shows the numerical values of the parameters used for the Hill's model implemented in the project.

Parameter	Units
Kse	120 N/rad
Bpe	30 N.s/rad
M (average moment arm)	0.03 m
τ1	0.01 s
τ2	0.005 s
Fmax(shoulder)	800 N
Fmax(elbow)	700 N
Fmax(double joint)	1000 N

**Table 2.2** – Numerical values of the Hill's parameters

According to [52], the difference of force between the muscles of each single joint is implemented on the actuators by means of different maximal amplitudes of the corresponding forces. The values of the forces are related to maximal values that are represented in Table 2.2.

The process involving the transformations of the neural excitations (NIP) to the associated muscular contraction follows a sequence of numerical integration. The implementation of this kind of model has allowed to gain knowledge on the phenomena concerning the force behaviour during rapid movements and the stiffness variation in the particular case of the presence of force field in the environment. Summarizing the process, the neural inputs ( $n_i$ =NIP) efferent from the Central Nervous System are transformed into EMG envelopes:

$$\overline{emg} = \frac{1}{\tau_1} (n_i - emg) \quad (8)$$

where  $\tau_1$  is the low-pass filter constant. Then the sEMG signal is transformed into the force  $\mathbf{F}$ , that is expressed as the percentage of the maximum force that any muscle can produce.

$$\overline{F} = \frac{1}{\tau_{ac}} (emg - F) \quad (9)$$

Finally, the effects of the obtained torques are summed in order to obtain the overall torques on each joint, as in Equation 10.

$$\begin{aligned} \tau_1 &= F_{1-flex} - F_{1-ext} + \phi \cdot F_{3-flex} - \phi \cdot F_{3-ext} \\ \tau_2 &= F_{2-flex} - F_{2-ext} + \varphi \cdot F_{3-flex} - \varphi \cdot F_{3-ext} \end{aligned} \quad (10)$$

where  $\Phi = 0.6$  and  $\varphi = 0.4$  are non dimensional units and the  $\mathbf{F}$  values in the equation are the values of the torque applied by each muscle of the corresponding joint (1: shoulder; 2: elbow). The results of this modelling demonstrate that, even in this simplified version, the synthesized system is able to execute accurate planar movements.

It is interesting to outline that the first version of the model included only two pairs of monoarticular muscles, while in this one it includes also a pair of biarticular muscles. The training phase has not been increased, thus showing a good scalability of the model.

The overall trajectory in the working plane is obtained from a double integration at each sampling time of the acceleration of the end point of the effector obtained by the changes in the overall torque applied to both joints.

## **Study of Variations of the Hill's parameters**

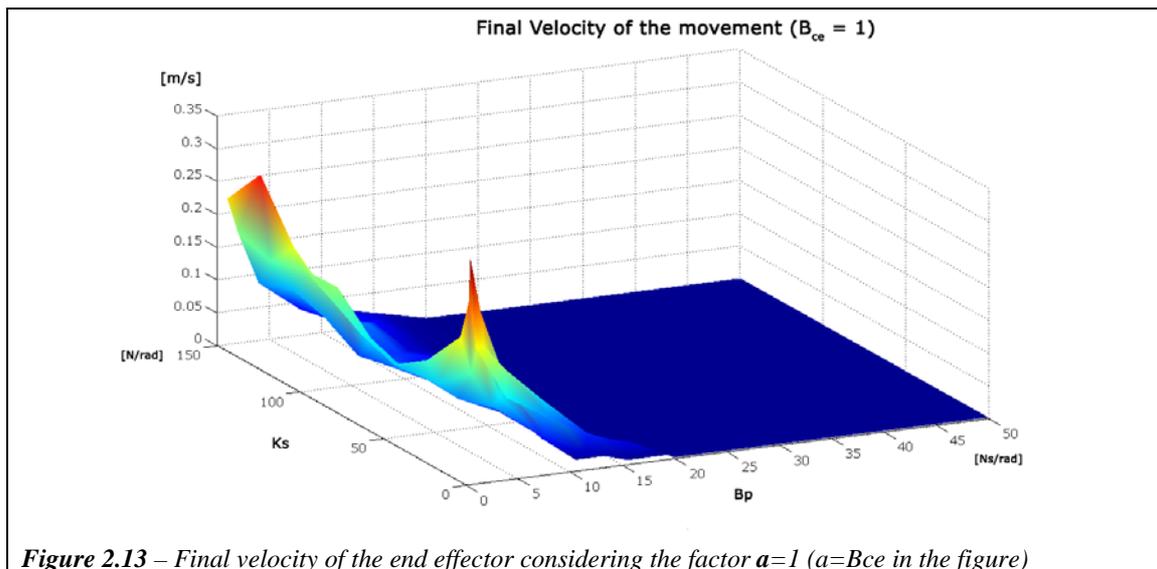
Muscle models play an important role in the study of motor control mechanisms and in the design of motor system neuroprostheses. There are several muscular models differing for structure and complexity that had been studied and implemented [72] [73].

Many musculoskeletal simulations of human movements use variations of Hill muscle models to predict muscle forces, but their sensitivity to model parameter is not well understood; furthermore the parameters which define the behaviour of the active and passive properties of the musculo-tendinous units are usually mean values taken from human and animal muscle experiments [74]. It is important to underline that the parameters related to the Serial Element (SE), Parallel Element (PE) and the force-velocity properties of the Contractile Element (CE) were found to have differing sensitivities, and dependent on the movement that is simulated [75]. The parameters of the Hill's model typically used in [69] for fast planar reaching movements don't allow to achieve a movement being at the same time fast and long.

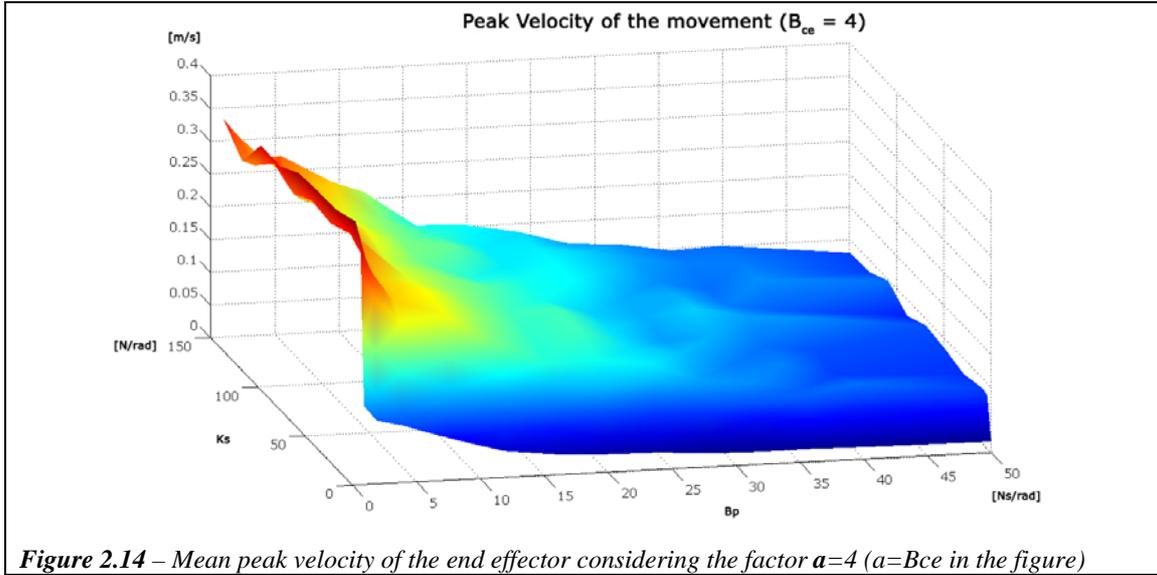
For this reason a study for the assessment of the principal parameters has been carried out. The method used for this study has been directed to discover the optimal values of both the viscous (PE) and elastic (SE) constants and the parameter  $a$  of the non-linear contractile element, in order to make the biomechanical arm to execute movements defined by proper specifications. The  $K_s$  coefficient has been varied in a range varying

between 1 and 150; the  $Bp$  coefficient in the range 1-50; while the values of the factor  $a$  have been chosen between 1 and 10.

For each combination of this triplet, a standard pulse train has been provided to the upper limb model. 10 different starting points within the working plane have been used, and the pulse train varied from 1ms to 10ms. At the end of the movement a kinematics analysis has been carried out. The variables used to make this comparative study were the peak and the mean wrist velocity during the movement, the presence of more than one local maximum in the end-effector velocity profile, the duration and the length of the movement. A movement was considered good if it showed a high peak velocity – length of the movement ratio, a near-zero end velocity value, the minimum number of local maximum in the velocity profile, and a low duration.



**Figure 2.13** – Final velocity of the end effector considering the factor  $a=1$  ( $a=B_{ce}$  in the figure)



This study allowed to find the parameters that optimally fit the Hill's muscle model to face the control of the ballistic movements. The values of  $Ks$ ,  $Bp$  and  $a$  are reported in Table 2.2.

# **Chapter 3**

*Learning Paradigm: Implementation.*

*Hierarchical Neural Controller.*

## Summary

In this chapter the learning paradigm of the neural controller is presented. The connectionists systems are nowadays widely applied in different areas of robotics, and show great capabilities of controlling even complex motor tasks. In most of the cases these systems learn the task in a supervised way, using the feedback loop between the effector and the environment.

In this way the sensorial information are used to minimize the error done while executing the requested task. This methodology, generally applied on forward multilayer networks with the Back Propagation algorithm or its variants, is highly efficient in terms of accuracy and precision of the movement, but at the same time doesn't show a real biological plausibility with the human motor control.

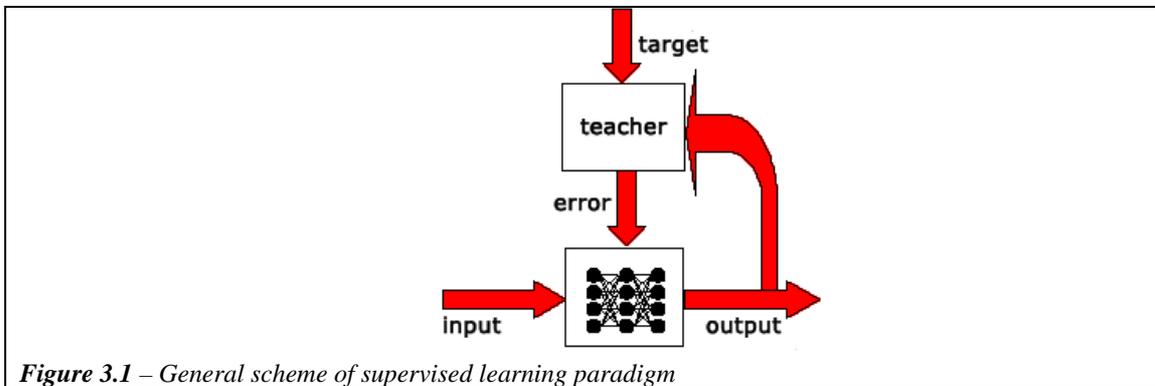
## Neural Network Training Mechanisms

The interest on the training mechanisms of the connectionist models started at the end of 19<sup>th</sup> century, with the first studies on the neural structures of the brain, aiming at understanding if the memorization process took place in the synaptic sites in relation to the learning process, thus showing the plasticity of the nervous system.

From a modelling viewpoint three main training paradigm for the adaptation of the weights of a neural network have been defined: the supervised training, the reinforcement training and the unsupervised training. This classification is based on the nature of the signals driving the training phase: error signals in the supervised training, quality signals in the reinforcement learning and any kind of signal in the unsupervised one.

In the supervised algorithm the presence of a “external teacher” during the training is assumed. This teacher provides the net with the stimulus (or the input) and the corresponding desired output to the network. The error between the network output and

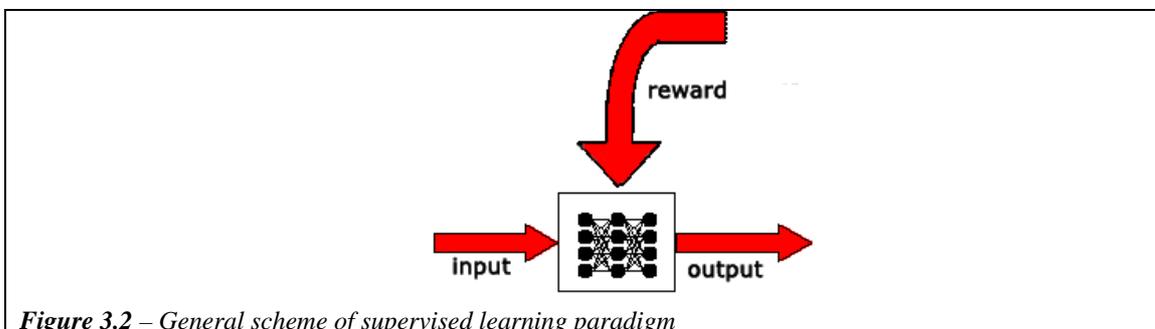
the desired one is used to adapt the synaptic weights in order to reduce the mean error value of the future output (see figure 3.1)



*Figure 3.1 – General scheme of supervised learning paradigm*

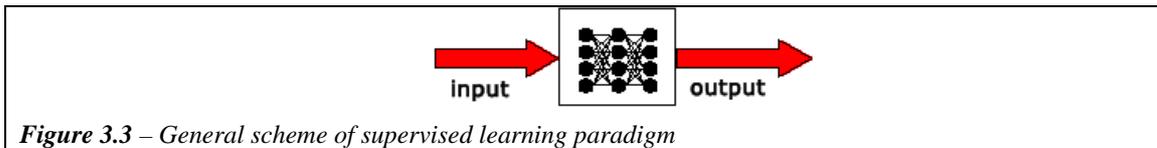
The learning procedure stops when the mean error with respect to the training pattern becomes small enough. This paradigm is usually adopted in non recursive multilayer networks.

If in the supervised learning the aim is to minimize an output error, in the reinforcement learning the objective is to maximize a reward or a reinforce parameter. This parameter must reach a specific value in order to end the learning phase. For each output of the network a new reward is generated, which can be a function of the input, of the output or of the weights connecting the single units. The modification of the weights is evaluated in order to increase the probability of future rewards. This paradigm is extremely useful when a task can be decomposed in different sub-tasks, whose sequence can influence the overall reward value (see figure 3.2).



*Figure 3.2 – General scheme of supervised learning paradigm*

In the unsupervised training, generally used in the Self Organized Maps, only the input set is known. The aim of this learning procedure is not to train an output unit to respond to clusters of pattern within the input., but to discover statistically salient features of the input population. Unlike the supervised learning paradigm, there is no *a priori* set of categories into which the patterns are to be classified; rather the system must develop its own representation of the input stimuli.



All learning rules for both the supervised and unsupervised models can be considered as a variant of the Hebbian rule. D. Hebb, in his book “Organization of Behaviour (1949)”, suggested that if two units  $j$  and  $k$  are active simultaneously, their interconnection must be strengthened. If  $j$  receives input from  $k$ , the simplest version of Hebbian learning prescribes to modify the weight  $w_{jk}$  with

$$\Delta w_{jk} = \gamma y_j y_k \quad (12)$$

Where  $\gamma$  is a positive constant of proportionality representing the learning rate. Another common rule uses not the actual activation of unit  $k$  but the difference between the actual and desired activation for adjusting the weights:

$$\Delta w_{jk} = \gamma y_j (d_k - y_k) \quad (13)$$

in which  $d_k$  is the desired activation provided by a teacher.

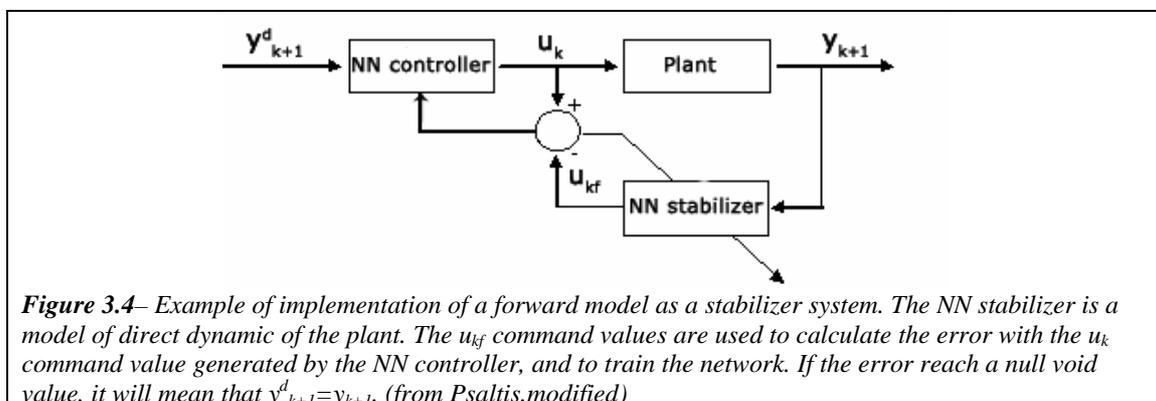
Recent models of motor learning [76] assume that the functionality distinction assigned to these three kind of paradigm could be reported at a physiological level, correlating the supervised learning to some of the cerebellum functions, the

reinforcement learning to the basal ganglia and the unsupervised learning implemented in the cerebral cortex [77]

## Learning Paradigm: Dynamics of the Reaching Tasks

On the basis of these standard training mechanisms different learning and adaptation schemes have been developed. How can they be related with the definition of the internal models? In humans the learning of the motor apparatus is mainly divided into the motor-sensory transformation, or forward model [78], and the sensory-motor transformation, or inverse model [79] [80]. The forward model aims at predicting the behaviour of a dynamic system having as input data the variables necessary to stimulate the system; an example could be the model of dynamic transformation from the forces applied to a specific plant to the action carried out by the controlled object in terms of kinematics variables.

The role of the forward models has been mainly directed to solve the high level problem of the motor planning that is the mapping from joint coordinates to endpoint coordinates; this has been put in evidence also in the works of [81] for the eyes movement control. Its primary use is as a system stabilizer by means of an internal feedback loop [59], like the one depicted in the figure 3.4.



The advantages of having a direct model as a “feedback teacher” is that the weights of the controller tend to be tuned towards the correct solution state, and that it is possible to overcome the sensory motor delays [82]; in effect the causality of the forward models allow them to represent well-defined functions [83]; the main drawback is, of course, that they are not biologically relevant.

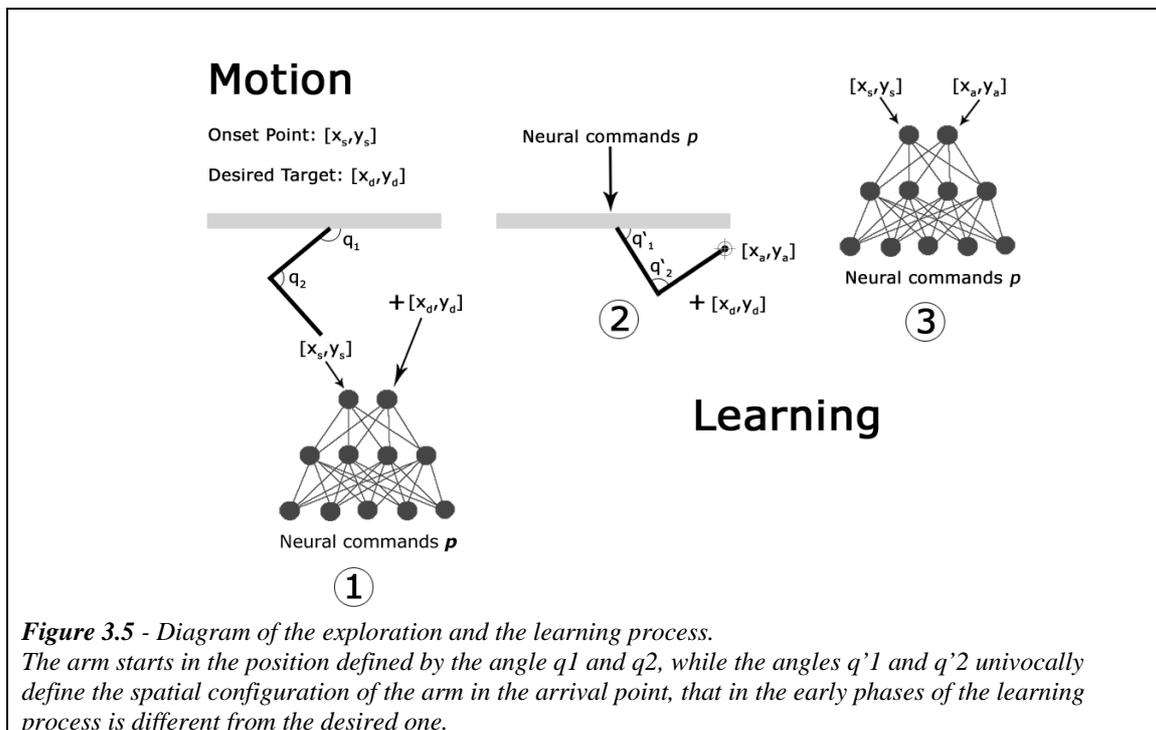
The other kind of control is the inverse model control. It acts in the opposite way the direct model does: in fact, it uses as input the behaviour (the state) of the system and produces the causes that generate that behaviour. This is the basic module in open-loop control schemes including the voluntary ballistic movement, allowing the control system to compute an appropriate control signal without relying on error-correcting feedback [84]. Both forward and inverse models capture aspects of the kinematics and dynamic behaviour of the environment external to the brain [85], and even if their single role in the central nervous system is still controversial, in motor control theories hypotheses both adaptive feedback and feedforward control structures usually work in combination [86][87].

## **Construction of the Internal Model : Biological Learning Paradigm**

One key point of the present work is the set-up of a training paradigm for the neural controller with the aim of defining a specific internal model during voluntary ballistic movements of the arm, that is to establish a correlation map between the desired movements within the working plane and the necessary neural controls, without any a priori knowledge to be inserted into the system. In this way, the controller learns the inverse dynamics of the biomechanical arm model with respect to the interaction with the environment.

The algorithm must adapt the neural weights and biases so that, if the 4 inputs of the network respectively correspond to the coordinates of the starting point  $[\alpha, \beta]$ , and of the desired target  $[\alpha', \beta']$ , then the output of the net must be  $p$ .

More precisely, as shown in the scheme depicted in figure 3.5, the output  $p$  of a non-trained network (phase 1) can be an input for the biomechanical arm model (phase 2); this input allows the execution of a reaching movement different from the desired one, that is towards a different target. However, a key feature is that these neural inputs  $p$  together with the starting and ending points coordinates become the new data for the training of the network (phase 3). In this way, a mapping between muscular activations and points of the working space can be attained.



The aim of the controller is to achieve a complete exploration of the state space: obviously the equations that connect the input space to the output space are not uniquely identified (i.e. considering the time as a variable of the state space), and this precludes any type of enumerative search for a solution; it's not always true that the solution

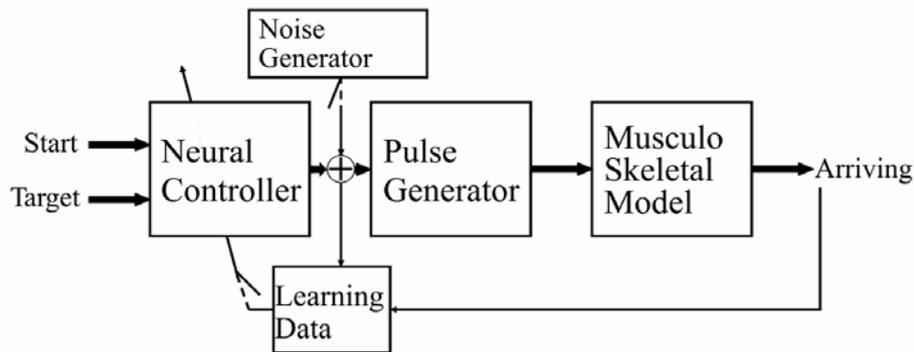
belongs to a whole state space; on the contrary, in many cases the actual problem rests on a lower dimensionality set [<sup>88</sup>].

The reason for this approach is that, following the studies of [<sup>89</sup>], a supervised training mechanism for the controller must be excluded, thus meaning that no sensorial feedback can be used. More specifically, the knowledge of the error made in carrying out the movement can't be used to set-up the neural network. The exclusion of a feedback circuit both in the phases of learning and executing the task, reflects the motor control system capacity to explore the workspace without relying itself on pre-existent information (batch supervised training) or processing the data coming from the environment (feedback error learning).

In the learning phase of the network, the association: “starting point – neural inputs generating the movement from the starting point to an ending point” is therefore used. This is a step-by-step procedure in which the controller learns to make different movements.

Once again, it is important to outline that, unlike other models proposed in the literature, the controller learns the movement actually carried out, not the wanted one. Thus, during the training phase, the neural controller tends to achieve an optimal behaviour in reaching a desired target point by improving the correlation between the sensory map (starting and ending point) and the motor map (muscular activations which generate the movement between these two points) through the entire working plane.

The reduction of the error on the final position can thus be considered as a consequence and not a cause of the learning procedure. This is an innovative and unique characteristic of the proposed neural model (see figure 3.6).



**Figure 3.6** - Learning scheme of the proposed model.  
 The noise is added to the neural input generated by the controller. The new vector  $n_i$  is thus used for the generation of the muscular activities and for the controller training process.

## Simulating the Internal Model: the training phase

In every epoch of the training phase, the system automatically and randomly chooses the starting and ending points and receives, by the controller, the parameters to be used in the Pulse Generator. Great importance has the uniformity of the distribution and the normalization of the input signals that have to be sent to the ANN.

By allowing the neural network to explore only a subspace of the input data the possibility to achieve a good generalization decreases. In this case the weights would be not optimally balanced to process the data not observable in the training phase. On the contrary, with an uniformly distributed random explorative search it's possible to test all the possible associations between input and output data, without taking the risk that a part of the state space might remain unexplored; so the need to provide the controller with random inputs is related to a better characterization of task properties by means of an exploration strategy. Therefore the input data, in order to avoid problems related to the saturation of the transfer functions of the synapses, are normalized in the value  $[-1,1]$  or  $[0,1]$ .

In addition, during the exploring phase a random noise generator acting on the output of the neural network is used to prevent convergence on a state of local minima, which would imply a limitation in direction or amplitude of upper limb movements. The random noise generator is a module that can act on the output of the neural controller or directly to the neuron connections during the training phase. Noise has undoubtedly an important role in driving the exploration of the state space, which is necessary for learning to be carried out. The generation of the controller needs an identification of the controlled object that has to be driven. The fact that the biomechanical arm model shows a non-linear dynamics requires many different inputs to the neural network in order to characterize completely the system's response.

From a physiological viewpoint the noise generator can be related to the hypothesis of the presence of “noise sources” in the brain which behave like “stimulators” during the development of the motor system. In [10] different examples of this endogenous mechanism for the system identification are presented: 1) the development of patterns of neural innervations [90] consists of obtaining well defined correspondences between motor neurons and muscular fibres after a process, starting at the birth of any individual, where those correspondences are not fixed but work in a random way, 2) as an example of the organization of the brain it is possible to mention the association of several climbing fibres linked to the same Purkinje cell which evolves into an univocal association; 3) another interesting example of the presence of “noise generators” is the lack of myelination of the neuron present from the birth [91]. A lower sustain of myelin acts on the communication channels on neuron sometimes leading to a complete block of the transmission of neural signals and to imperfect motor control. The presence of all these noisy system seems to have a great importance in the development of the internal model.

The system has to be “excited enough” in order to attain a perfect identification of the inverse dynamic of the effector. The presence of the noise has to be connected to the dynamic progress of the controller: a complete overstatement or absence of it could bring respectively to a never learning state or to a limited learning of motor control.

From the developmental viewpoint it must be considered that each association initial-final position and the neural commands that carry out the corresponding specific movement is used to train the network only for one epoch; therefore it is possible to have, especially for the very first period of exploration, small variations in the neural controller weights.

This could possibly bring the neural network to converge to a state where the weights are not optimally calibrated to face the problem of the arm control in the entire working space. In order to simulate the reduction of this effect during time the noise generator has been implemented, so as to intervene on the output parameters  $p$  of the neural controller with a probability exponentially decreasing with the number of overall movements, according to the following equation:

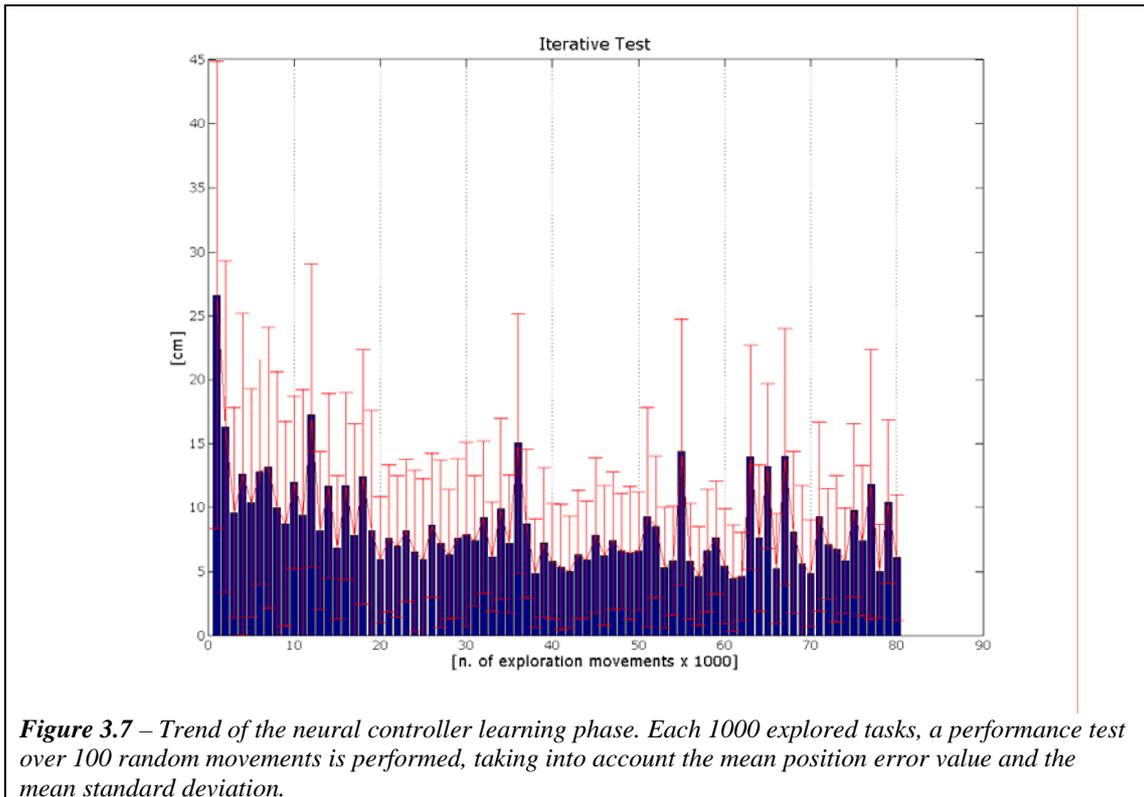
$$Pn(i) \propto e^{-\frac{i*100}{N_{max}}} \quad (14)$$

where  $N_{max}$  is the number of the total movements carried out during the exploration phase,  $i$  is the  $i^{th}$  movement and  $Pn$  is the probability of the noise intervention.

As initially the controller is not trained, there is no correspondence between the desired target and the one actually reached by the movement of the biomechanical model of the arm. At the end of each task, the training of the network (based on a standard back-propagation algorithm with momentum) begins. Only one epoch of training is used, in order to simulate a physiological behaviour in the building of the internal model. The training of the artificial neural network and a complete coverage of the working plane, with respect to both the starting and the target points, can be acquired between about 100.000 and 200.000 random generations (epochs). In the end, in this phase of pure exploration, the possibility to observe a statistically distributed association of starting and ending point and the related parameters of muscular activation, let the generalization capacity of the neural network define the final behaviour of the controller.

The training of the controller is based on an adaptive on-line paradigm; the supervised paradigm with respect to the desired output is avoided. Thus the development

of the neural network has to integrate the two actions of exploration of the state space, and exploitation of the current abilities at the given phase [10]. For this reason, during the learning phase, the system is tested on a variable number of motor tasks within the working plane, and the error of position is evaluated in terms of mean value and standard deviation.



## Hierarchical Neural Controller: a coarse to fine approach

In the introduction of this work the development process of the infants has been described. The exploration phase is necessary to obtain a initial internal model of the inverse dynamics of the upper limb with respect to the environment. It is well-known,

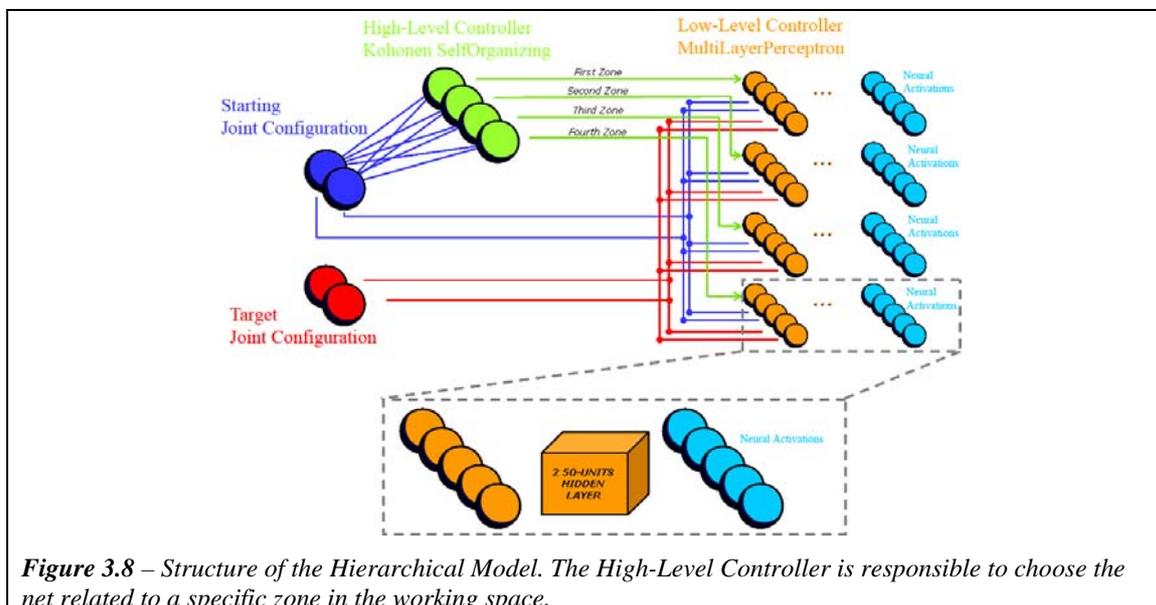
however, that for an adult subject the high complexity of the Central Nervous System architecture depends mostly on the fact that the brain is a highly structured entity with localized regions of neurons specialized in performing specific tasks. Each module at the macro-structural level has its own micro-structure of various cell types and connectivity [92]. A sub-division of complex tasks into simpler tasks is also evident in human and animal brains.

The observed modularity in brains is of two types. Structural modularity which is evident from sparse connections between strongly connected neuronal groups (with the trivial example of the two hemispheres of the brain) and/or functional modularity, which is indicated by the fact that neural modules have different neural response patterns, are grouped together. Along with the brain having a modular structure, it also exhibits a functional and structural hierarchy. Information in the brain is processed in a hierarchical fashion. First, the information is processed by a set of transducers which transform the information into the formats that each specialist modules can process. Specialist modules after processing the information, produce the information which is suitable for central or domain general processing. The hierarchical representation of the information is evident in the cortical visual areas where specialized modules perform individual tasks to accomplish highly complex visual tasks. For example, in the visual cortex of the macaque monkey, there are over 30 specialized areas with each having some 300 interconnections [93]. An increase in brain size does not necessarily increase the sophistication or behavioural diversity, unless accompanied by a corresponding increase in specialized brain modules [94]. The functioning of the brain can be summarized as the cohesive co-existence of functional segregation and functional integration with a specialized integration among and within functionally segregated areas mediated by a specific functional connectivity.

Decoupled modules architecture uses both unsupervised and supervised learning in two sequential stages [95]. In the first stage of the decoupled modules architecture, there is a decomposition of the input data into its inherent clusters in an unsupervised fashion. After classification of the input data into its inherent classes, each class is assigned to an individual module for learning. These modules are then trained in parallel using a supervised learning paradigm; and, there is no communication between modules

during the training. The final classification is obtained using the absolute maximum of the activation of all the individual modules.

On the basis of a decoupled modules architecture, a hierarchical structures comprising a Self Organizing Map (for a deeper analysis of the Self Organizing Maps see Appendix A) and 4 normal neural controller, as the one exposed in the Chapter 2, has been designed. The idea is based on a main net which is responsible to divide the input signals into different zone or clusters, thus choosing a proper neural controller associated to that specific zone. Figure 3.8 can clearly represents the general scheme.



The system is based on an autonomous learning process which can be synthetic as follows:

- A Kohonen Net is created together with a MultiLayerPerceptron.
- During the training phase of the MLP, performed by the exploration phase as described in the previous paragraph, input signals are randomly presented to the high-level controller.

- If the variation of the weights connections of the low-level controller between to subsequent time steps shows a value lower than a prefixed threshold then:
  - Both the training of the Kohonen Net and of the MLP is stopped.
  - Three copies of the original MLP are generated and are connected to the high-level controller.
  - The coarse-to-fine training starts.

## **Simulating the Internal Model: Testing the performance of the model**

The study of planar ballistic movements has been extensively undertaken in literature, because it provides important insights into the functional organization of the nervous system [96]; the attention has been focused both on able bodied subjects [48] [79] [97] [98] [46] and on young infants [99] [42].

Almost all of the studies agree about some kinematic invariance in reaching planar movements: the path of the movement is essentially straight [48] [79] [47], the velocity profiles are single-peaked and bell shaped [48], and the peak acceleration and peak velocity scale systematically with movement amplitude [79] [98]. Moreover, a great variety of computational models has been designed aiming to describe the motor control [100][101]; anyway most of them incorporate a supervised learning algorithm while in others the idea of a pre-planned complete trajectory is present.

In order to validate the correct implementation and functioning of the neural controller learning paradigm from a physiological standpoint, several tests have been performed; the purpose of these investigations has been directed to compare the results obtained from the data acquired on human tasks with the ones emerging from the present model. What is generally studied in reaching arm movements are the position error with

respect to the final target, the velocity end-point profile [48], the acceleration and the correlation between the peak velocity and the length of the movement [98].

The neural controller has been tested by presenting a high number of pairs of randomly chosen start-target points, and the errors in the reaching of the target have been recorded: the movements have been divided in group of 200 of almost the same length (i.e. 5 cm tolerance) and the whole set covers an amplitude from 5cm. to 60cm. In this way it has been possible to observe the behaviour of the neural controller for 11 different spaced intervals.

The observation window of the motor tasks has been fixed as 1.5s long.

Three main geometrical errors have been taken into account:

- The absolute position error of the arrival position reached by the biomechanical arm model with respect to the desired final position (or target)
- The module error
- The phase error

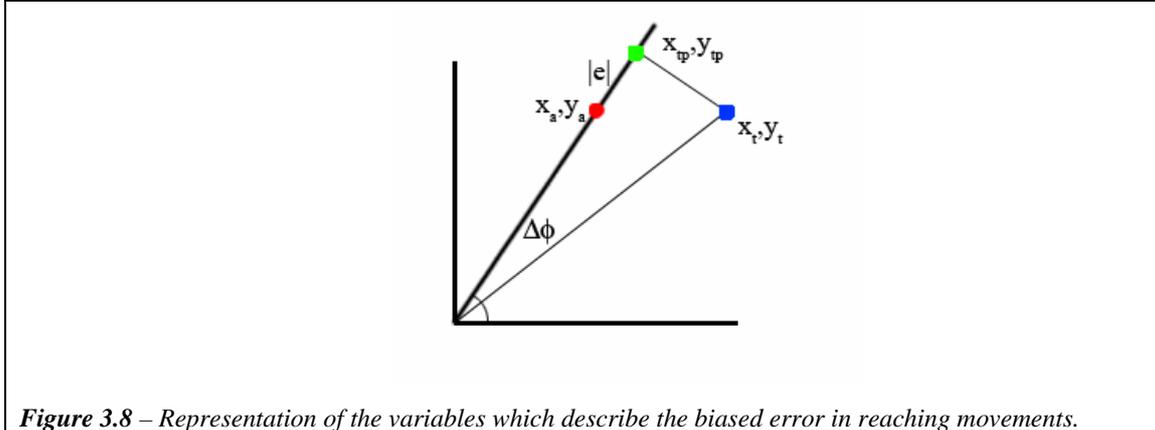
The last two have been chosen in order to reveal the presence of a biased behaviour:

$$\begin{bmatrix} m_a & -1 \\ -\frac{1}{m_a} & -1 \end{bmatrix} \begin{bmatrix} x_{tp} \\ y_{tp} \end{bmatrix} = \begin{bmatrix} 0 \\ y_t - \frac{1}{m_s} x_t \end{bmatrix} \quad (15)$$

$$\Delta\varphi = \varphi_a - \varphi_t \quad (16)$$

- the module error  $|e|$  is defined as the euclidean distance between the arrival point  $(x_a, y_a)$  and the projection of the target  $(x_{tp}, y_{tp})$  (both in Cartesian coordinates) on the line passing through the starting point and the arrival point (see equation 15).

- the phase error  $\angle e$  ( $\Delta\varphi$ ) is defined as the difference of the angles which identify the two lines connecting the starting point with respectively the target and the arrival point (see equation 16).



**Figure 3.8** – Representation of the variables which describe the biased error in reaching movements.

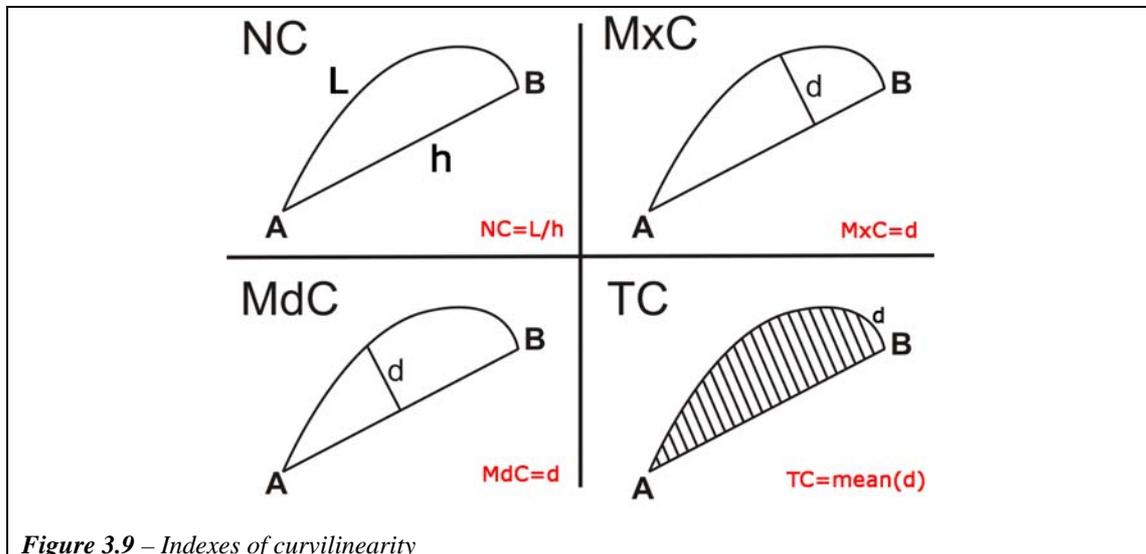
Moreover the characteristics of the tasks performed have been analyzed and compared to the data obtained from experimental tests on human beings. The index used was the index of curvilinearity, which can give a quantitative index of how straight is a movement. In literature there are 4 main ways in which this index is evaluated.

In [46] it is defined as the ratio between the length of the movement performed and the minimum distance between the starting and the arrival point (see equation 17).

$$NC = \frac{\sum_{i=1}^{N-1} \sqrt{dx_i^2 + dy_i^2}}{\sqrt{(x_f - x_s)^2 + (y_f - y_s)^2}} \quad (17)$$

In [102] two indexes are used: the ratio between the distance from the medium point of the straight line connecting the start (A) and the arrival point (B) and the trajectory performed by the subject (*MdC*), and a similar value considering the maximum point of the straight line (*MxC*). In [79] this index is defined as the medium value of all the distances from the points defining the trajectory and the line defining the minimum

distance from the two extremities of the path ( $TC$ ). Figure 3.9 graphically describes these differences.



## Simulating the Internal Model: Testing the performance of the model facing external forces

A fundamental feature in the study of the motor control of the upper limb and of the generation of the internal models is certainly the aim to understand in which way the central nervous system adapts to variations of the dynamics interactions with the environment. There have been several researches aimed to the investigation of this high synaptic plasticity, by means of robotic systems able to produce controlled “dynamic environment” [24] [103].

The most interesting results showed that while under normal conditions the inverse dynamic model calculates motor commands which compensate the arm dynamics, under altered conditions these motor commands are insufficient, and this leads to distorted movements and great errors in point to point tasks.

Anyway the repetition of the actions inside the “modified” environment leans toward a modification of the internal model of the upper limb. The continuous cycle perception-action rapidly adapts the central nervous system so that the new motor commands comprise the overall effects both of the arm dynamics and of the external forces. A number of works [105][104] are focused on the study of the way the central nervous system faces these instabilities of the environments; the method is based on the variation of the joint impedance by means of a change of the level of cocontraction of the antagonist muscles.

In order to test the biological plausibility of the model implemented in the current work, the neural controller trained in a normal environment has been tested applying a simulated external force proportional to the tangential velocity of the end effector.

The results have been examined from a kinematics viewpoint (trajectory, velocity profile) and a special examination has been performed on the stiffness variation of the limb while facing the unstable interactions.

# **Chapter 4**

## ***Neural Controller in Normal and Distorted Environment: Results and Observations***

## Neural Controller in a Normal Environment

A first interesting feature of the proposed neural system is its capacity to achieve a complete coverage of the working plane, unlike other models [69] which are limited to short amplitude motor tasks, usually around 10-20cm.

This characteristic can be appreciated in figure 4.1 where, for visualization purposes, the same starting point and 1000 target points have been considered.

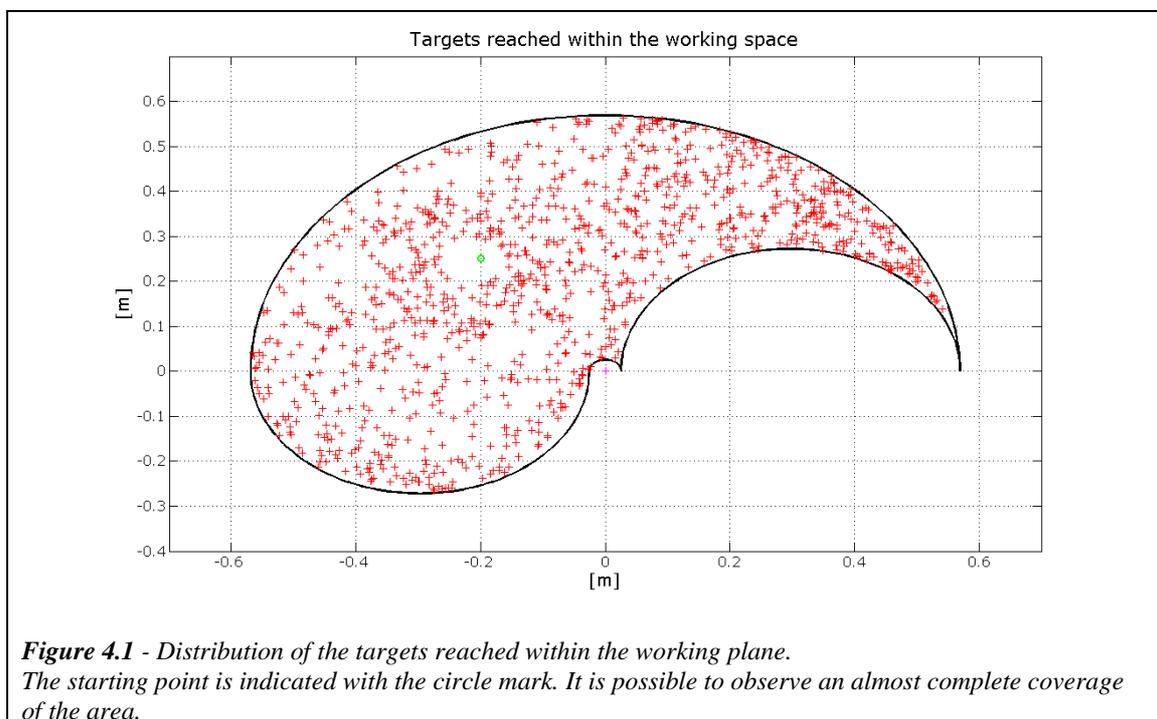
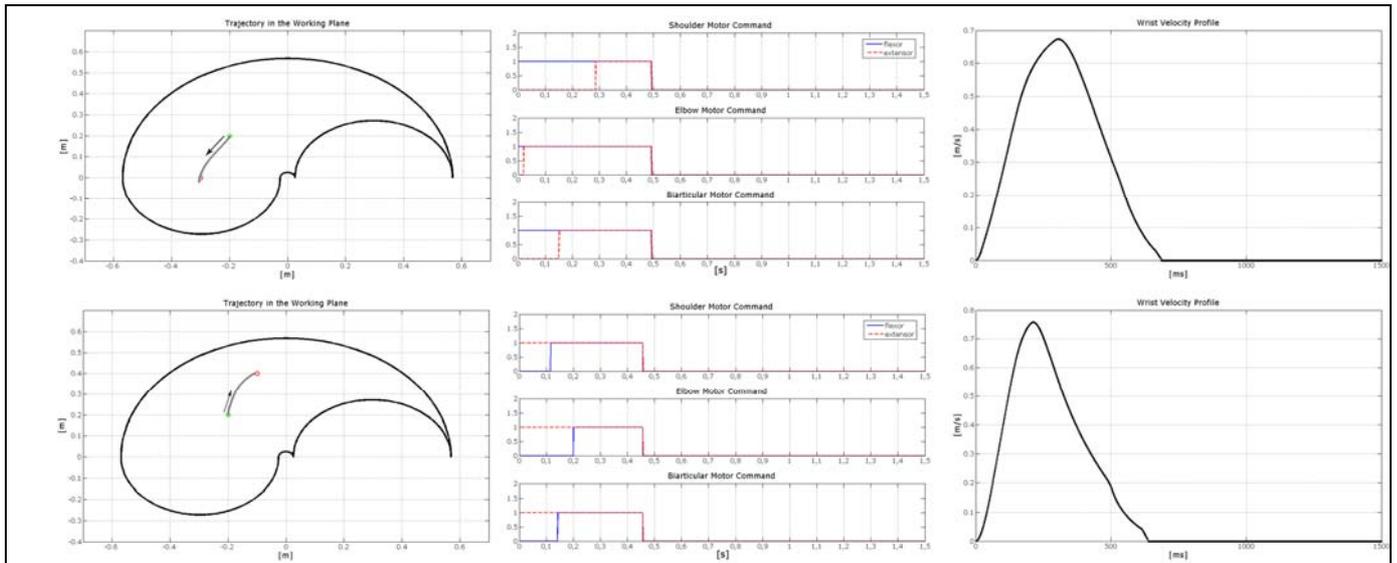


Figure 4.2 shows two different movements starting from the same point, together with the neural outputs  $p$  and the relevant velocity profiles.

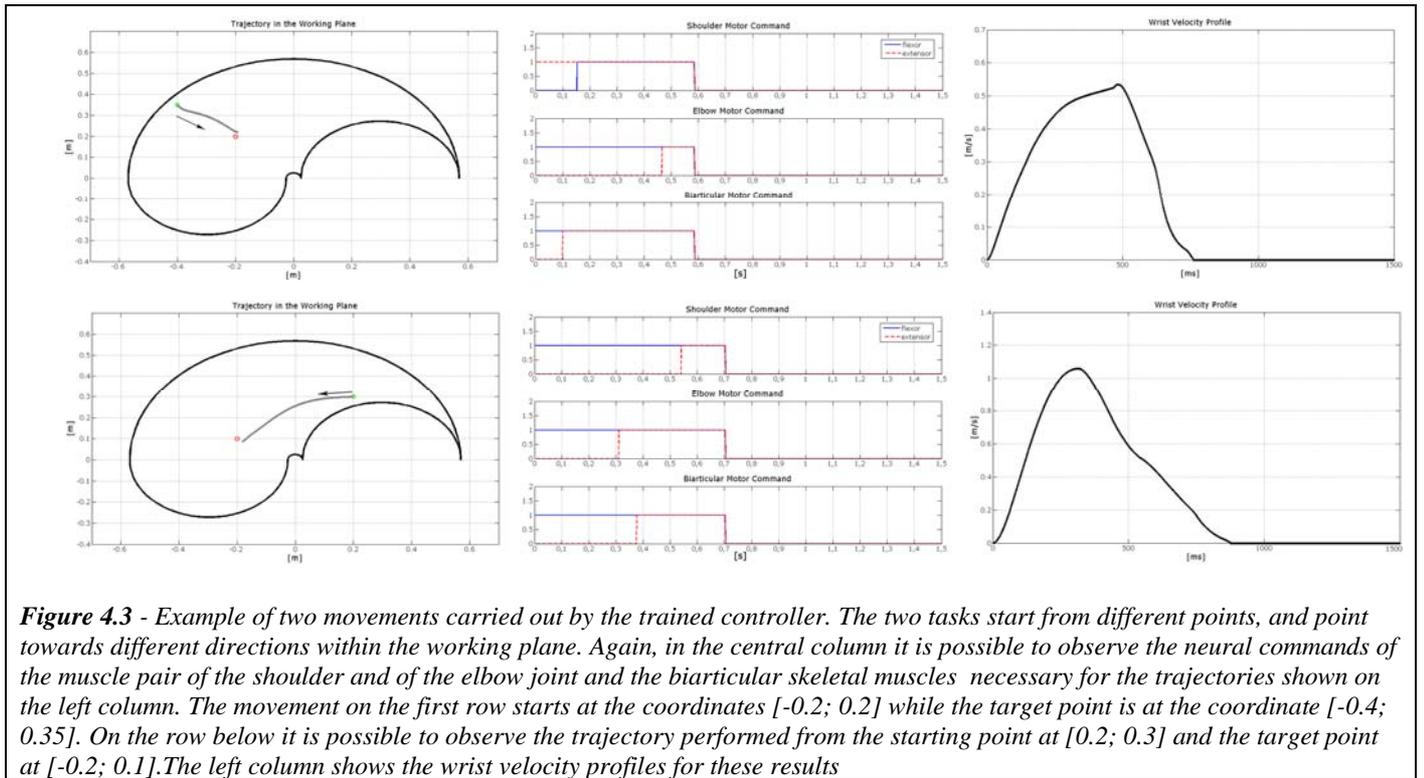


**Figure 4.2** - Example of two different tasks carried out by the arm model guided by the trained neural controller. The starting point is the same for the 2 tasks (coordinates:  $x = -0.2$ ;  $y = 0.2$ ); the arrival points have been chosen in 2 different position symmetric with respect to the starting point, at the ideal distance of 22.5cm. Each row represents a different movement. The left column of this image depicts the trajectory followed by the wrist. The central column shows the neural inputs of the flexor and extensor muscles acting on the shoulder and the elbow joint, and the muscle couple acting as biarticular. The right column shows the wrist velocity profile. In both the figure the starting point is identified by the green point while the target is identified by the red point. Moreover the neural commands governing the flexor muscle are depicted with a solid blue line while the ones related to the extensor muscle with a dotted red line.

The upper movement highlights the role of the Pectoralis Major, in the shoulder joint, and the Biceps, in the elbow joint, for targets located in a position to the west with respect to the starting point, while the second one implies the use of the Deltoid and the Triceps for the target in a position to the east with respect to the starting point.

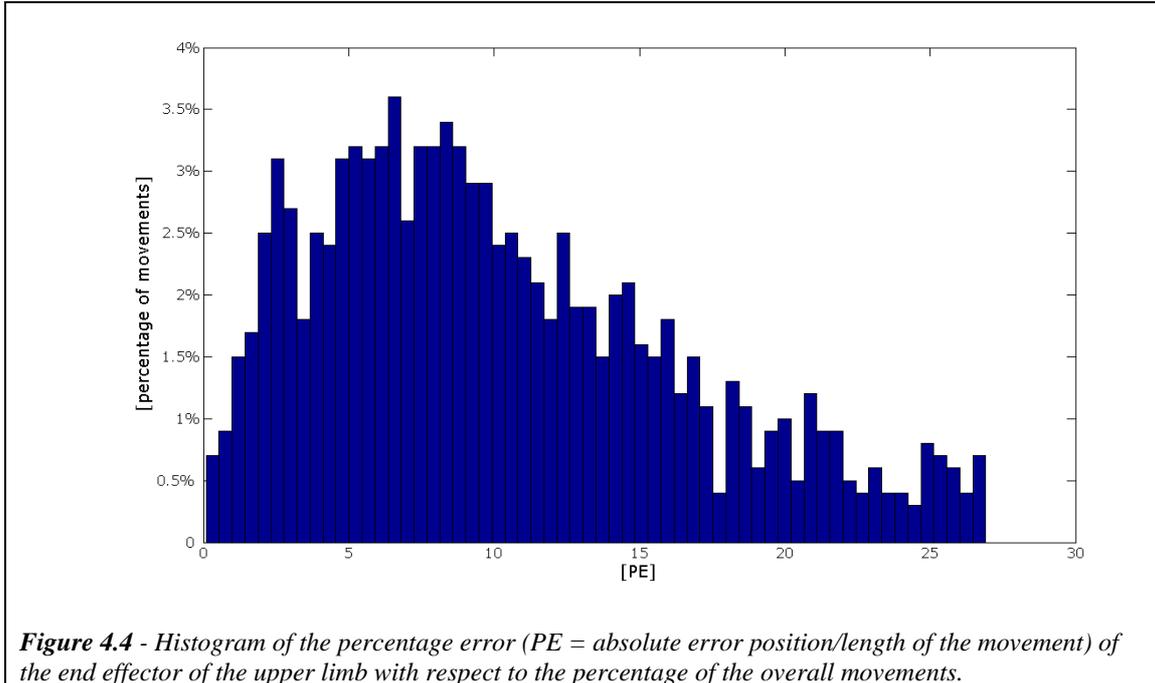
The graphs of the wrist velocity reflects the results present in the literature (see for instance [48][98]): a bell shaped and non symmetrical profile related to a smooth movement.

Figure 4.3 shows the interesting result that even when changing the starting point, the relations between the direction of the movement and the neural inputs persist.



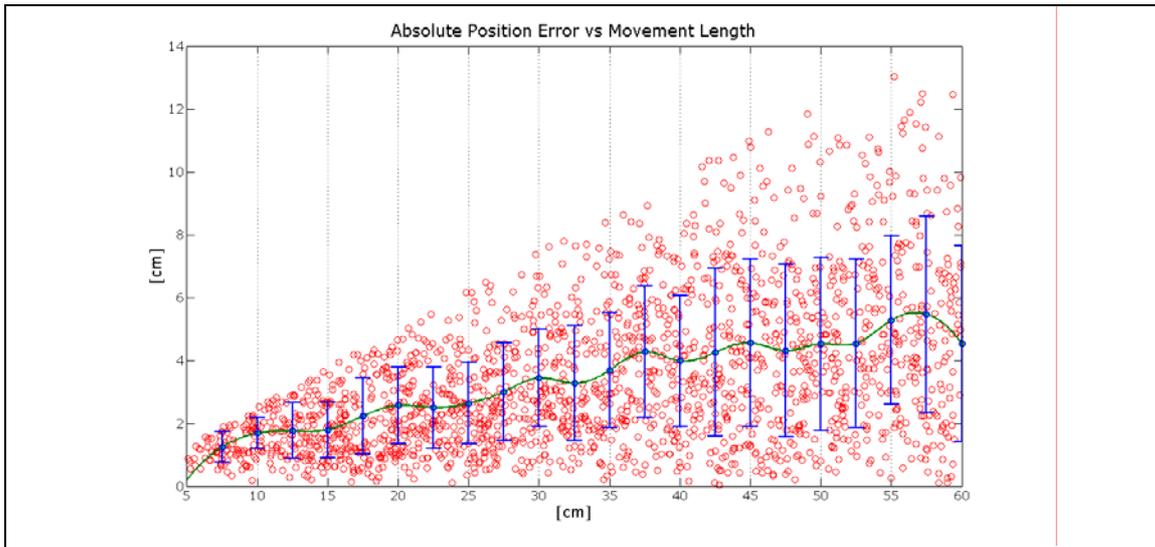
A set of 200 movements ranging from 5cm. to 60cm. have been divided for length into groups space out by 5cm. Of these 1200 movements whose starting and target points are equally distributed within the working space the mean position error reached through these movements has been of about **4.82cm** with a standard deviation of about **4.05cm**. This result is similar to the ones provided in [48], where movements with a maximum amplitude of 60cm have been analyzed.

Figure 4.4 shows the histogram of the percentage of the absolute position error with respect to the length of the movement. For the movements within  $\pm 30$ cm, the mean absolute error, normalised with respect to the length of the movements, resulted to be within **27%**. These findings show that the model is able to accurately simulate ballistic (unobstructed) movements of the arm.



Interesting results emerge also from the graphic of the distribution of the absolute position error and the standard deviation with respect to the increasing length of the movements analyzed (Figure 4.5). The values are evaluated within intervals of 5cm. starting from a value of amplitude of 5cm. up to the maximum amplitude of 60cm.

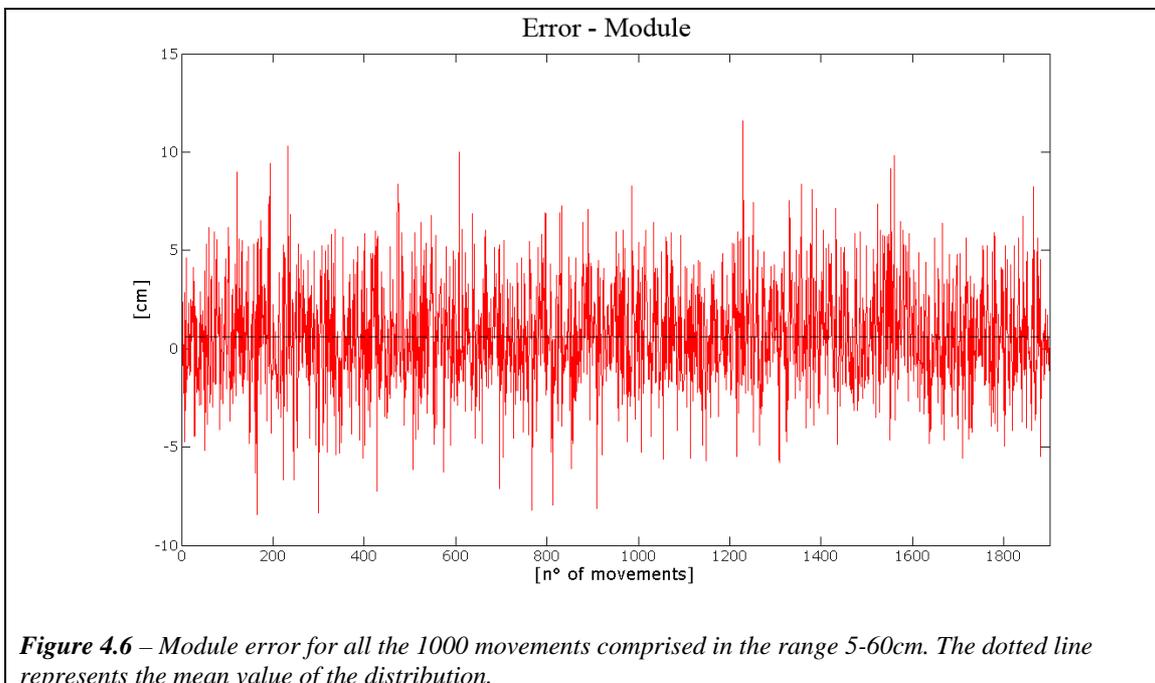
The black line is generated by means of a cubic spline interpolation (command `spline.m` from MATLABR2006©), while the height of the blue lines represents the value  $2*STD$  (where  $STD$  is the standard deviation). It is possible to see that the mean absolute position error has a limited variation with the increase of the movement length.



**Figure 4.5** – Distribution of the mean absolute error values and mean standard deviations with respect to the amplitude of the movements.

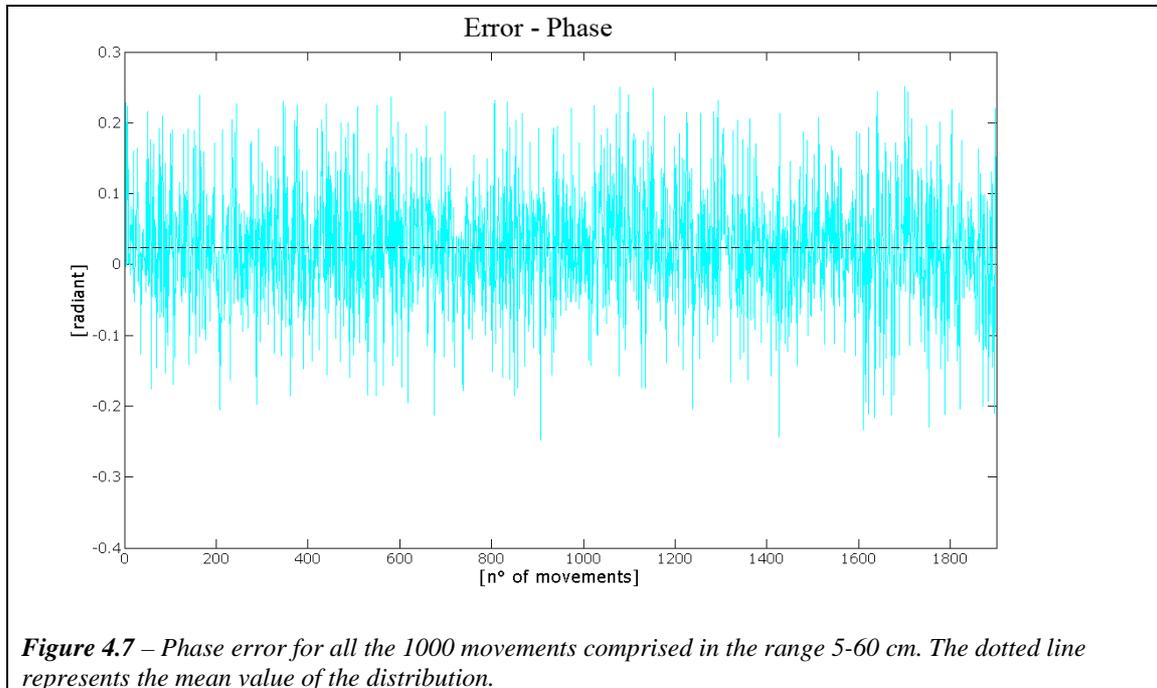
If we consider the module error of the final position of these movements it is possible to observe a value of about **0.52cm**.

This results represents the movement amplitude variability. Figure 4.6 shows better this result.



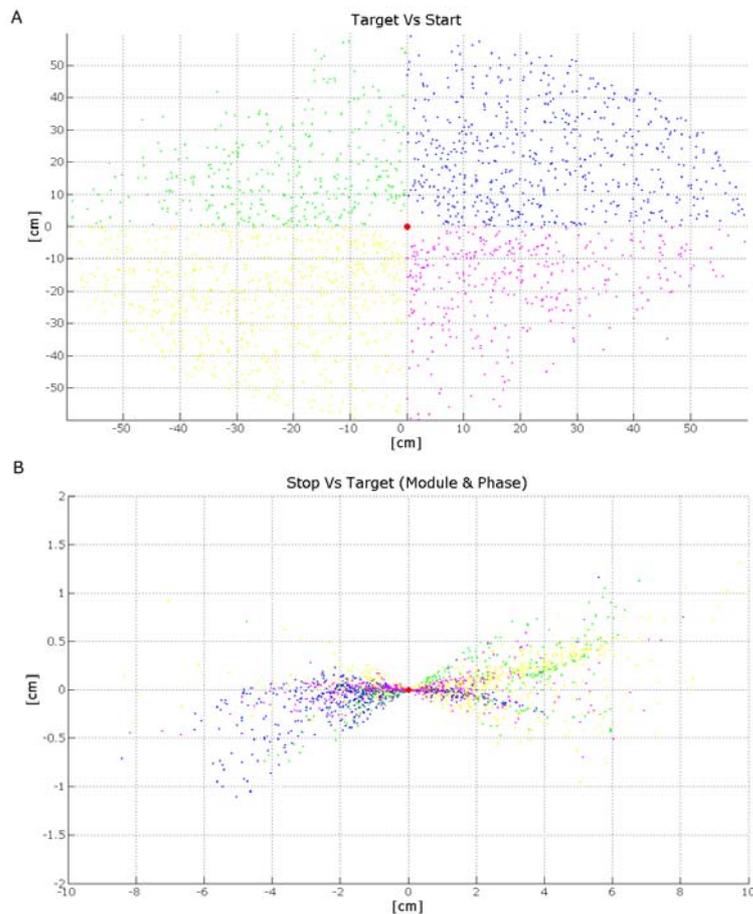
**Figure 4.6** – Module error for all the 1000 movements comprised in the range 5-60cm. The dotted line represents the mean value of the distribution.

The mean value of the phase error results almost equal to **0.02rad**, thus showing that the artificial neural controller gives almost unbiased results with respect to the direction variability; it is able to correctly point (in average) towards the target (figure 4.7).



**Figure 4.7** – Phase error for all the 1000 movements comprised in the range 5-60 cm. The dotted line represents the mean value of the distribution.

It is interesting to underline that this result is consistent with the data observed in [96]: the difference between the variability of the amplitude of movement toward a given target is typically greater than the variability of direction. This is extremely clear from figure 4.8, which shows the distribution of the arrival points with respect to the desired targets, assuming that the direction of the movement is specified by the line parallel to the x axis passing through the target point.



**Figure 4.8** – The figure A is the distribution of the target point with respect to the start. All the different start points have been centred to the same position. Subsequently the target points have been discriminated with respect to four quadrants. In the figure B the discriminated points have been positioned along the same direction to observe the amplitude variability with respect to the direction variability for each quadrant of movement.

From the figure 4.8B it is possible to observe that the greater amplitude variability values (module error) are related to the movements within the upper right quadrant (blue points) and the lower left quadrant (yellow points), while the greater direction variability error (phase error) are associated to both of the upper quadrants (blue point and green points).

If the outliers (which are the movements that show a ratio between final error position and length of the desired task greater than 25%) are not considered, the results obtained

including only one starting point and movements with a maximum amplitude of 30cm., show a mean error position value of about **2.38cm** with a standard deviation of **1.78cm**.

This value is absolutely equivalent to the results obtained by the kinematics analysis on human subjects. In [105], the constant error (defined as the mean distance between the finger tip at the movement end and each target location), for target with a radius 45mm. at 15cm. of distance from the starting point is about 13mm. Still, it is important to highlight that the estimation of the absolute position error in the current model is intended with respect to an adimensional point.

A comparison between the experimental data reported in [46] [106] and the data extracted from the simulated model of the present work is interesting because it puts in evidence the behaviour of the proposed neural model for what concerning the curvilinearity.

To unify the study of this parameter with the analysis extrapolated by the literature, the four values of curvilinearity have been taken into account. The table 4.1 shows the mean values of the normal curvilinearity (*NC*), of the maximum curvilinearity (*MxC*), of the medium curvilinearity (*MdC*) and of the total curvilinearity (*TC*). The value of *NC* reported in [46] is about 1.02, for movements with a maximum amplitude of 42cm., while in this system the mean value is **1.09**. If this value is calculated considering 42cm as maximum length it decreases to **1.06**.

Two main things must be outlined:

- The biomechanical arm model is still far from a real upper limb structure, in which further muscle activations, even if with a low influence, act on the overall movement; nevertheless the results are very interesting.
- All the experiments on human subject from the literature are replications of the same set of movements in different direction or

with different amplitude; this brings a specialization of the tasks during the trials.

Normal Curvilinearity <b>NC</b>	1.09
Maximum Curvilinearity <b>MxC</b>	0.63 cm
Medium Curvilinearity <b>MdC</b>	0.61 cm
Total Curvilinearity <b>TC</b>	0.16 cm

**Table 4.1** – mean values of the curvilinearity indexes for the set of movements analyzed

In [106] the normalized maximum curvilinearity shows a value of  $0.0525 \pm 0.0224$ . This result has been estimated as the ratio between the maximum distance from the straight line connecting the starting and the arrival point (that is the value MxC of the present system) and the length of the straight line connecting the two of them; moreover the values reported are related to tasks performed on the sagittal plane. As verified by [104], the reaching movements on the transversal plane show a lower mean value, that is about 0.25cm. The results obtained by the neural controller implemented in this work are very similar, showing a normalized MxC of about **0.29cm**.

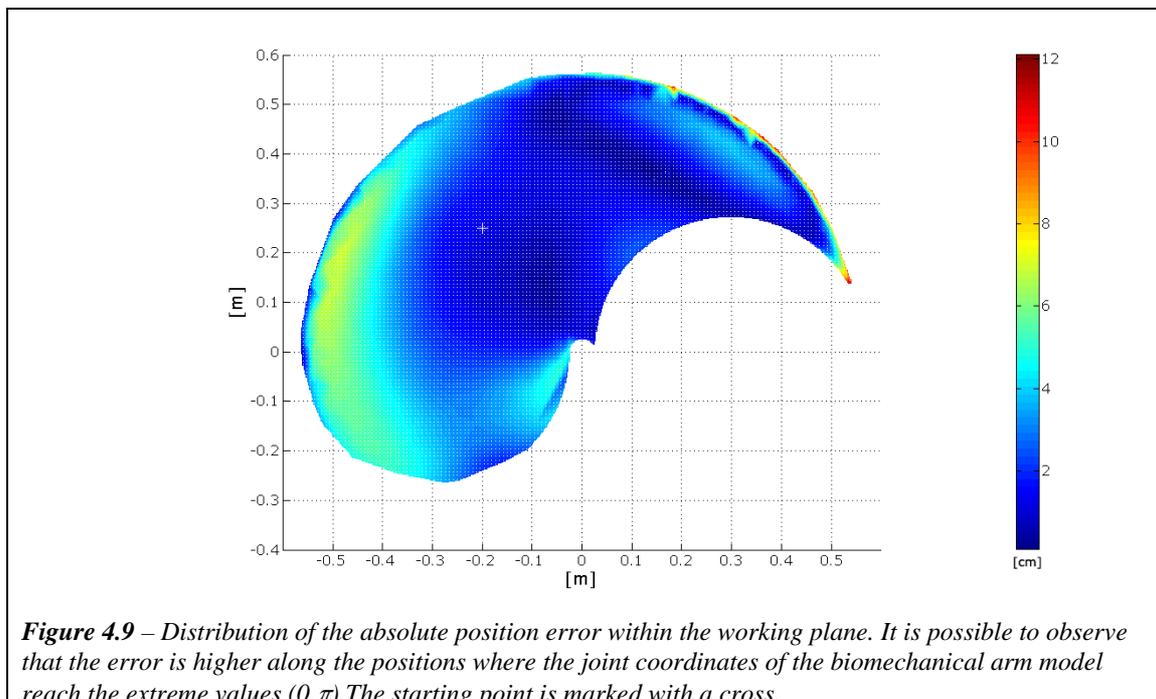


Figure 4.9 depicts a bidimensional projection of the wrist final position error after testing 1000 movements from the same starting point. It is possible to notice that the behaviour is reasonably uniform, even if there are some error peaks far from the starting point along the borderline of the working space.

Figure 4.10 shows the behaviour of the velocity profile whose peak, considering the movements starting from the same point, increases accordingly with the length of the movements. From the model it has been possible to evaluate the presence of the “scaling effect” which explains the invariant property of the wrist velocity profile: when the length of the movement increases, so does the maximum velocity reached along the trajectory while maintaining the same profile.

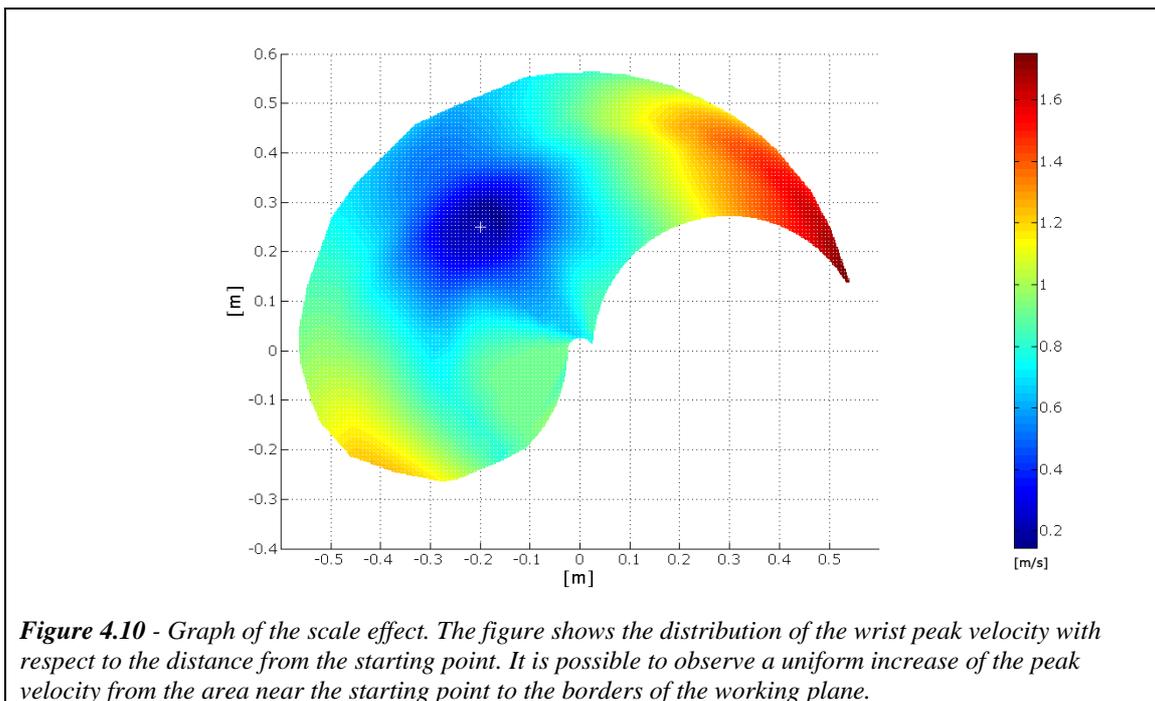
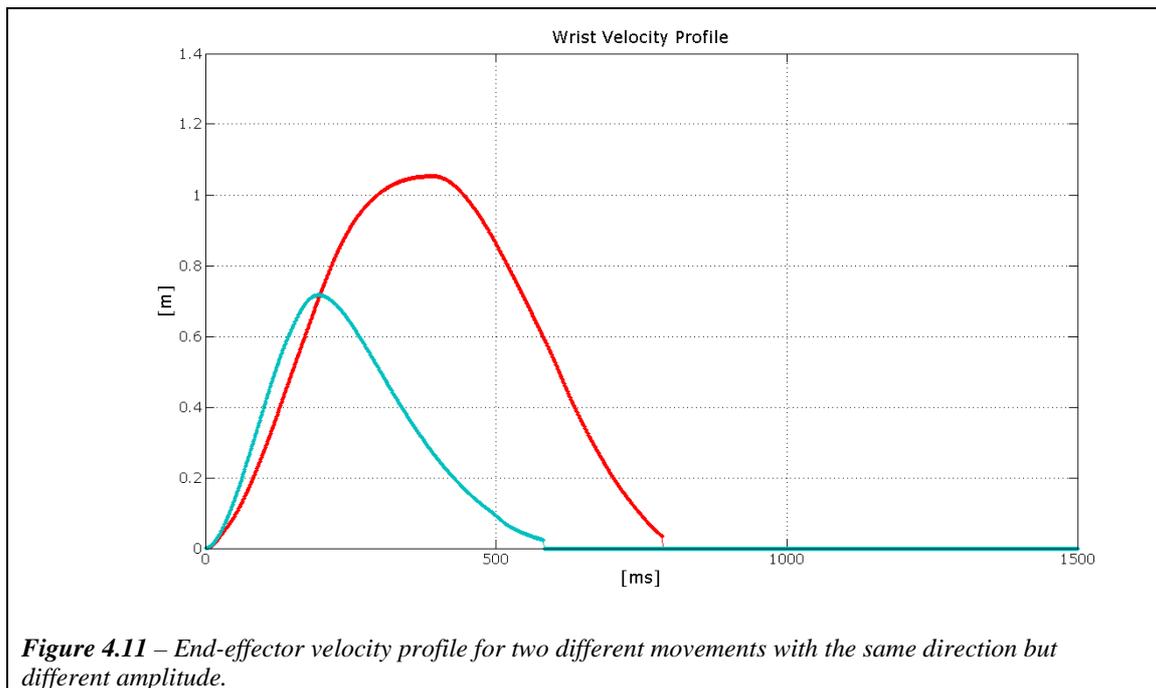


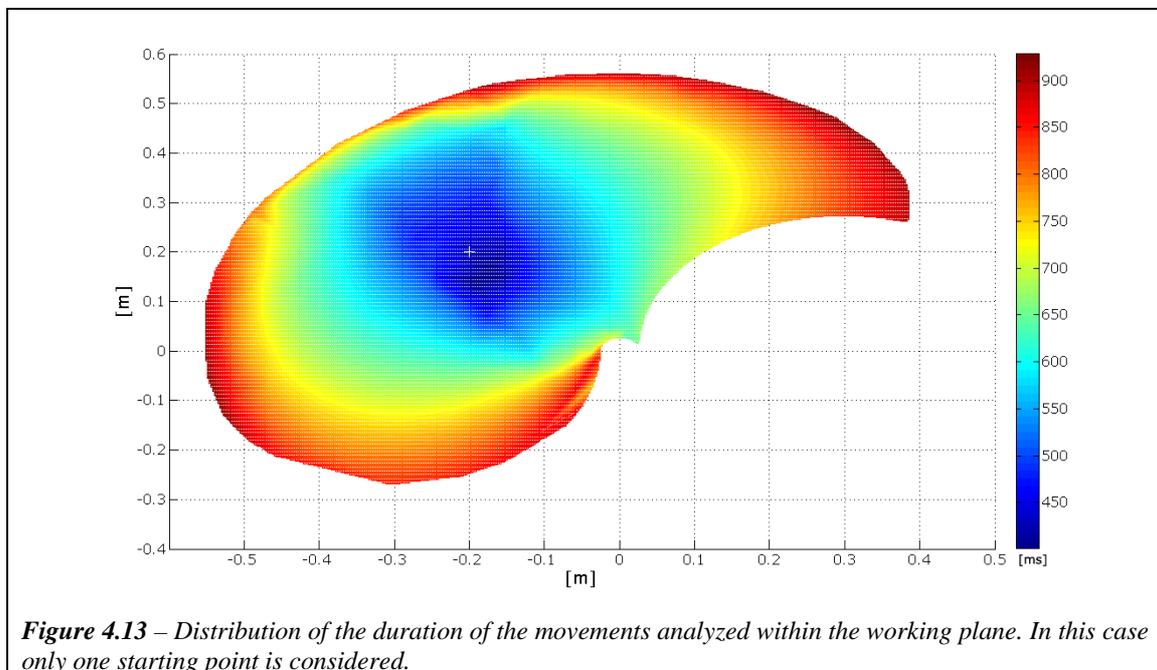
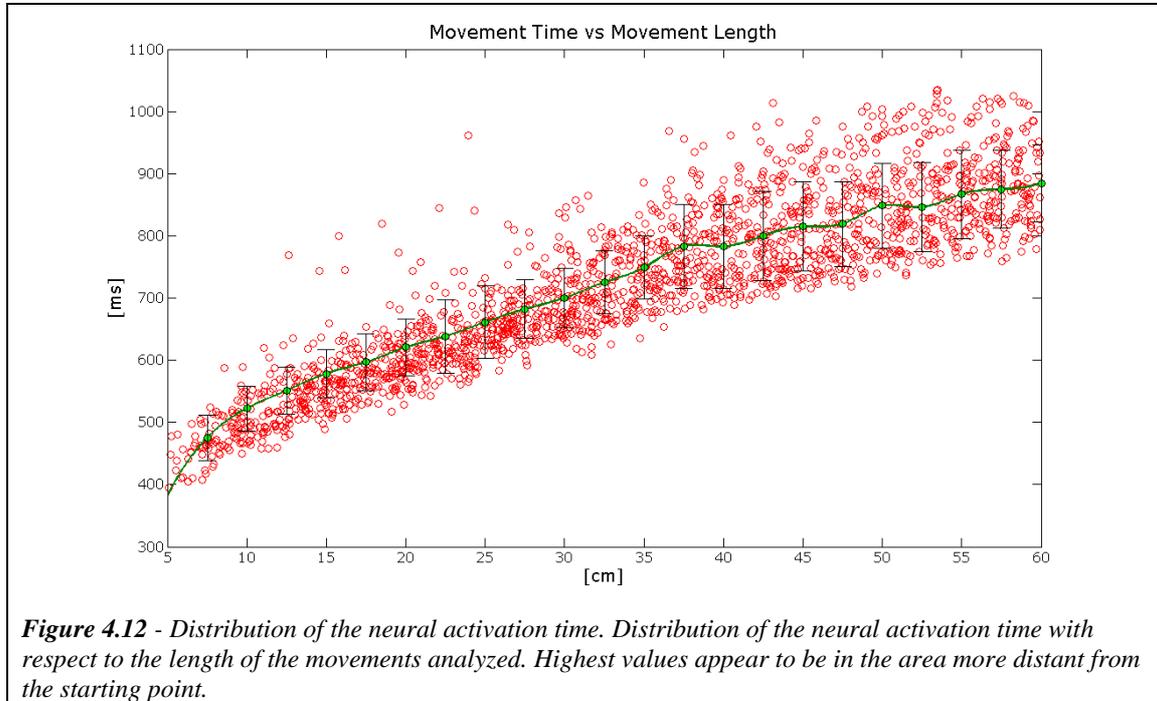
Figure 4.11 also shows that the velocity profile drawn for two movements, with different lengths, is approximately the same. It is interesting to observe that considering 2 movements having the same reaching direction but different amplitudes, the shorter task usually shows a peak velocity lower than the longer task, and at the same time, a higher value of the acceleration. The mean peak velocity of the entire set of the movements is about **0.8 m/s**.



**Figure 4.11** – End-effector velocity profile for two different movements with the same direction but different amplitude.

As far as the duration of these movements is concerned, the neural controller leans towards a biological behaviour showing that the duration of the movements does not vary linearly with the length (see figure 4.12). These findings are similar to those present in literature [103], [98]. This means that similar activations bursts are associated to similar movements: i.e. it is possible to see that for the movements directed towards the same area inside the working plane, not only the same muscles of the shoulder and the elbow joint are activated first, but also the intervals of the neural activations of these muscles show the same duration. This finding can be correlated with a feature that could be defined as a global isochrony of the movements.

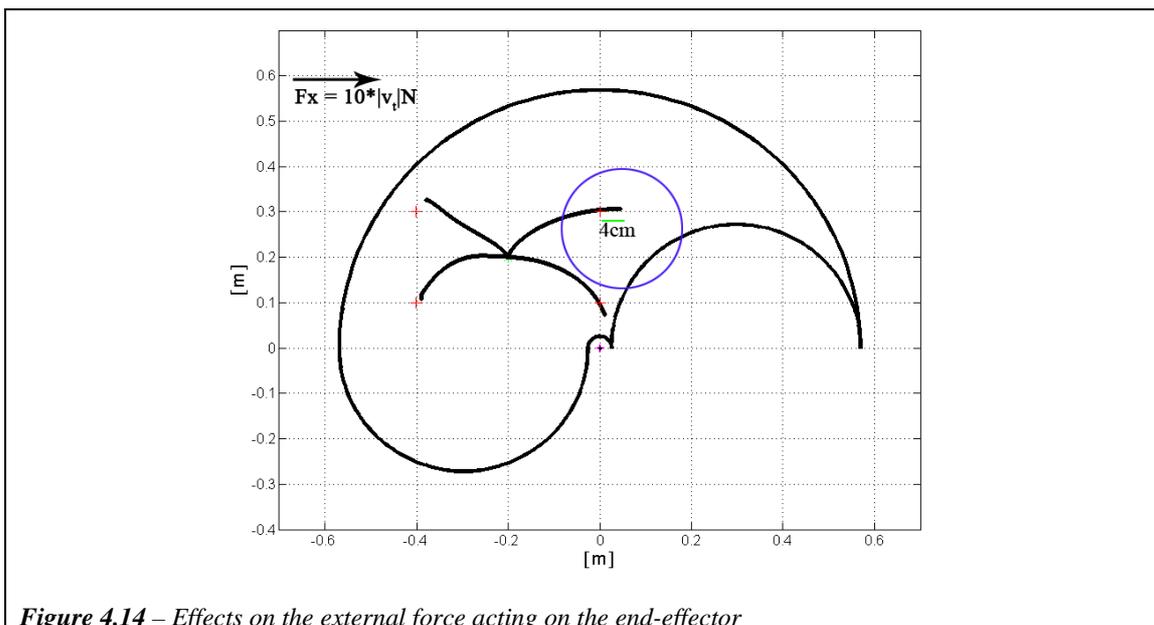
Figure 4.13 depicts the bidimensional projection on the working plane of the values of duration of the movements.



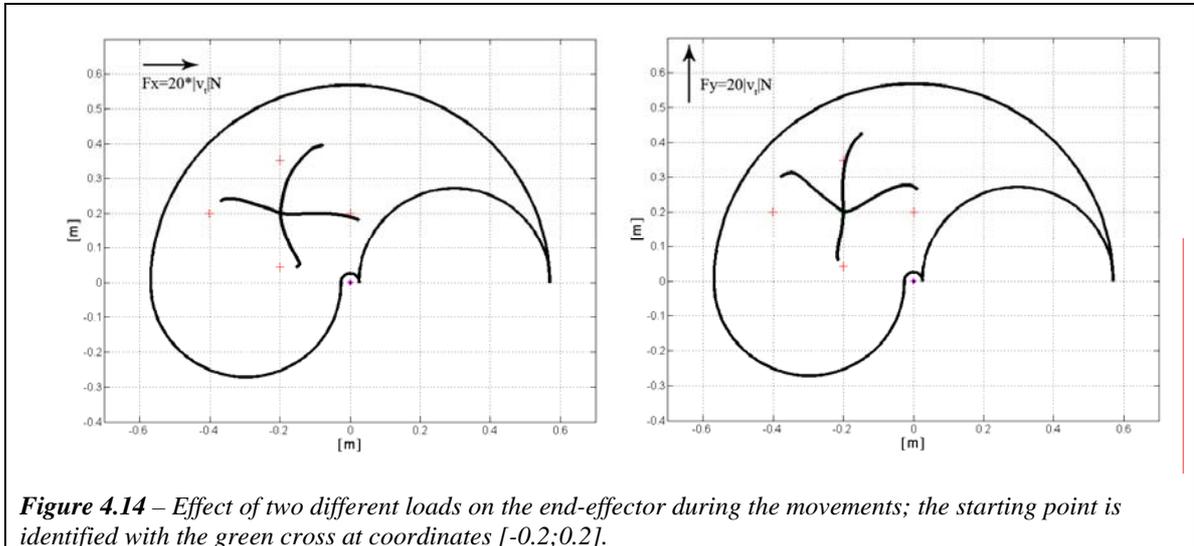
## Adaptation of the Neural Controller to Force Fields: Results and Observations

The learning paradigm has been tested by applying a force field, in order to highlight the adaptability of the neural controller while facing instability of the environment. The force acting on the end-point of the biomechanical arm model has been assumed proportional to the modulus of the tangential velocity of the end-effector, and directed respectively towards the x axis and y axis.

In figure 4.14 it is possible to observe the effect of the force A)  $F_x = 10[Nm/s] * |v_t|$  on the neural controller trained in a environment with no external loads. The starting point is at the coordinates  $[-0.2; 0.2]$ , and four target symmetrically located at the distance of about 23cm from the central point have been selected.



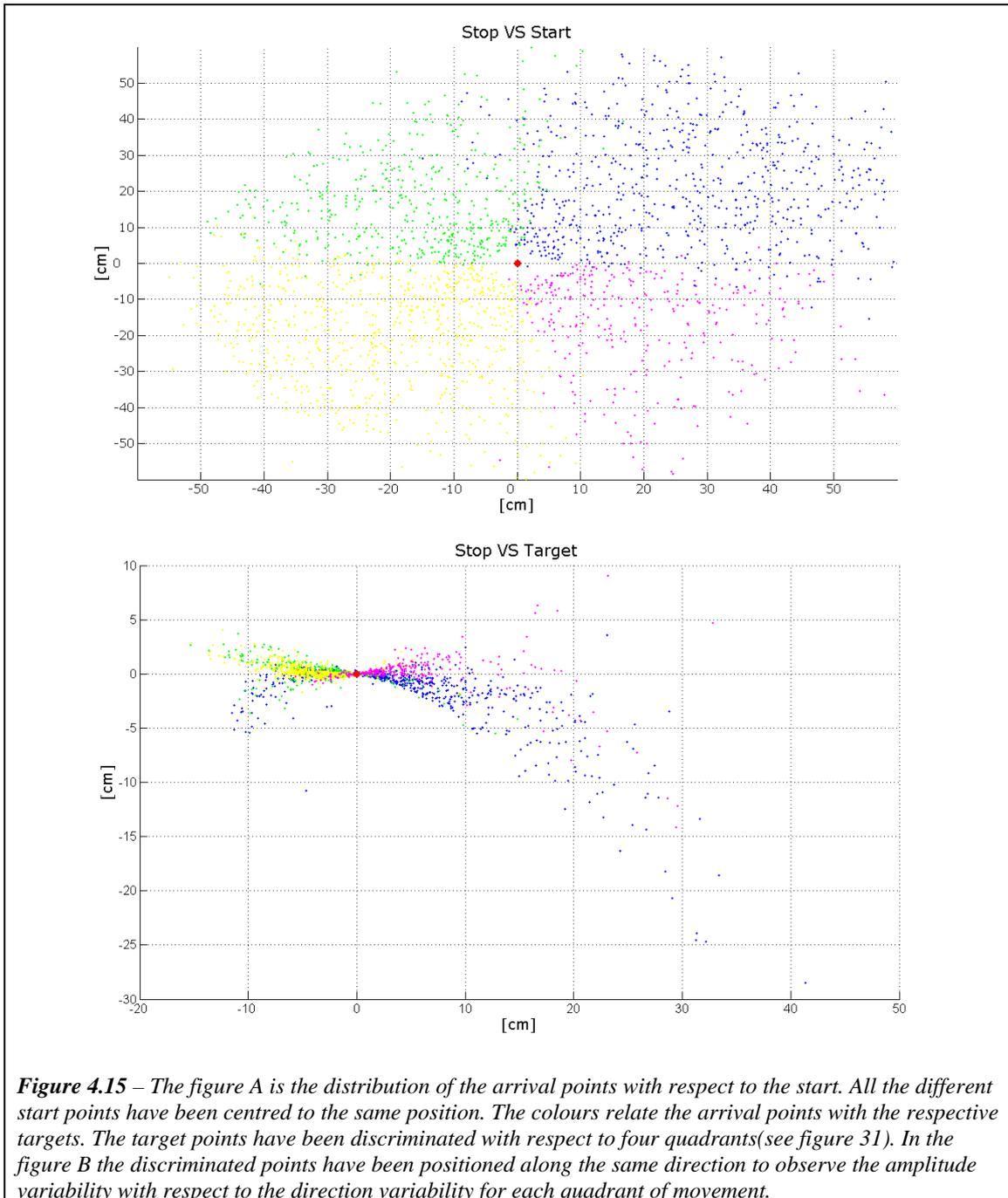
The effect is obviously clearer imposing a constant value of 20, respectively toward the x axis and y axis (see figure 4.15).



It is interesting to notice that the force effect is greater when its direction is perpendicular to the stiffness ellipse whose orientation is established by the virtual segment connecting the shoulder with the hand <sup>[107]</sup>.

This feature is evident if we consider the graph of the arrival points distribution with respect to the target (see figure 4.8 for an explanation of the distribution of the points).

Again, a set of 200 movements ranging from 5cm. to 60cm. have been divided for length into groups space out by 5cm. under the presence of a force directed along the x axis.

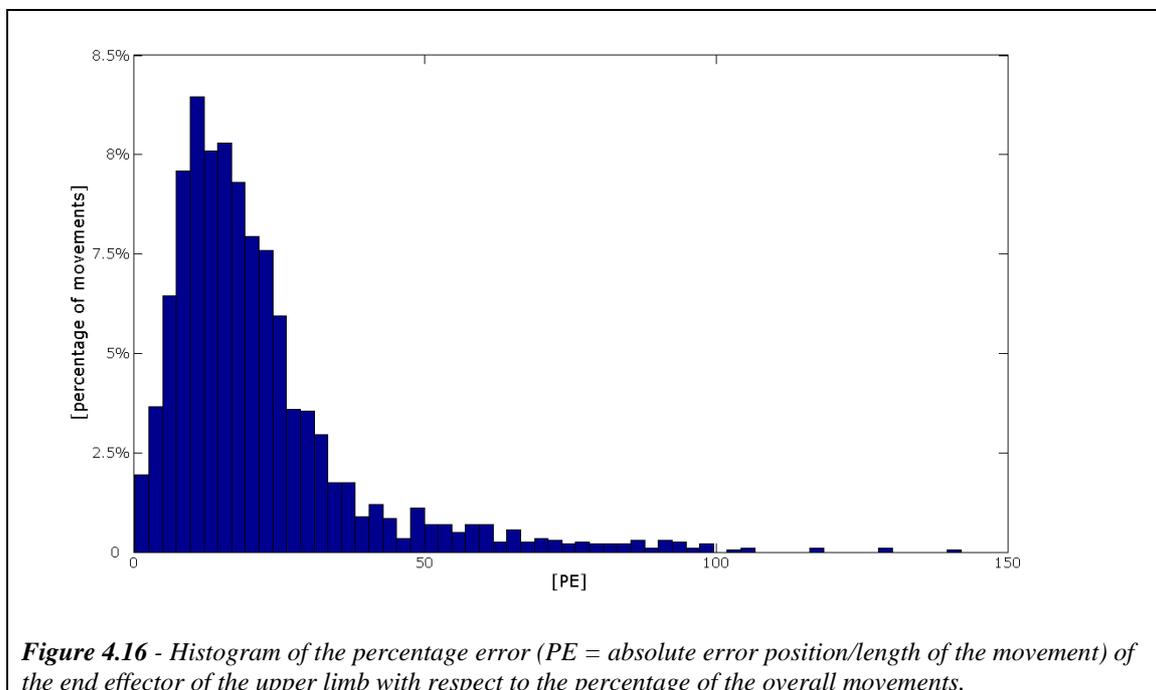


**Figure 4.15** – The figure A is the distribution of the arrival points with respect to the start. All the different start points have been centred to the same position. The colours relate the arrival points with the respective targets. The target points have been discriminated with respect to four quadrants(see figure 31). In the figure B the discriminated points have been positioned along the same direction to observe the amplitude variability with respect to the direction variability for each quadrant of movement.

The main effect of the force field is to translate the amplitude variability error along the x axis (see figure 4.15A); furthermore it shows an even greater effect on the direction variability error (see fig. 4.15B). Also the effect of these forces on the deviation of the trajectories [<sup>108</sup>][<sup>109</sup>] is apparent.

The central nervous system has to face with alteration in the dynamic interaction with the surrounding.

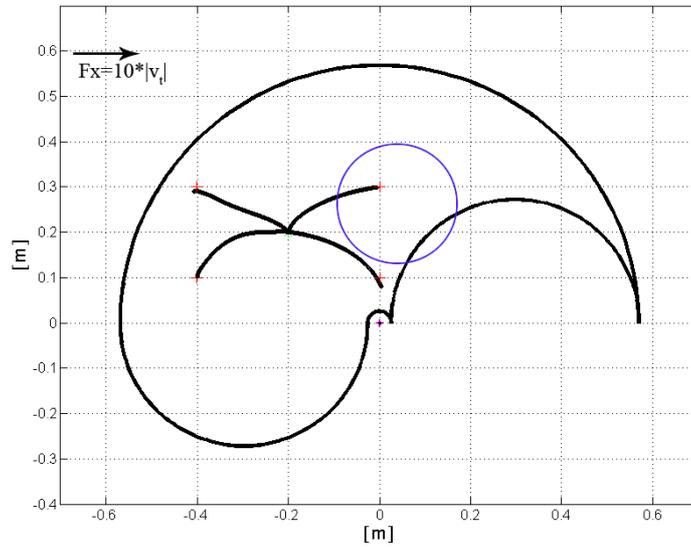
Figure 4.16 depicts the trend of absolute error position, showing a mean value of **7.5cm**.



Afterwards, the neural controller has been trained in the modified environment.

As a result, it came out that the additional training needed by the artificial neural network to be able to cope with this force required only from 2% to 4% of the epochs necessary for the training all over the working plane for unobstructed movements.

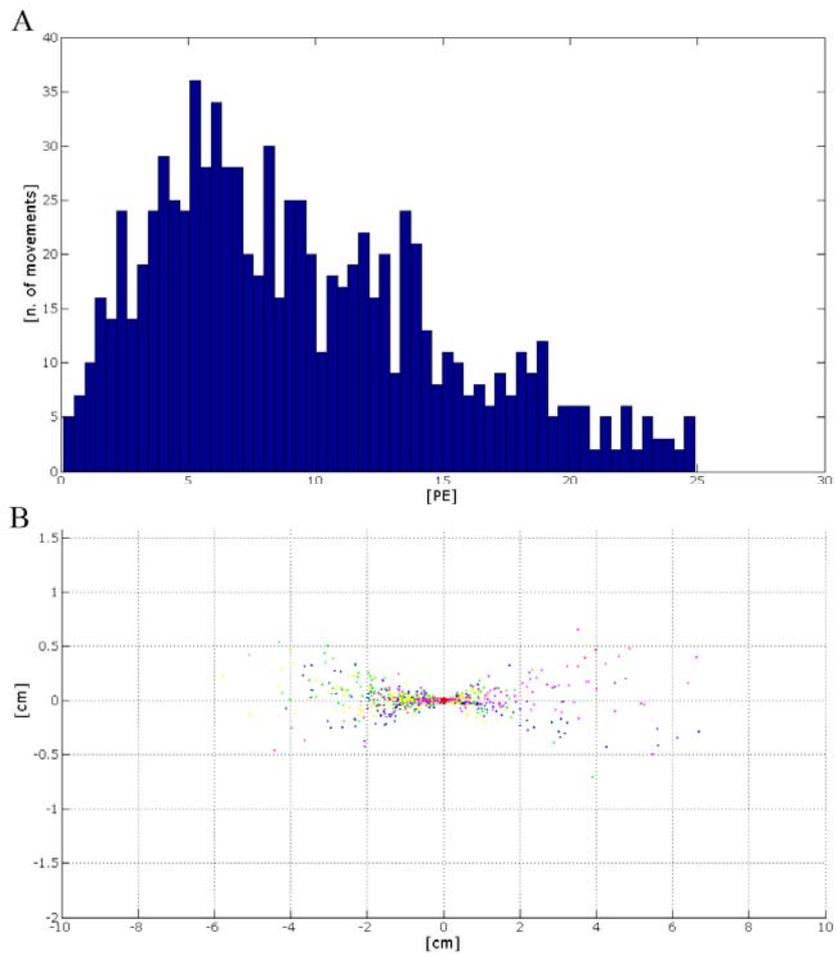
After the additional training, in the testing phase, the model showed a behaviour similar to those obtained with no force. In figure 4.17 it is possible to observe the behaviour of the system in the field force and after the short learning phase in the new environment.



**Figure 4.17** – Trajectory toward 4 points symmetrically located from the starting point under the presence of the a force acting along the x axis.

*The controller has been able to modify the weighted connections thus generating the internal model both of the inverse dynamics of the upper limb and of the dynamics interactions with the environment.*

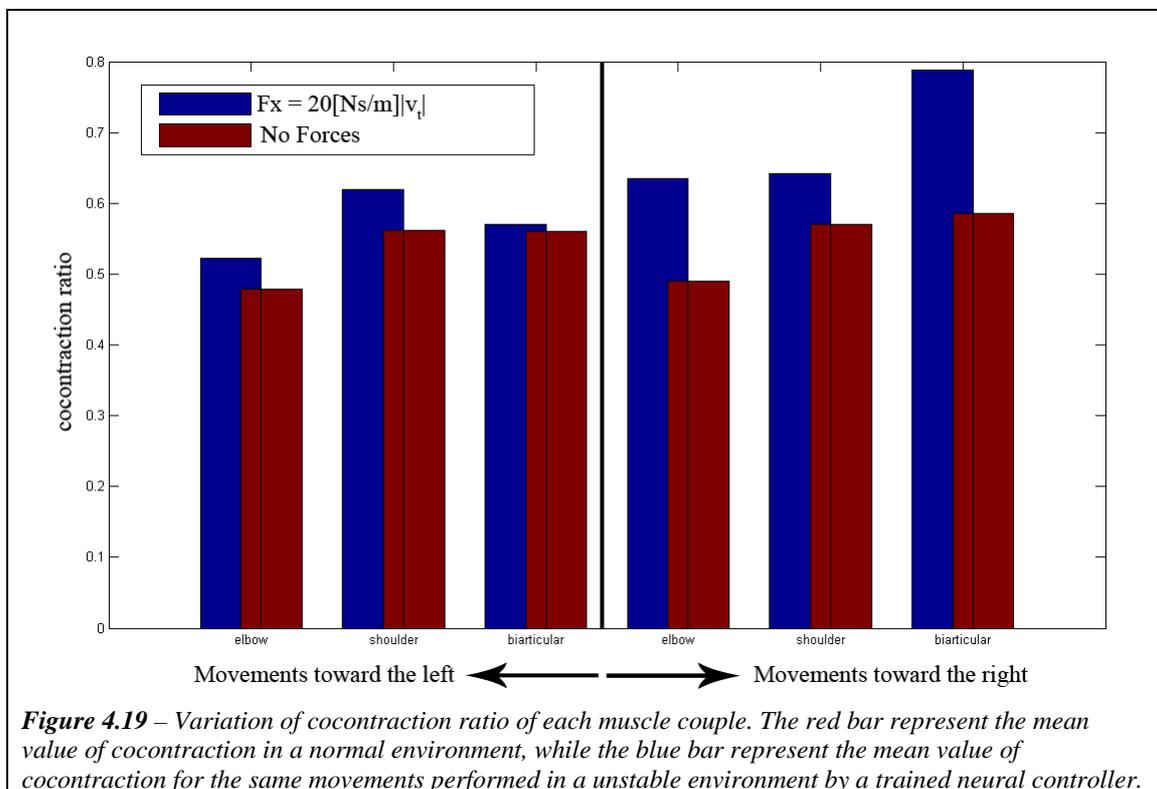
In figure 4.18A and 4.18B the histogram of the percentage position error and the distribution of the arrival points with respect to the target are depicted: the neural controller learned how to face with the unstable dynamics acting on the end effector. The mean error position decreases to a value of about to **1.4cm**.



**Fig 4.18** –Histogram of the percentage error with respect to the length of the movement (A) and space distribution of the arrival point with respect to the targets (B)

Another significant result concerns the evidence of the high adaptability of the neural controller which, as a biological controller, exploit the possibility to change the overall stiffness to overcome the environmental constrictions.

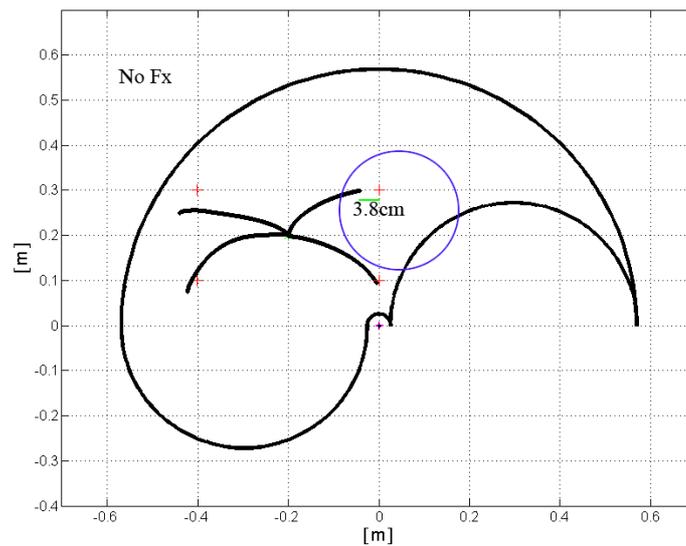
200 movements with different starting point and target have been taken into account: 100 of them showing a direction toward right with respect to the starting point and 100 toward left. The mean values of cocontraction ratio between flexor and extensor of each couple considered in the model in a normal environment have been recorded. Subsequently the same movements have been performed by a trained neural controller acting in a environment with the presence of a force directed along the x axis.



The results clearly show that in the movements pointing to the right of the starting point, when the extensor muscles operate as agonists, the cocontraction ratio is much higher in case of instability of the environment: this is due the fact that the flexor muscle has to face not only the action of the extensor muscle, but also of the external force that

combines its effect with this one. The neural controller learn to increase the overall stiffness acting on the level of cocontraction of the flexor muscle.

The last result has been tested in order to emphasize the presence of the so-called “after-effects” [21]: the force field ( $F_x=10[Nm/s]*|vt|$ ) active during the training phase has been removed and the absolute error position has been evaluated. In figure 4.20 it is possible to see the trajectories toward four targets; the neural controller generates the motor commands taking into account the non-existing unstable dynamics of the environment, trying to compensate them, and this leads to distortions of the trajectories in the opposite direction (compare with figure 4.14 and figure 4.17).

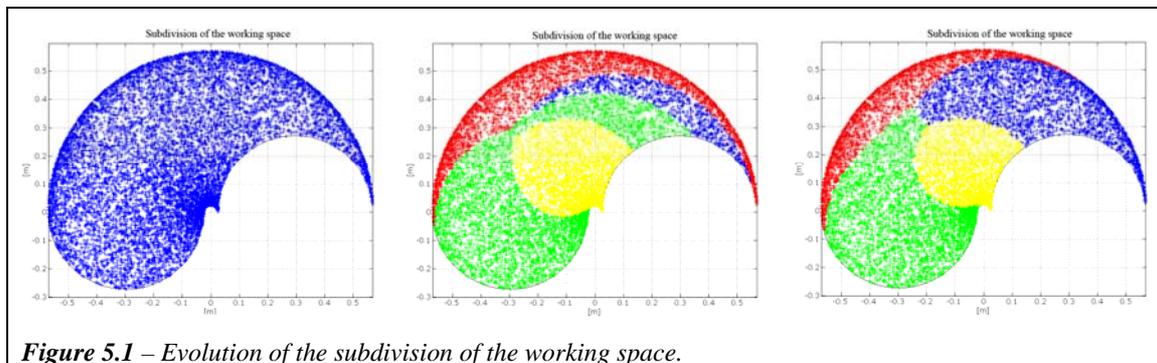


**Figure 4.20** – Trajectory toward 4 points symmetrically located from the starting point in an environment with no external disturbances.

## Hierarchical Model: Results and Observations

In order to evaluate the performance of the hierarchical system, a comparison study between this model and a single controller has been carried out. The same set of starting and target point have been provided to the two structures and the results have been analyzed.

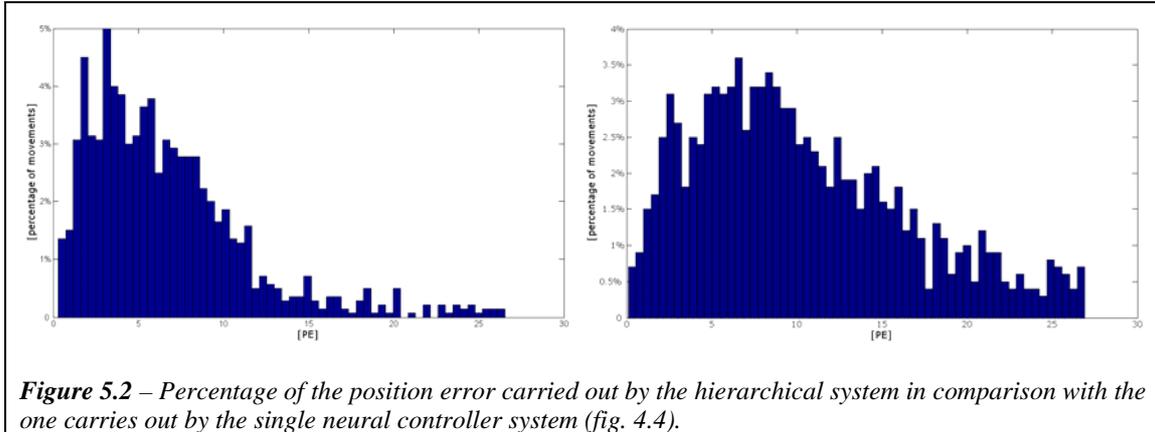
The figure 5.1 it is possible to observe the dynamic evolution of the high level controller which, in a unsupervised way, performs a subdivision of the working plane in 4 different zones related to 4 different neural controllers. It is interesting to observe that the final configuration can be reach in less than **1000** input patterns.



*Figure 5.1 – Evolution of the subdivision of the working space.*

Also in this case movements whose amplitude is included between 5 and 60cm have been taken into account.

As expected, in this coarse-to-fine approach, the mean position error is lower than in the single controller system: **1.91cm** with a standard deviation of about **2cm**. The figure 5.2 depicts the histogram of percentage error of the movements analyzed.



**Figure 5.2** – Percentage of the position error carried out by the hierarchical system in comparison with the one carries out by the single neural controller system (fig. 4.4).

The values of the indexes of curvilinearity which have to be taken into account are similar to the ones reported by the single neural controller; this means that, undoubtedly, the hierarchical system performs a better correlation between the starting and the arrival point and the necessary muscular activations, but this correlation is optimized in function of the position error and not of the curvilinearity of the movements, which probably depends most on the limits derived from the design simplification of the biomechanical arm model.

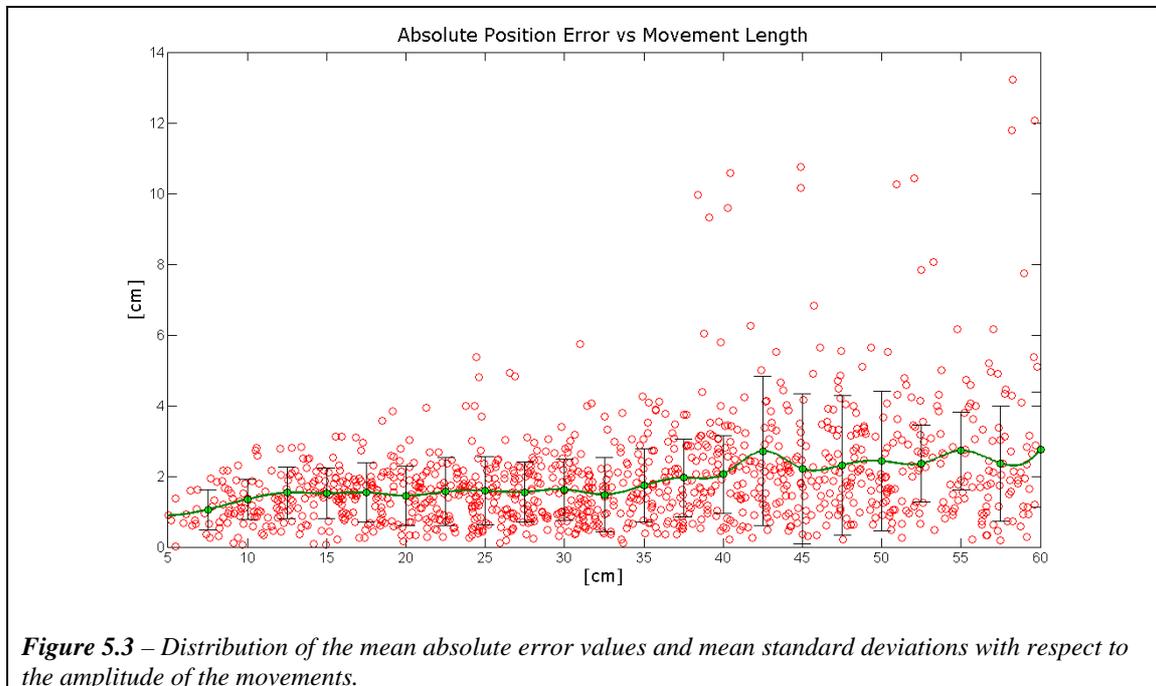
Normal Curvilinearity <b>NC</b>	1.09
Maximum Curvilinearity <b>MxC</b>	0.60 cm
Medium Curvilinearity <b>MdC</b>	0.57 cm
Total Curvilinearity <b>TC</b>	0.15 cm

**Table 5.1** – mean values of the curvilinearity indexes for the set of movements analyzed

The figure 5.3 shows the result related to the graphic of the distribution of the absolute position error and the standard deviation with respect to the increasing length of the movements analyzed. As in the previous cases (refer to Chapter 4), the values are evaluated within intervals of 5cm. starting from a value of amplitude of 5cm. until the maximum amplitude of 60cm.

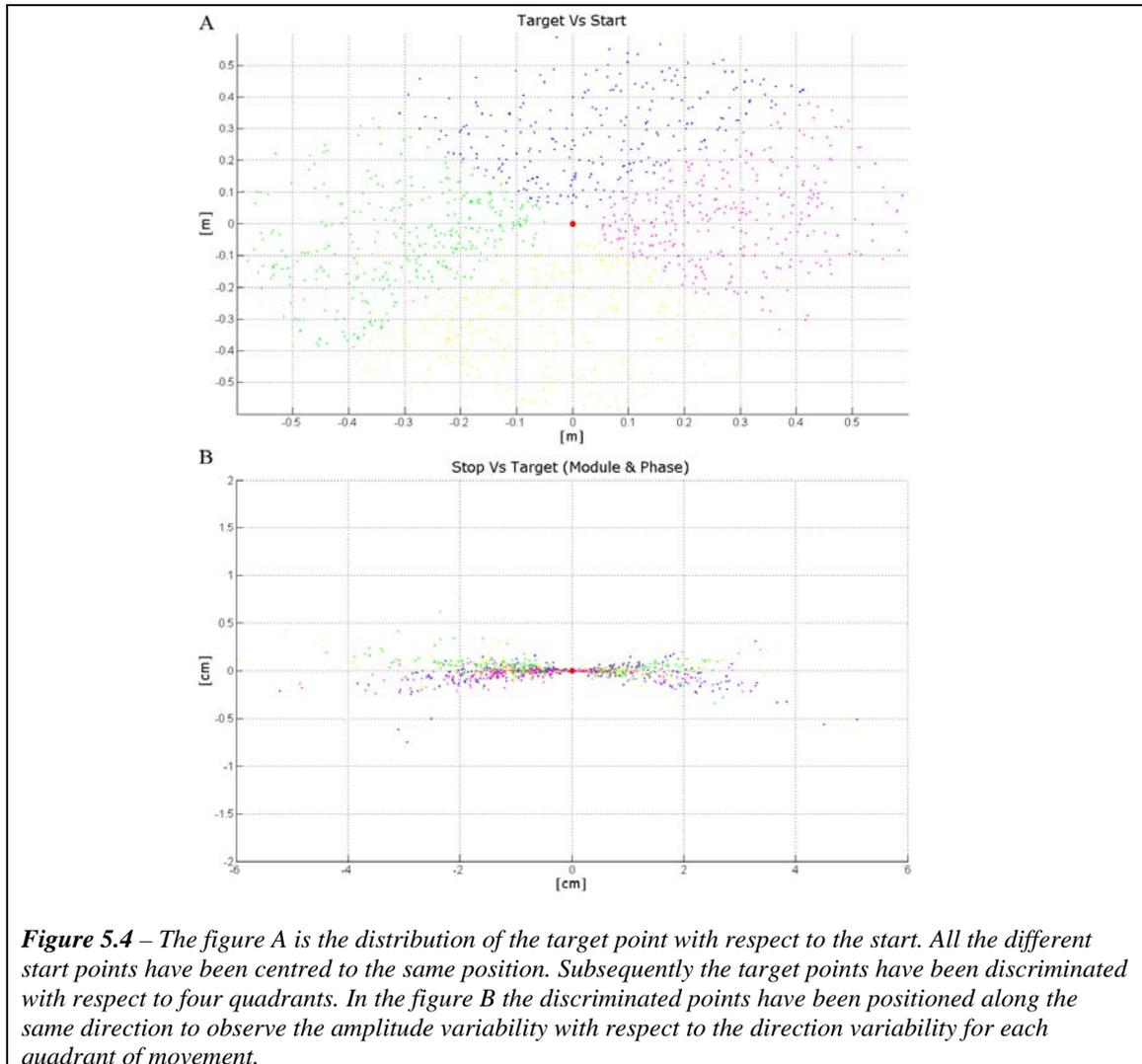
The black line is generated by means of a cubic spline interpolation (command spline.m from MATLABR2006©), while the height of the blue lines represents the value  $2*STD$  (where  $STD$  is the standard deviation). It is clear from the figure that in this case,

the mean absolute position error shows a very little variation with respect to the increase of the movement length.

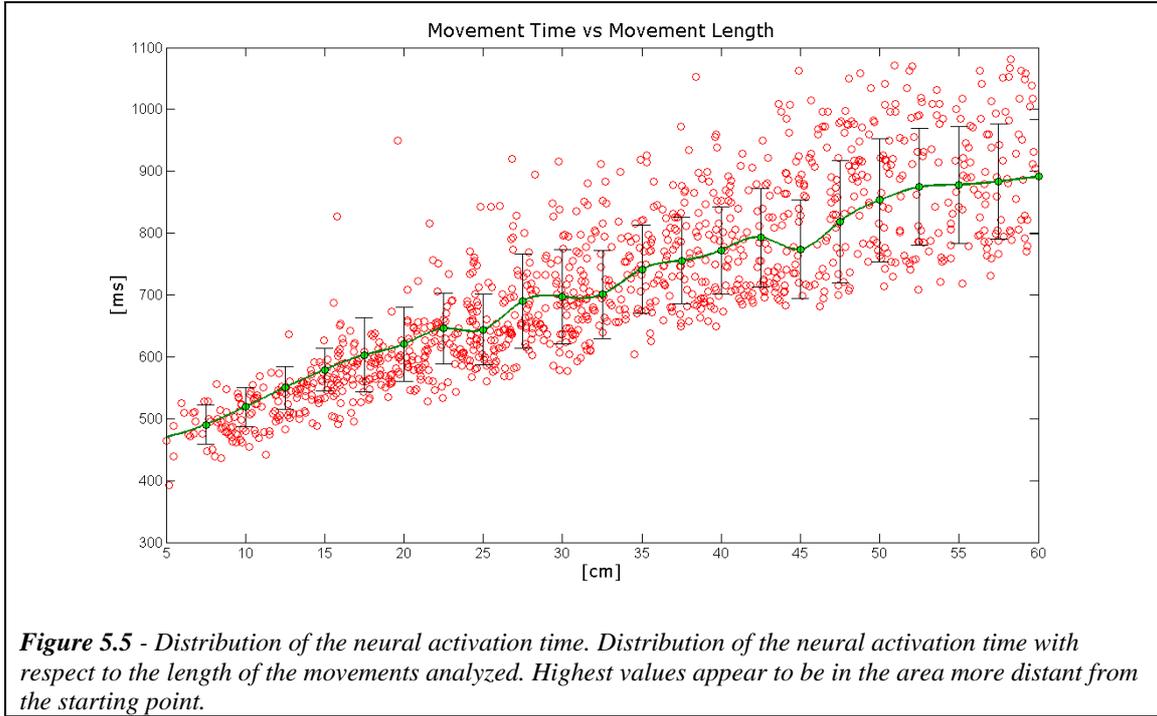


**Figure 5.3** – Distribution of the mean absolute error values and mean standard deviations with respect to the amplitude of the movements.

Again, if the module error and the phase error are analyzed, it is possible to observe a value that in both cases is lower than the single controller. In particular the module error shows a value of **-0.32cm**, thus highlighting the fact that there isn't a strong polarization on the amplitude error of the movement; moreover, the value of **0.006rad** for the phase error is a evident result consistent with the data observed in [96]. In this case the fact that the difference between the variability of the amplitude of movement toward a given target is typically greater than the variability of direction is even more marked, and the figure 5.4 can graphically shows this; it is possible to observe the distribution of the arrival points with respect to the desired target, assuming that the direction of the movement is specified by the line parallel to the x axis passing through the target point (compare with figure 4.8).



The last result concerns the distribution of the neural activation time with respect to the length of the movement. Also in this case it is possible to observe the global isochrony; there isn't a linear variation of the duration value with respect to the length of the movement: similar activations bursts are associated to similar movements.



**Figure 5.5** - Distribution of the neural activation time. Distribution of the neural activation time with respect to the length of the movements analyzed. Highest values appear to be in the area more distant from the starting point.

# **Chapter 5**

## *Neural Controller: Application*

## Summary

The work of the last year has been focused on the development of a functional application which might exploit the potentiality of the system presented in the current work. From a neurophysiologic and computational viewpoint the ability of the neural controller to adapt itself to the dynamics of the controlled object and to the dynamics acting on it is an interesting feature.

Movement functions of patients with neurological injuries are typically improved by adaptive equipments and environmental modifications [<sup>110</sup>] [<sup>111</sup>] which are not satisfactory from the point of view of the quality of life. In fact, the typical solutions to surrogate lost functions make the patients clearly perceive the loss of independence, and can often induce pain, anxiety.

In the literature, significant improvements in understanding the cellular and molecular events of injury and regeneration are reported and, even if clinical treatments employing these discoveries are not yet at hand [<sup>112</sup>], the deeper understanding of motor control and learning mechanisms [<sup>113</sup>] has strengthened the empirical foundations of the rehabilitative practice. In particular, long-term strengthening (i.e. where synapses are able to encode new information to represent a movement skill) has been considered to play a relevant role in restoring impaired functions.

A critical element for the success of these mechanisms resides in presenting a repetition of inputs for the motor cortex, which act as a biological teacher for the neurons acquiring novel skills. This process could easily be implemented through experience and training, which induce physiological and morphological plasticity, by strengthening synaptic connections between neurons encoding common functions [<sup>114</sup>]. Thus the key concept behind rehabilitation is the repetition of movements in a learning-by-examples paradigm: by repeating movements, in either passive or assisted way, the brain is exposed to reinforcement and the neurons can strengthen their connections.

Then the question to answer is how to make patients execute and repeat movements in an assisted way. Functional Electrical Stimulation (FES) is one of the technologies now used to restore functions of patients with neurological injury through electrical activation

of the muscular system. FES has grown to become an accepted therapy and treatment for subjects impaired by stroke, multiple sclerosis and infantile cerebral palsy [<sup>115</sup>], [<sup>116</sup>], [<sup>117</sup>]. This stimulation has overcome the simple functional limb substitution [<sup>118</sup>] to come up to the requirements of rehabilitation, and has been proven as successful both in lower [<sup>119</sup>] and in upper limb movements [<sup>120</sup>]. These encouraging results recently brought to the development of FES-assisted rehabilitation programs in paretic patients [<sup>121</sup>].

However, in order to increase the number of subjects who may benefit from this technology, novel and more sophisticated ways for the subject to command FES-generated movements are needed. Current technologies tend to use residual motor function [<sup>122</sup>] or EMG recordings from sound muscle activity [<sup>123</sup>]. However, this approach limits the application of FES to subjects with some remaining functions and excludes those with complete lesion of the cortico-spinal tract producing plegia.

In this general context, a smart FES (sFES) system can give rise to “an artificial teacher” that allows exploration of the workspace, thus representing a driver for different examples to be executed and then repeated.

The sFES system should overcome some of the limitations related to the use of FES in rehabilitation programs, due to the rather raw and un-physiological control of the stimulation, as well as the invasiveness of the approach. While for the latter issue, advancements in technology made it possible to obtain efficient non-invasive stimulators (see e.g. Handmaster [<sup>124</sup>] and the Bionic Glove [<sup>125</sup>]), the issue of biological plausibility of stimulation waveforms has not yet been deeply investigated, though some pioneering work is present in literature [<sup>126</sup>]. Therefore, the resolution of the inverse dynamics, i.e. the extraction of muscular forces needed to execute a specific movement in a particular environment, is one of the key problems to be solved to efficiently drive the stimulation. To this end, Artificial Neural Networks (ANN) have been hypothesized as biologically plausible controllers [<sup>127</sup>], and then shown as an efficient tool in the resolution of the inverse problem [<sup>128</sup>].

The neural controller developed in this work can implement a high level motor controller receiving inputs from the patient who could indicate his/her intention to make a

specific movement. The focus is therefore shifting from a control driven step-by-step by the patient (for instance, with the contraction of residual muscles), to the use of high level motor controllers, that is systems that could implement the whole control of the end effectors once the subject has decided the action to be implemented (i.e. move the arm from position A to B, grasp an object and so on) [<sup>129</sup>].

In this perspective, after receiving the information regarding the movement to be implemented, the stand alone controller could drive a stimulator block to make the arm move in the requested way. The rehabilitation exercise will thus consist of movements shown by a “teacher” and reproduced by the patient helped by a stimulator block driven by the controller.

Following this approach, the aim of the current work is to provide a general framework for the integration of three blocks that could constitute a stand-alone rehabilitation system for the upper limb, also for an in-house rehabilitation perspective, that is:

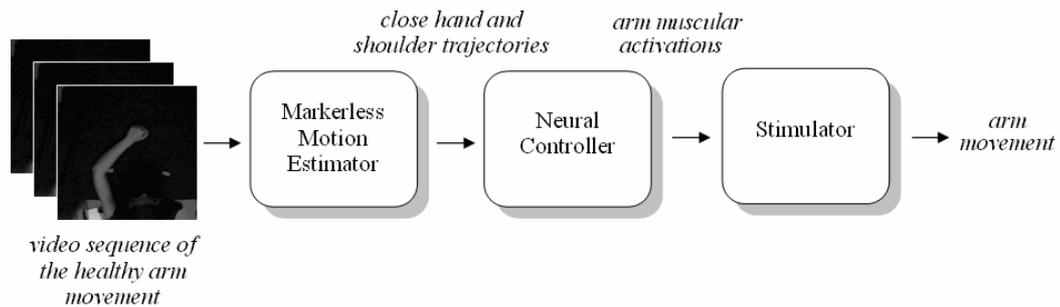
- a motion tracking system providing information on the movement to be executed. This system will estimate the desired trajectory (obtained from the images of any subject who could implement the movement to be reproduced through the sFES system) by using a markerless silhouette tracking approach;
- a neural controller solving the inverse dynamic problem to obtain the proper stimulation for the desired movement;
- a stimulator block serving as effector to move the arm.

In synthesis, the sFES will be driven by the integration of a markerless system for movement tracking with ANN to control the muscular stimulations.

In particular, this chapter exposes the work as it is at the actual state thus dealing only with the first two blocks of the system, that is the recognition of the movements, and the use of the high level motor control.

## Materials and Methods

Figure 6.1 shows a non formal flow diagram of the proposed method, while in the following subparagraphs the first two blocks are described in detail.



**Figure 6. 1** – Block diagram of the proposed system

### The markerless motion estimation method

The markerless motion estimation method aims at estimating the movement of the entire arm, dealing with the problems related to the high deformability of the human silhouette, that prevents the use of a rigid body approximation [130], [131], [132]. In order to combine the visual system with the neural controller only planar arm movements have been considered.

For silhouette extraction, energy-minimising deformable models, such as the Active Contour Model, called Snake, offer a partial solution, and have been widely applied in literature for segmentation and contour detection (vehicular traffic monitoring, surveillance or medical images segmentation [133]).

A Snake is a contour representation defined by a fixed number of points, called contour points (CP), whose matching to the image contour is achieved by minimizing a cost function, defined “energy” [134]. The contour is a controlled discrete spline and the snake position can be represented parametrically by a sequence of samples  $v(s)$ .

$$\mathbf{v}(s) = (x(s), y(s))$$

The energy expression in the case of  $N$  contour points  $CP(i)$  ( $i=1, \dots, N$ ), where the samples  $\mathbf{v}(s)$  are evaluated at  $s=s_i$ , is the following:

$$E_{tot} = E_{int} + E_{ext} = \sum_{i=1}^N E_{CP(i)}$$

The internal energy  $E_{int}$  can be written as a functional which includes the inter-points distance and the contour curvature

$$E_{int} = \frac{\alpha \left| \frac{d\mathbf{v}}{ds} \right|^2 + \beta \left| \frac{d^2\mathbf{v}}{ds^2} \right|^2}{2}$$

where  $\alpha$  and  $\beta$  are respectively the measure of the elasticity and the stiffness of the snake. The first derivative term makes the snake act like a membrane, where the constant  $\alpha$  controls the tension along the contour. On the other hand, the constant  $\beta$  and the second order term governs the rigidity of the curve (if  $\beta$  is zero, the contour is discontinuous in its tangent, i.e. it may develop a corner at that point).

The external energy of the snake,  $E_{ext}$ , is derived from the image data in order to make the snake be attracted to lines, edges and terminations:

$$E_{ext} = E_{line} + E_{edge} + E_{term}$$

where

$$E_{line} \propto f(x, y)$$

$$E_{edge} \propto |\nabla f(x, y)|^2$$

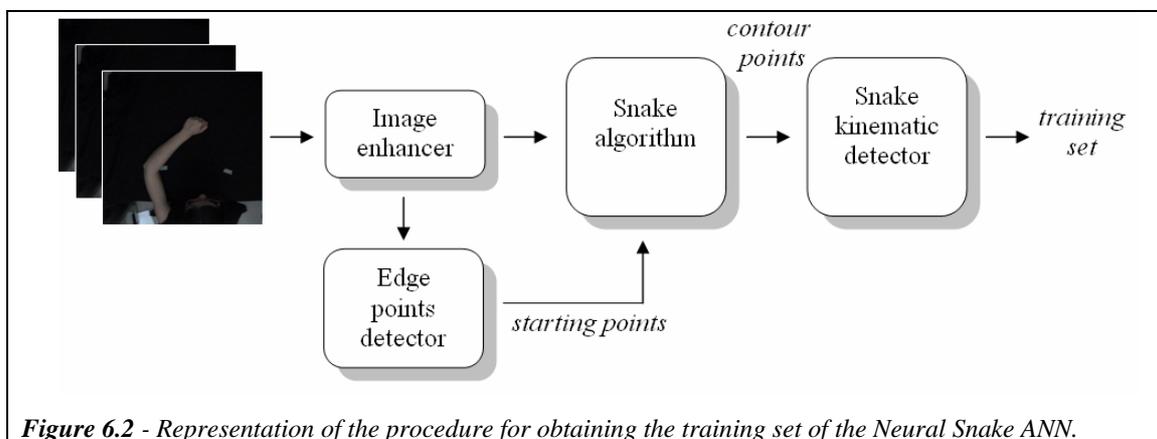
$$E_{term} \propto \frac{\partial \theta(x, y)}{\partial n_r}$$

and  $f(x,y)$  is the image intensity,  $\theta(x,y)$  is the gradient direction along the contour and  $n_r$  is an unit vector perpendicular to the gradient direction.

The snake algorithm is mainly suited for static or slightly varying shapes, but in a dynamic context, such as human body tracking, the constraint of shape preservation during the movement appears to be restrictive. As a matter of fact, in human movement analysis it is often needed to track silhouettes which greatly change from frame to frame (because of fast movements, such as the ballistic ones, or because of low acquisition frame rates).

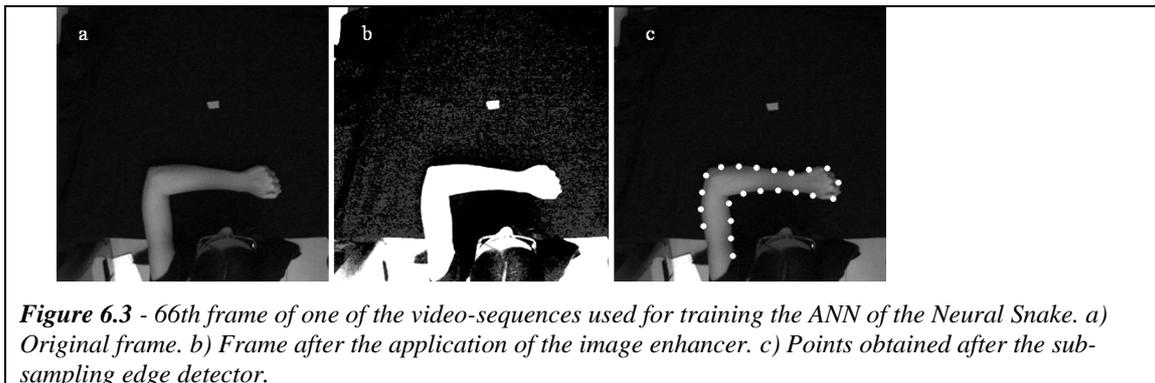
In order to apply the Snake algorithm in this dynamic context, the present study introduces a new approach, called Neural Snakes (NS). The algorithm is based on the use of an ANN which acts as a predictor for the shape of the contour, thus making a coarse estimation of its future position, which constitutes the starting point of a subsequent closer approximation (in a coarse-to-fine approach).

Figure 6.2 shows the construction of the training set for the ANN.



**Figure 6.2** - Representation of the procedure for obtaining the training set of the Neural Snake ANN.

Any frame of the video-sequences representing the arm movement undergoes an edge detection procedure: it is first converted to greyscale, then the distribution of its histogram is modified by using the VirtualDub program [135], in order to increase contrast (both a contrast of 200% and a sharpening filters are used). Then, after filtering by a 5-by-5 median filter window, the arm silhouette is extracted as reported in Canny [136] (Figure 6.3).



**Figure 6.3** - 66th frame of one of the video-sequences used for training the ANN of the Neural Snake. a) Original frame. b) Frame after the application of the image enhancer. c) Points obtained after the sub-sampling edge detector.

The upper limb edge is then uniformly sub-sampled, thus preserving only a relatively low number of points of the contour in order to limit the computational burden of the algorithm (for frames shown in Figure 6.3 the number of points is 22) while maintaining the possibility of tracking movements/deformations of the contour. The edge-points are then used as starting points for the Snake algorithm as reported in Kass

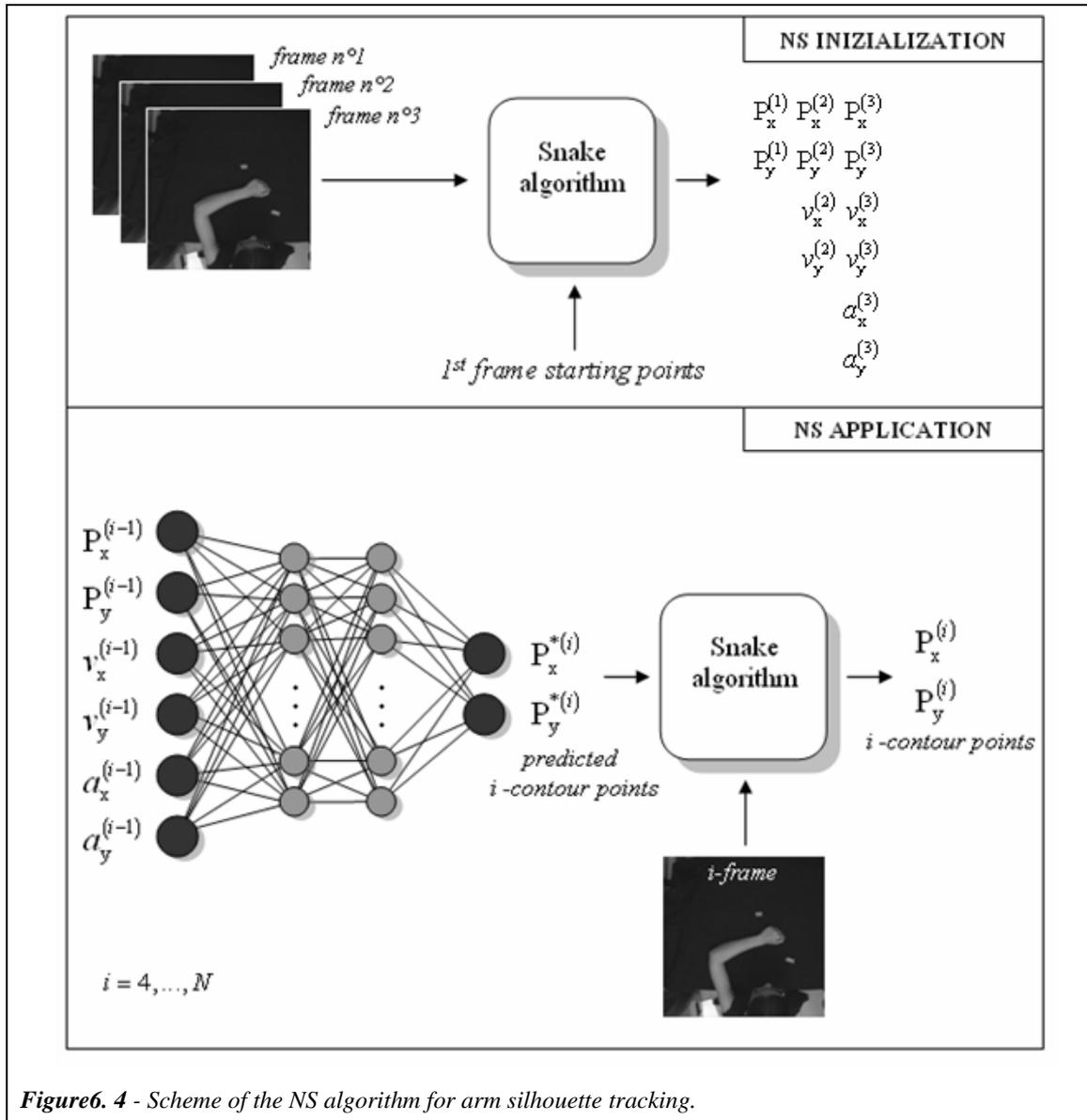
The resulting horizontal and vertical positions of the contour points are used to calculate velocities and accelerations over time.

The kinematic data of the contour points extracted by the video-sequences represent the training set of the ANN which is a multilayer Perceptron composed of 2 hidden layers with 15 neurons each, chosen after a trial-and-error optimisation with respect to complexity, accuracy and real-time implementation. The network is fed by the horizontal and vertical components of position, velocity and acceleration of each contour point in the current frame (which means that the number of the input neurons is the number of contour points multiplied by 6), while the output is given by the horizontal and vertical components of the position of each contour point in the subsequent frame (number of points multiplied by 2).

For the training, a Resilient Back Propagation (see Appendix A) algorithm has been chosen. At the end of the training of the ANN (2000 epochs were necessary for convergence), the NS algorithm can be applied.

After an initialization phase, for each  $i$ -th frame of the video-sequence, the Snake Algorithm inputs are both the  $i$ -th frame and by the positions of the CPs on the  $i$ -th frame, as predicted by the ANN on the basis of the kinematic data of the  $(i-1)$ -th frame (Figure 6.4).

The initialization phase is necessary to have the initial kinematic data of the CPs (i.e. position, velocity, acceleration). The first three frames of the video sequence are therefore necessary for the initialization phase, while the subsequent  $N-3$  frames ( $N$  being the total number of frames of the video-sequence) are elaborated by applying the Snake algorithm to the output of the ANN (the  $M$  predicted contour points  $P^*x$  and  $P^*y$  for each frame). The result is the estimated silhouette over time.



**Figure6. 4** - Scheme of the NS algorithm for arm silhouette tracking.

The CPs positions obtained by the NS approach are then used to estimate the close hand and shoulder trajectories in order to obtain the biomechanical analysis of the gesture.

The method has been tested on synthetic video-sequences [137] in order to evaluate its accuracy in tracking the arm silhouette. The test results show that the mean error of the NS algorithm in determining the movement of the barycentre of the hand during planar movements is typically between 1 and 2cm. Therefore, it can be confidently used to determine the movement of the arm to be provided to the neural motor controller.

## **The proposed neural controller of the upper limb model**

The trajectory's parameters extracted by the NS algorithm are used to drive a neural controller which activates a biomechanical model of a simulated human arm. To this purpose, a second ANN (ANN2) has been used to implement the neural controller which solves the inverse dynamic problem giving rise to the movements. The controller, after knowing the movement intended by the patient (that can be simply specified in terms of starting and ending coordinates of the movement), generates the neural activations that will make the muscles exert the forces necessary to drive the arm model. The process of transformation of the neural controller output to the activation commands necessary to pilot the biomechanical upper limb and details on the implementation of the neural controller can be found in chapter 2.

The ANN2 expressly developed for this study has been designed by using a Multi-Layer Perceptron with two hidden layers, is fed by four inputs, and generates 4 output, on the basis of the one presented.

## **Experimental trials**

Experimental trials have been designed and implemented in order to assess the performance of the proposed system. Tests aim at assessing the capability of the neural motor controller to make the arm execute movements corresponding to those determined by the markerless algorithm that tracked the movements made by an “external teacher”.

The experiments follow this rationale:

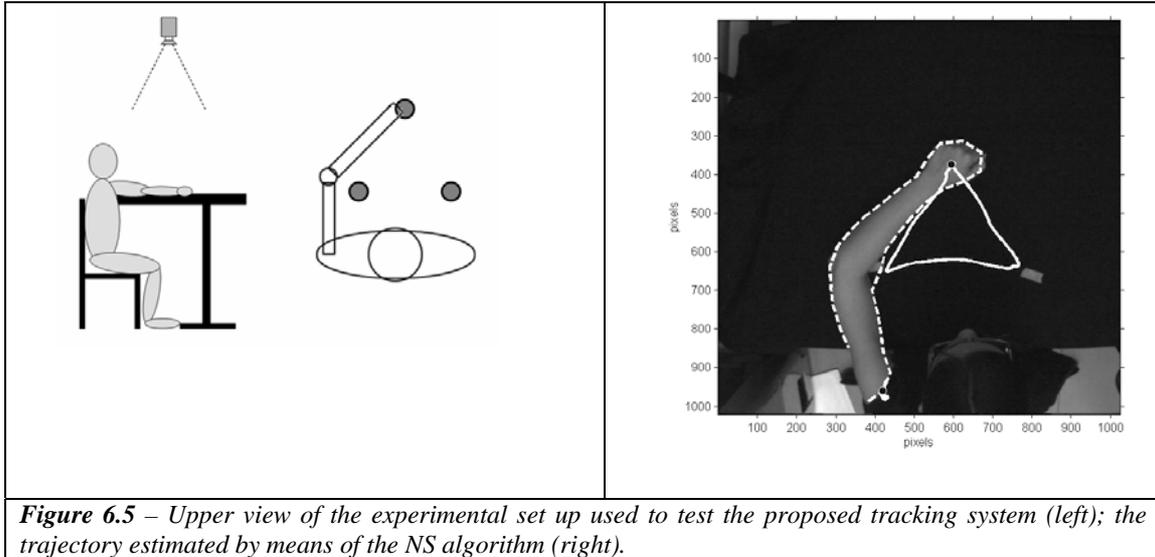
- an “external teacher” (i.e. a rehabilitator, a physician, a healthy subject etc.) executes a planar movement with the arm. This movement is the “executed movement” and will be indicated by the subscript  $e$ ;
- the movement is video-recorded;

- the NS algorithm processes the video-sequence and estimates the kinematic data of the movement's trajectory (i.e. coordinates of the starting and ending points of either the movement or the sub-movements in which the entire movement can be subdivided) which feed the neural controller;
- the neural controller (i.e. the integration of the ANN2 and the Pulse Generator) produces the neural activations to make the biomechanical arm move on the plane giving rise to the reconstructed movement that will be indicated by the subscript  $r$ ;
- the reconstructed movement is compared to the one executed by the teacher.

The set-up used for the recording of the movements is shown in Figure 6.5.

The “external teacher” sits on a chair in front of a desk whose height is the same of the subject's arm-pit, with the trunk close to the desk border. In this way the upper limb movements on the desk are planar. The arm has 2 DOF, with wrist joint locked, and shoulder and elbow joints are allowed to move in flexion-extension. The motor task consists of 10 repetitions of a sequence of 3 counterclockwise fast reaching movements toward 3 target points on the table surface (the movement's trajectory can be represented by a triangle). The subject executes the movement with his dominant hand, and the barycentre of the closed hand is considered as the end-effector.

The movement has been recorded by a digital video camera (Silicon Imaging MegaCamera SI-3300RGB) from an upper view, by using a temporal resolution of 60 frame/s and a spatial resolution of 1024 x 1020 pixels. The videos have been composed of 250 frames.



The NS algorithm has been applied to the video-sequences to estimate the positions of both arm and shoulder over time. The shoulder has been considered as the centre of the reference system while the positions of the end-effector in each of the three movements composing the sequence represented the estimated target positions. These estimates are the only information provided to the neural motor controller that drives the “reconstructed movement”.

The differences between the estimated (by the NS) and the reconstructed movements on the target points have been evaluated in terms of mean value and standard deviation.

Moreover, the curvature of the reproduced movements has been chosen as a further parameter to evaluate the system performance. In the literature, it is reported that ballistic movements are typically smooth (see for instance Morasso [48]), with a limited curvature, for which different definitions are given ([138], [139] and [106]).

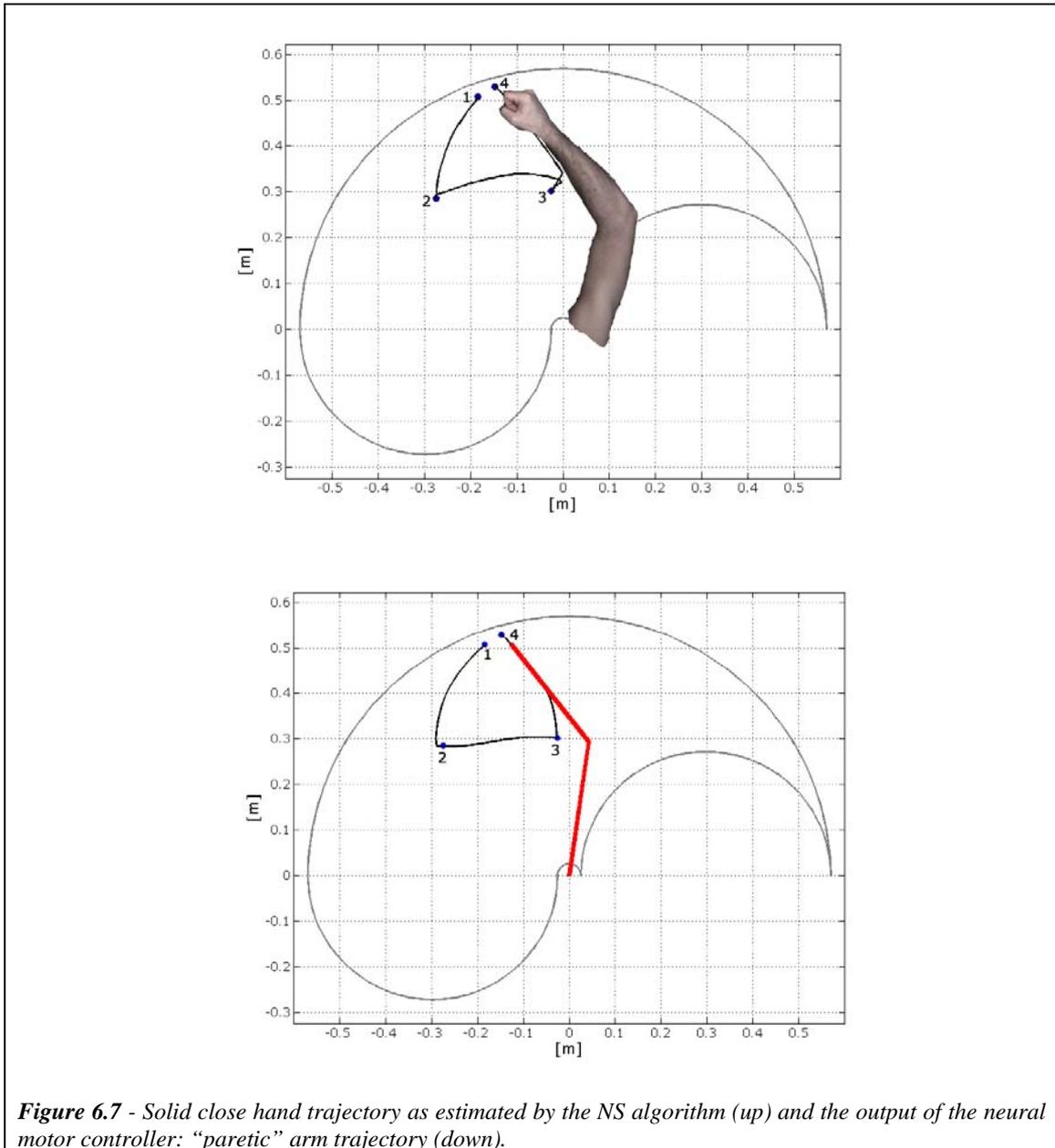
Here we report only the comparison with the definition in [46], where the curvature is defined as the ratio between the length of real trajectory and the Euclidean distance between the starting and the ending points of the movement, according to the equation:

$$C = \frac{\sum_{i=1}^{N-1} \sqrt{dx_i^2 + dy_i^2}}{\sqrt{(x_f - x_s)^2 + (y_f - y_s)^2}}$$

The numerator indicates the length of the movement carried out, and  $x_s$  and  $x_f$  are the starting and final points of the trajectory. We think that this definition gives a more comprehensive and robust index, with respect to other definitions which consider point values (the maximum deviation from straight line or the deviation in mid-trajectory) or simply the mean value of the deviation all over the trajectory.

## Results

In Figure 6.7 an example of both the executed and the reconstructed movement over the arm workspace is reported.



The mean absolute error, that is the difference between the estimated and reproduced movement, in 10 repetitions of the same triangular movement, together with the standard deviation resulted:

2,3 cm  $\pm$  0.56 cm for movement 1 – 2 (direction 225 °)

0.9 cm  $\pm$  0.46 cm for movement 2 – 3 (direction 0 °)

1.2 cm  $\pm$  0.90 cm for movement 3 – 1 (direction 135 °)

The average curvature of the real movements and reconstructed movements resulted to be 1,03 for real movements (similar to the results reported in [<sup>46</sup>]) and 1,06 for the reconstructed ones, thus showing a good agreement, not only for the final points but also for the trajectory followed.

## Discussion

The results obtained show that the proposed biologically inspired neural motor controller, together with a markerless algorithm able to track the rapidly changing silhouette of the moving arm, can drive an arm model in order to reconstruct planar unobstructed movements of the arm using the information on arm movement, and specifically on the initial and final point of the movement.

The accuracy in reproducing the movement of the arm silhouette on the plane by means of the neural motor controller has been evaluated, and is fairly adequate for the intended application. In fact, the difference between the position of the end-effector (as estimated by the NS algorithm) and the reproduced one through the use of the biologically inspired neural motor controller, has average errors of 1,4cm with limited standard deviations and a similar curvature.

In conclusion, a neural system able to drive a biomechanical arm model in order to reproduce planar movements made by any subject has been proposed.

One interesting feature of the proposed approach is that an adequate model of the arm can be trimmed to any specific subject and used in the neural controller.

In the present version, the system can implement only planar unobstructed movements, but it is adequate to show the proof of the concept.

A study of the kinematic parameters of the movement has been carried out in order to underlie the capacity of the system to achieve this task.

The third block, not presented here, is a FES system, able to receive the neural commands from the neural controller and correspondently stimulate the arm of the subject, in order to make him do the intended movements. In this way, an innovative smart FES (sFES) system can be implemented.

The availability of the proposed system opens interesting perspectives for use together with systems of virtual/augmented reality for a timely rehabilitation of the arm movements in injured (stroke, multiple sclerosis) patients.

# **Chapter 7**

## *Conclusion*

Clarifying the principles underlying sensorimotor control represents a hot challenge, which does not only fascinate researchers in the field of Neurology, Neurophysiology, Psychophysiology, and Theoretical Neuroscience, but is also sought by researchers in the field of Biomedical Engineering, with the common aim of providing the scientific framework in which the functional aspects are enclosed, of shedding light on the mechanisms underlying changes in sensorimotor function driven by the presence of pathologies, and of deploying rehabilitative tools targeted to the restoration of function in patients. In this general context, human sensorimotor function has to face with a non trivial problem constituted by controlling a basically unreliable system in a dynamic environment, by using a redundancy of not-so-fast biological sensors, generally characterized by a not more than fair signal to noise ratio. Modelling the solution to this control theory problem by means of a traditional means not only neglecting the biological foundations of the human nervous system, but also not taking advantage from the intrinsic adaptability coming from physiological systems. It is thus clear that, if researchers wanted to model the functioning of the sensorimotor control either to increase the knowledge in this field, or to provide a synthetic solution to the presence of a pathology in this area, a distributed approach enclosing artificial neural networks would represent the most plausible solution, both in terms of efficiency and in terms of fault tolerance, generalization and adaptability. In this general context, the work presented here gives a contribution in this perspective, by proposing a smart distributed framework that models the functioning of the human sensorimotor control in the movements of the upper arm.

In particular, after detailing the overall structure of the human sensorimotor control replica presented in the work, the third chapter introduces an innovative framework for the implementation of the learning process, called Biological Learning Paradigm, which takes its foundation on the principle of the exploration-learning association. In the same chapter, the bases for a new scheme involving a hierarchical control of more general movements are detailed. The results obtained in chapter 4 provide the reader with information on the accuracy of the learning process of the scheme, both in absolute terms, and in comparison with real movements of the upper arm obtained on different experimental campaigns on young adults. The sixth chapter describes the proposal of an

application of the entire scheme, as the core of a system for the rehabilitation and assistance of patients with severe upper arm motor disorders, thus setting the bases for a new smart functional electrical stimulation, controlled at a higher level than what is generally proposed not only in the scientific literature, but also in the market.

# Appendix A

## Back-Propagation

Although the back-propagation might be applied on networks with any number of layers, only one layer networks has been demonstrated to be adequate to solve and approximate whichever function with a limited number of discontinuities, while considering the activation functions of the single unit non linear. (Hornik, Stinchcombe & White, 1989; Funahashi, 1989; Cybenko, 1989; Hartman, Keeler & Kowalski, 1990).

The problem that it had to face while training multi layer networks is related to the use to a updating mechanism similar to the delta rule (in which the error is evaluated as the difference between the desired and the real output of each unit). With this system it is possible to update only the weights connected to the output neurons, but not the weights connected to the neurons of the intermediate layers. In fact, while for the output layer the desired output vector is well-known, nothing is known about the desired output of the hidden neurons.

The problem was solved in 1986, when the backpropagation algorithm was introduced. This algorithm is based on the evaluation of the error calculated on the output neurons which are connected to a unit of the previous layer. This process is repeated for all the neurons of the layer. The backpropagation algorithm considers that for each example of the training set, the signals travel from the input to the output in order to evaluate the answer of the net. Subsequently there is a second phase in which the error signals are back-propagated along the same connections on which, during the first phase, the input signals travel. During this phase the weights are modified. The weights of the neurons are initialized with random values. Practically the back propagation uses a generalization of the delta rule. The activation function is a differentiable function of the total input, given by:

$$y_k^p = \mathfrak{S}(s_k^p) \quad \text{where} \quad s_k^p = \sum_j w_{jk} y_j^p + \theta_k$$

For a generalization of the delta rule, it becomes necessary to follow these modifications:

$$\Delta_p w_{jk} = -\gamma \frac{\partial E^p}{\partial w_{jk}}$$

The measure of the error  $E_p$  is defined as the total squared error for the pattern  $p$  of the output units:

$$E^p = \frac{1}{2} \sum_{o=1}^{N_o} (d_o^p - y_o^p)^2$$

where  $d_o^p$  is the desired output for the single unit.

Moreover it's possible to compel  $E = \sum_p E^p$  as sum of the squared errors. It's possible to write:

$$\frac{\partial E^p}{\partial w_{jk}} = \frac{\partial E^p}{\partial s_k^p} \frac{\partial s_k^p}{\partial w_{jk}}$$

From the equation written above it's possible to observe that the second factor is similar to:

$$\frac{\partial s_k^p}{\partial w_{jk}} = y_j^p$$

When  $\delta_k^p = -\frac{\partial E^p}{\partial s_k^p}$  it's considered, it's therefore possible to obtain a updating rule

which proves to be proportional to the descending gradient on the error surface if:

$$\Delta_p w_{jk} = \gamma \delta_k^p y_j^p$$

Theoretically the value of  $\delta_k^p$  should be known for each k unit of the net, but there is a recursive procedure for this calculus that can be performed back-propagating the error signals on the net.

It is possible to write the partial derivative which represents  $\delta_k^p$  as the product of two factors: the first one reflects the error variation based on the output of the single unit, while the second factor shows the output variation in function of an input variation. It is possible to obtain:

$$\delta_k^p = -\frac{\partial E^p}{\partial s_k^p} = -\frac{\partial E^p}{\partial y_k^p} \frac{\partial y_k^p}{\partial s_k^p}$$

It's possible to calculate the second factor noticing that  $\frac{\partial y_k^p}{\partial s_k^p} = \mathfrak{S}'(s_k^p)$ , which represents the derivate of the output function of the single unit.

In order to compute the first factor of the derivate, two different possibilities have to be considered:

Case A) The unit  $k$  is a net output unit  $k=o$ . In this case:

$$\frac{\partial E^p}{\partial y_o^p} = -(d_o^p - y_o^p)$$

That is the same result obtainable with the common delta rule. It's possible to achieve the following result:

$$\delta_o^p = (d_o^p - y_o^p) \mathfrak{F}'_o(s_o^p)$$

for each output unit  $o$ .

Case B) If  $k$  is a hidden unit  $k=h$ , it's impossible to know the contribution of the unit to the output error of net. Anyway the error value can be written in function of the net inputs from the hidden layers to the output layer:

$$E^p = E^p(s_1^p, s_2^p, \dots, s_j^p, \dots)$$

Thus it is possible to obtain:

$$\frac{\partial E^p}{\partial y_h^p} = \sum_{o=1}^{No} \frac{\partial E^p}{\partial s_o^p} \frac{\partial s_o^p}{\partial y_h^p} = \sum_{o=1}^{No} \frac{\partial E^p}{\partial s_o^p} \frac{\partial \sum_{j=1}^{No} w_{ko} y_j^p}{\partial y_h^p} = \sum_{o=1}^{No} \frac{\partial E^p}{\partial s_o^p} w_{ho} = - \sum_{o=1}^{No} \delta_o^p w_{ho}$$

Substituting this equation:

$$\delta_h^p = \mathfrak{F}'_h(s_h^p) \sum_{o=1}^{No} \delta_o^p w_{ho}$$

The equations provides a recursive procedure for the evaluation of the  $\delta$  for each unit of the net, and through it, it's possible to achieve the values of the weights modifications.

The backpropagation uses an intuitive method: when the training set is available, the activation values are propagated to the output units, and these output values are compared with the desired ones; usually it is possible to achieve an error on each output neuron, defined as  $e_o$  and it is necessary to bring this error near to zero.

The simplest method is to modify the connections of the neural network in a way that on the following passage the error is zero; what is carried out by the back-propagation algorithm is a distribution the single output unit error between all the units it is connected to weighting these distribution upon the value of the same connections.

## Self-Organising Networks

The multilayer networks perform a mapping by presenting the network examples  $(x^p, d^p)$  where  $d^p = \mathfrak{F}(x^p)$ . Problem exist where such training data, consisting of input and desired output pairs are not available, or are not biologically plausible. In this cases he relevant information has to be found within the training samples  $x^p$ . There are a lot of such problems: clustering, vector quantization, dimensionality reduction or feature extraction. There are anyway many types of self-organizing networks, applicable to a wide area of problems. One of the most basis schemes is competitive learning as proposed by Rumelhart and Zipser (1985). A similar network but with different emergent properties is the topology-conserving map devised by Kohonen.

The self-organizing nets are based on the “winner takes all” training method. An example of a competitive learning network is shown in figure (??). All output units  $o$  are connected to all input units  $i$  with weights  $w_{io}$ . When an input pattern  $\mathbf{x}$  is presented, only a single output unit of the network (the winner) will be activated. In a correctly trained network, all  $\mathbf{x}$  in one cluster will have the same winner. For the determination of the winner and the corresponding learning rule two methods exist.

## Winner Selection: Dot Product

Assuming that both input vectors  $\mathbf{x}$  and weight vectors  $\mathbf{w}_o$  are normalised to unit length. Each output unit  $o$  calculates its activation value  $y_o$  according to the dot product of input and weight vector:

$$y_o = \sum_i w_{io} x_i = \overline{w}_o^T x$$

In a next pass, output neuron  $k$  is selected with maximum activation

$$\forall o \neq k : y_o \leq y_k$$

Activations are reset such that  $y_k = 1$  and  $y_{o \neq k} = 0$ . This is the competitive aspect of the network, and we refer to the output layer as the winner-take-all layer. The winner-take-all layer is usually implemented in software by simply selecting the output neuron with highest activation value.

Once the winner  $k$  has been selected, the weights are update according to:

$$w_k(t+1) = \frac{w_k(t) + \gamma(x(t)) - w_k(t)}{\|w_k(t) + \gamma(x(t)) - w_k(t)\|}$$

where the divisor ensures that all weight vectors were normalised. Only the weights of winner  $k$  are updated.

The weight update given in the previous equation effectively rotates the weight vector  $w_o$  towards the input vector  $x$ . Each time an input  $x$  is presented, the weight vector closest to this input is selected and is subsequently rotated towards the input. Consequently, weight vectors are rotated towards those areas where many inputs appear: the clusters in the input.

In case of unnormalised data, the winning neuron  $k$  is selected with its weight vector  $w_k$  closest to the input pattern  $x$ , using the euclidean distance measure:

$$k : \|w_k - x\| \leq \|w_o - x\|, \forall o$$

Instead of rotating the weight vector towards the input, the weight update must be changed to implement a shift towards the input:

$$w_k(t+1) = w_k(t) + \gamma(x(t) - w_k(t))$$

In particular the Kohonen networks (Kohonen, 1982-1984), can be seen as an extension to the competitive learning network. In these kind of nets, the output units  $S$  are ordered in some fashion, often in two dimensional grid or array. When learning patterns are presented to the network, the weights to the output units are thus adapted such that the order present in the input space  $\mathfrak{R}^N$  is preserved in the output, i.e., the neurons in  $S$ . This means that learning patterns which are near to each other in the input space (where 'near' is determined by the distance measure used in finding the winning unit) must be mapped on output units which are also near to each other, i.e., the same or neighbouring units. Thus, if inputs are uniformly distributed in  $\mathfrak{R}^N$  and the order must be preserved, the dimensionality of  $S$  must be at least  $N$ .

Usually the learning patterns are random samples from  $\mathfrak{R}^N$ . At the time  $t$ , a sample  $\mathbf{x}(t)$  is generated and presented to the network. The winning unit  $k$  is therefore determined and the weights to this winning unit as well as its neighbours are adapted using the learning rule

$$w_o(t+1) = w_o(t) + \gamma g(o,k)(x(t) - w_o(t)), \forall o \in S$$

Where  $g(o,k)$  is a decreasing function of the grid-distance between units  $o$  and  $k$ , such that  $g(o,k)=1$ .

# References

---

- <sup>1</sup> Morasso P, **Motor Control Models: learning and performance**, International Encyclopedia of the Social & Behavioral Sciences, Section: Mathematics and Computer Sciences, Pergamon Press, 2000
- <sup>2</sup> Sabes PN, **The planning of visually guided arm movements: feedback perturbations and obstacle avoidance studies**, Ph.D. Thesis, 1996.
- <sup>3</sup> Scott H, **Optimal feedback control the neural basis of volitional motor control**, Nature Neuroscience Reviews, 2004, 5, pp. 534-546.
- <sup>4</sup> Hadders-Algra M, Van Eykern LA, Klip-Van den Nieuwendijk AWJ, Prechtl HFR, **Developmental course of general movements in early infancy**, Early Human Development, 1992, 28, pp. 231-251.
- <sup>5</sup> Jansen-Osmann P, Richter S, Konczak, J, Kalveram, KT, **Force adaptation transfers to untrained workspace regions in children: Evidence for developing inverse dynamic models**, Experimental Brain Research, 2002, 143, pp. 212-220.
- <sup>6</sup> Konczak J, Dichgans J, **Goal-directed reaching: development toward stereotypic arm kinematics in the first three years of life**, Experimental Brain Research, 1997, 117, pp. 346-354.
- <sup>7</sup> Trevarthen C, **Neurological development and the growth of psychological functions**. In J Sants (ed), Developmental Psychology and Society, 1980, pp. 46-95
- <sup>8</sup> Meltzoff AN, Moore MK, **Explaining facial imitation: a theoretical model**, Early Development and Parenting, 1997, 6(2), pp. 1-14.
- <sup>9</sup> Bekoff A, Kauer JA, Fulstone A, Summers TR, **Neural control of limb coordination**, Experimental Brain Research, 1989, 74, pp. 609-617.
- <sup>10</sup> Metta G, **Babyrobot: A Study on Sensori-motor Development**, 1999.
- <sup>11</sup> Von Hofsten C, **Predictive reaching for moving objects by human infants**, Journal of Experimental Child Psychology, 1980, 30, pp. 369-382.
- <sup>12</sup> Konczak J, Borutta M, Dichgans J, **Development of goal-directed reaching in infants: II. Learning to produce task-adequate patterns of joint torque**, Experimental Brain Research, 1997, 113, pp.465-474.
- <sup>13</sup> Abeele S, Bock O, **Sensorimotor adaptation to rotated visual input**, Experimental Brain Research, 2001, 140, pp. 407-410.
- <sup>14</sup> Miall RC, Jenkinson N, Kulkarni K, **Adaptation to rotated visual feedback: A re-examination of motor interference**. Experimental Brain Research, 2004, 154, pp. 201-210.
- <sup>15</sup> Krakauer JW, Ghilardi M, Ghez C, **Independent learning of internal models for kinematic and dynamic control of reaching**, Nature Neuroscience, 1999, 2, pp. 1026-1031.

- 
- <sup>16</sup> Scheidt RA, Reinkensmeyer DJ, Conditt MA, Rymer WZ, Mussa-Ivaldi FA, **Persistence of motor adaptation during constrained, multi-joint, arm movements**, *Journal of Neurophysiology*, 2000, 84, pp. 853–862.
- <sup>17</sup> Shadmehr R, Holcomb HH, **Neural correlates of motor memory consolidation**, *Science*, 1997, 277, pp. 821–825.
- <sup>18</sup> Conditt MA, Mussa-Ivaldi FA, **The motor system does not learn the dynamics of the arm by rote memorization of past experience**, *J Neurophysiol*, 1997, 78(1), pp. 554–560.
- <sup>19</sup> Franklin DW, Milner TE, **Adaptive control of stiffness to stabilize hand position with large loads**, *Exp Brain Res*, 2003, 152(2), pp. 211–220.
- <sup>20</sup> Osu R, Burdet E, Franklin DW, Milner TE, Kawato M, **Different mechanisms involved in adaptation to stable and unstable dynamics**, *J Neurophysiol*, 2003, 90, pp. 3255–3269.
- <sup>21</sup> Kawato M, **Internal models for motor control and trajectory planning**, *Current Opinion in Neurobiology*, 1999, 9, pp. 718–727.
- <sup>22</sup> Topka H, Koneczak J, Schneider K, Dichgans J, **Analysis of intersegmental dynamics in cerebellar limb ataxia**, *Soc. Neurosci. Abstr.*, 1994, 712.10.
- <sup>23</sup> Bastian AJ, Martin TA, Keating JC, Thach WT, **Cerebellar ataxia: abnormal control of interaction torques across multiple joints**, *J Neurophysiol*, 1996, 76, pp. 492–509.
- <sup>24</sup> Shadmehr R, Mussa-Ivaldi FA, **Adaptive representation of dynamics during learning of motor task**, *J Neurosci*, 1994, 14, pp: 3208–3224.
- <sup>25</sup> Corcos DM, Jaric S, Agarwal GC, Gottlieb G, **Principles for learning single-joint movements**, *Experimental Brain Research*, 1993, 94(3), pp. 499–513.
- <sup>26</sup> Gottlieb GL, **The generation of the efferent command and the importance of joint compliance in fast elbow movements**, *Experimental Brain Research*, 1994, 97(3), pp. 545–550.
- <sup>27</sup> Thoroughman KA, Shadmehr R, **Electromyographic Correlates of Learning an Internal Model of Reaching Movements**, *Journal of Neuroscience*, 1999, 19(19), pp. 8573–8588.
- <sup>28</sup> Bock O, Schneider S, **Sensorimotor adaptation in young and elderly humans**, *Neurosci Biobehav Rev*. 2002, 26(7), pp. 761–767.
- <sup>29</sup> Sabes PN, **The planning and control of reaching movements**, *Curr Opin Neurobiol*, 2000, 10(6), pp. 740–746.
- <sup>30</sup> Wolpert DM, Miall RC, Kawato M, **Internal models in the cerebellum**, *Trends in Cognitive Sciences*, 1998, 2(9), pp. 338–347.
- <sup>31</sup> Beggs WDA, Howarth CI, **Movement Control in a Repetitive Motor Task**, *Nature*, 1970, 225, pp. 752 – 753.
- <sup>32</sup> Crossman ER, Goodeve PJ, **Feedback control of hand-movement and Fitts' Law**, *Q J Exp Psychol A*, 1983 35(2), pp. 251–278.
- <sup>33</sup> Plamondon R, **A kinematic theory of rapid human movements. Part I.**, *Biological Cybernetics*, 1995, 72(4), pp. 295–307.

- 
- <sup>34</sup> Elliott D, Binsted G, Heath M, **The control of goal directed limb: correcting errors in trajectory**, Human Movement Science, 1999, 18, pp. 121-136.
- <sup>35</sup> Bock O, Schneider S, Bloomberg JJ, **Conditions for interference versus facilitation during sequential sensorimotor adaptation**, Experimental Brain Research, 2001, 38, pp: 359–365.
- <sup>36</sup> Wigmore V, Tong, C, Flanagan JR, **Visuomotor rotations of varying size and direction compete for a single internal model in motor working memory**, Journal of Experimental Psychology, 2002, 28, pp. 447–457.
- <sup>37</sup> Flash T, **The control of hand equilibrium trajectories in multi-joint arm movements**, Biological Cybernetics, 1987, 57, pp. 257-274.
- <sup>38</sup> Gribble PL, Ostry DJ, Sanguineti V, Laboissiere R, **Are complex control signals required for human arm movement?**, J Neurophysiol, 1998, 79, pp. 1409-1424.
- <sup>39</sup> Gomi H, Kawato M, **Equilibrium-point control hypothesis examined by measured arm-stiffness during multi-joint movement**, Science, 1996, 272, pp. 117-120.
- <sup>40</sup> Morasso PG, Schieppati M, **Can muscle stiffness alone stabilize upright standing?**, J Neurophysiol, 1999, 82, pp. 1622-1626.
- <sup>41</sup> Shadmehr R, **Generalization as a behavioral window to the neural mechanisms of learning internal models**, Hum Mov Sci, 2004, 23, pp. 543-568.
- <sup>42</sup> Thelen E, Corbetta D, Kamm K, Spencer JP, Schneider K, Zernicke RF, **The transition to reaching: mapping intention and intrinsic dynamics**, Child Development, 1993, 64, pp. 1058–1098.
- <sup>43</sup> Desmedt JE, **Size principle of motoneuron recruitment and the calibration of muscle force and speed in man**, In Desmedt, J. E. ed., Motor control mechanisms in health and disease, 1993, New York: Ra-ven Press.
- <sup>44</sup> Plamondon R, **A kinematic theory of rapid human movements. Part II. Movement time and control**, Biological Cybernetics, 1995, 72, pp. 309-320.
- <sup>45</sup> Elliott D, Binsted G, Heath M, **The control of goal-directed limb movements: Correcting errors in the trajectory**, Human Movement Science, 1999, 18, pp. 121-136.
- <sup>46</sup> Caselli P, Conforto S, Schmid M, Accornero N, D'Alessio T, **Difference in sensorimotor adaptation to horizontal and vertical mirror distortions during ballistic arm movements**, Hum Mov Sci, doi:10.1016/j.humov.2005.12.003.
- <sup>47</sup> Abend W, Bizzi E, Morasso P, **Human arm trajectory formation**, Brain, 105, pp. 331-348.
- <sup>48</sup> Morasso P, **Spatial control of arm movements**. Exp Brain Res, 1981, 42, pp. 223-227.
- <sup>49</sup> Hollerbach JM, Flash T, **Dynamic interactions between limb segments during planar arm movement**, Biological Cybernetics, 1982, 44, 1, pp. 67-77.
- <sup>50</sup> Flash T, Hogan N, **The coordination of arm movements: an experimentally confirmed mathematical model**, J Neurosci, 1985, 5, pp. 1688-1703.
- <sup>51</sup> Uno Y, Kawato M, Suzuki R, **Formation and control of optimal trajectory in human multijoint arm movement. Minimum torque-change model**, Biol Cybern, 1989, 61, pp. 89-101.

- 
- <sup>52</sup> Massone LL, Myers JD, **The role of the plant properties in arm trajectory formation: a neural network study**, IEEE Trans. Sys. Man Cyb., 1996, 26, pp. 719-732.
- <sup>53</sup> Miller WT, Sutton R, Werbos P, editors. **Neural Networks for Control**. MIT Press, Cambridge, Massachusetts, 1990.
- <sup>54</sup> Angeline PJ, Saunders GM, Pollack JB, **An evolutionary algorithm that constructs recurrent neural networks**, IEEE Transactions on Neural Networks, 1994, 5(1).
- <sup>55</sup> Williams RJ, **Simple statistical gradient-following algorithms for connectionist reinforcement learning**, Machine Learning, 1992, 8, pp. 229–256.
- <sup>56</sup> Pearlmutter BA, **Learning state space trajectories in recurrent neural networks**, Proceedings of the IJCNN, 1989, 2, pp. 365–372.
- <sup>57</sup> Funahashi K, Nakamura Y, **Approximation of dynamical systems by continuous time recurrent neural networks**, Neural Networks, 1993, 6, pp. 801–806.
- <sup>58</sup> Miller PI, **Recurrent neural networks and motor programs**, In Fifth European Symposium On Artificial Neural Networks, April 1997.
- <sup>59</sup> Psaltis D, Sideris A, Yamamura AA, **A multilayered neural network controller**, IEEE Control Systems Magazine, April 1988.
- <sup>60</sup> Schiffmann WH, Geffers W, **Adaptive control of dynamic systems by back propagation networks**, Neural Networks, 1993, 6, pp. 517–524.
- <sup>61</sup> Narendra KS, Parthasarathy K, **Identification and control of dynamic systems using neural networks**, IEEE Transactions on Neural Networks, 1990, 1, 1, pp. 4–27.
- <sup>62</sup> Arbib M, **The handbook of brain theory and neural network**, 2003.
- <sup>63</sup> Kröse B, Smagt P, **An introduction to neural networks**, 1996.
- <sup>64</sup> Zehr EP, Sale DG, **Reproducibility of ballistic movement**, Medicine & Science in Sports & Exercise 1997, 29(10), pp. 1383-1388.
- <sup>65</sup> Hannaford B, Stark L, **Roles of the triphasic control signal**, 1985, 90(3), pp. 619-634.
- <sup>66</sup> Krilow AM, Rymer WZ, **Role of intrinsic muscle properties in producing smooth movements**, IEEE Transaction on biomedical Engineering, 1997, 44(2), pp. 165-176.
- <sup>67</sup> Maurel W, Thalmann D, Hoffmeyer P, Beylot P, Gingins P, Kalra P, Thalmann MN, **A biomechanical musculoskeletal model of human upper limb for dynamic simulation**, 5<sup>th</sup> IEEE EMBS International Summer School on Biomedical Imaging
- <sup>68</sup> Kapandji IA, **The Physiology of the Joints**, 1970, E & S Livingstone, Edinburgh and London, 2nd edition, Vol. 1.
- <sup>69</sup> Karniel A, Inbar GF, **A model for learning human reaching movements**, Biol Cybern 1997, 77, pp. 173-183.
- <sup>70</sup> Lacquaniti F, Soechting JF, Terzuolo SA, **Path constraints on point-to-point arm movements in three-dimensional space**, 1986, 17, pp. 313-324.

- 
- <sup>71</sup> Drillis R, Contini R, Bluestein M, **Body Segment Parameters; a Survey of Measurement Techniques**, *Artif Limbs*, 1964, 25, pp. 44-66.
- <sup>72</sup> Hill AV, **The heat of shortening and the dynamic constants of muscle**, *Proc. Royal Soc.*, 1938, 126, pp.136-195.
- <sup>73</sup> Winters JM, Starks LW, **Analysis of fundamental human movement pattern through the use of in-depth antagonist muscle models**, *IEEE Transaction on Biomedical Engineering*, 1985, 32, pp. 826-839.
- <sup>74</sup> Scovil CY, Ronsky JL, **Sensitivity of a Hill-based muscle model to perturbations in model parameters**, *Journal of Biomechanics*, 2006, 39(11), pp. 2055-2063.
- <sup>75</sup> Lehman SL, Stark LW, **Three algorithms for interpreting models consisting of ordinary differential equations: sensitivity coefficients, sensitivity functions, global optimization**, *Mathematical Biosciences*, 1982, 62, pp. 107-122.
- <sup>76</sup> Doya K, Kimura H, Kawato M, **Neural Mechanism of learning and control**, *IEEE Control System Magazine*, 2001, 21(4), pp. 42-54.
- <sup>77</sup> Flash T, Sejnowski TJ, **Computational approaches to motor control**, *Curr. Opinion in Neurobiology*, 2001, 11, pp. 655-662.
- <sup>78</sup> Jordan MI, Rumelhart DE, **Forward Models: supervised learning with a distal teacher**, *Cognitive Science*, 1992, 16(3), pp.307-354.
- <sup>79</sup> Atkeson, CG, Hollerbach, J M, **Kinematic features of unrestrained vertical arm movements**, *J. Neurosci.*, 1985, 5, pp. 2318-2330
- <sup>80</sup> Haruno M, Wolpert DM, Kawato M, **Multiple paired forward-inverse models of human motor learning and control**, *Proceedings of the 1998 conference on Advances in neural information processing systems II*, 1999, pp. 31-37.
- <sup>81</sup> Jeannerod M, **The cognitive neuroscience of action**, Oxford, Blackwell, 1997.
- <sup>82</sup> Wolpert DM, Ghahramani Z, Jordan MI, **An internal model for sensorimotor integration**, 1995, 269, 5232, pp. 1880-1882.
- <sup>83</sup> Miall RC, **Motor Control, biological and theoretical**, *The handbook of brain theory and neural networks* (M. Arbib), 2003, pp. 686-689.
- <sup>84</sup> Jordan MI, Wolpert DM, **Computational motor control**, *The Cognitive Neuroscience*, 1999.
- <sup>85</sup> Krakauer JW, Ghilardi MF, Ghez C, **Independent learning of internal models for kinematic and dynamic control of reaching**, *Nature Neuroscience*, 1999, 2, pp. 1026-1031.
- <sup>86</sup> Kawato M, Furukawa K, Suzuki R, **A hierarchical neural-network model for control and learning of voluntary movement**, *Biol. Cybern.*, 1987, 56, pp. 1-17.
- <sup>87</sup> Neilson PD, Neilson MD, O'Dwyer NJ, **Adaptive model theory: applications to disorders of motor control**, *Adv. Psychol.*, 1992, 84, pp. 495-548.
- <sup>88</sup> Schaal S, Atkeson CG, **Constructive Incremental Learning from Only Local Informatio**, *Neural Computation*, 1998, 10, pp. 2047-2084.

- 
- <sup>89</sup> Accornero N, Capozza M, **Controllo Motorio: Rete neurale autoapprendente**. Riv. Neurobiol, 1996, 42, pp. 206-207.
- <sup>90</sup> Purves D, Lichtman J W, **Elimination of Synapses in the Developing Nervous System**, Science, 1980, 210, pp: 153-157.
- <sup>91</sup> Kandel ER, Schwartz JH, Jessel TM, **Principles of Neuroscience**, 1991: Elsevier.
- <sup>92</sup> Shepherd GM, **The synaptic Organization of the Brain**, Oxford University Press, New York, 1974.
- <sup>93</sup> Van Essen DC, Anderson CH, Fellman DJ, **Information processing in the primate visual system**, Science, 1992, 255, pp. 419–423.
- <sup>94</sup> Bossomaier T, Snoad N, **Evolution and modularity in neural networks**, In I. Pitas, editor, Proc. IEEE Workshop on Non-Linear Signal and Image Processing, 1995, pp. 289–292.
- <sup>95</sup> Auda G, Kamel M, **Modular neural networks classifiers: A comparative study**, Journal of Intelligent and Robotic Systems, 1998, 21, pp. 117–129.
- <sup>96</sup> Messier M, Kalaska JF, **Differential effect of task conditions on errors of direction and extent of reaching movements**, Exp Brain Res, 1997, 115, pp. 469-478.
- <sup>97</sup> Georgopoulos AP, Massey JT, **Cognitive spatial-motor process**, Exp Brain Res, 1988, 69, pp. 315-326.
- <sup>98</sup> Messier J, Kalaska JF, **Comparison of variability of initial kinematics and endpoints of reaching movements**, Exp. Brain Res, 1999, 125, pp. 139-152.
- <sup>99</sup> Bushnell EW, **The decline of visually guided reaching during infancy**, Infant Behaviour and Development, 1985, 8, pp. 139-155.
- <sup>100</sup> Bullock D, Grossberg S, **Neural dynamics of planned arm movements: emergent invariants and speed accuracy properties during trajectory formation**, Psychological review, 1988, 95, pp. 49-90.
- <sup>101</sup> Kawato M, **Computational schemes and neural networks models for formation and control of multijoint arm trajectory**, Neural networks for control (WT Miller et al.), pp. 197-228.
- <sup>102</sup> Boessenkool J. J., Nijhof E. J., Erkelens C. J., **A comparison of curvatures of left and right hand movements in a simple pointing task**, Exp. Brain Res., 1998, 120, pp. 369 – 376.
- <sup>103</sup> Lackner JR, Dizio P, **Rapid adaptation to Coriolis force perturbations of arm trajectory**, J Neurophysiol, 1994, 72, pp. 299-313.
- <sup>104</sup> Franklin DW, et al., **Learning the dynamics of the external world: Brain inspired learning of robotic applications**, 2006.
- <sup>105</sup> Gribble PL, Mullin LI, Cothros N, Mattar A: **Role of cocontraction in arm movement accuracy**, J Neurophysiol 2003, 89, pp. 2369-2405
- <sup>106</sup> Atkeson C. G., Hollerbach J. M., **Kinematic features of the unrestrained vertical arm movements**, J. Neurosci., 1985, 5, pp. 2318 – 2330.
- <sup>107</sup> Morasso P, Sanguinetti V, **Computational Neuroscience – A comprehensive approach**, 2003

- 
- <sup>108</sup> Franklin DW, Burdet E, Osu R, Kawato M, Milner TE, **Adaptation to stable and unstable dynamics achieved by combined impedance control and inverse dynamics model**, *J Neurophysiol* , 90, pp. 3270 – 3282.
- <sup>109</sup> Burdet E, Tee KP, Mareels I, Milner TE, Chew CM, Franklin DW, Osu R, Kawato M, **Stability and motor adaptation in human arm movements**, *Biol Cybern* 2006, 77, pp. 173-183.
- <sup>110</sup> Craig A, Hancock K. **Living with spinal cord injury: longitudinal factors, interventions and outcomes**. *Clin Psychol Psychother* 1998;5:102-108.
- <sup>111</sup> Craig A, Moses P, Tran Y, McIsaac P, Kirkup L. **The Effectiveness of a Hands-Free Environmental Control System for the Profoundly Disabled**. *Arch Phys Med Rehabil* 2002;83:1-4.
- <sup>112</sup> Warren M. Grill and Robert F. Kirsch, **Neuroprosthetic Application of Electrical Stimulation**, *Review*, 12(1), pp6-20,2000.
- <sup>113</sup> Bruce H Dobkin, **Strategies for stroke rehabilitation**, *Lancet Neurol* (2004), 3: 528–36.
- <sup>114</sup> Nudo R, Wise B, SiFuentes F, Milliken G. **Neural substrates for the effects of rehabilitative training on motor recovery after ischemic infarct**. *Science* (1996), 272: 1791–94.
- <sup>115</sup> Popovich M.B., Popovic D.B., Sinkjaer T., Stefanovic A., Swirlich L. **Restitution of reaching and grasping promoted by Functional Electrical Therapy**. *Artificial Organs* 2002. 26:271-275
- <sup>116</sup> Teeter JO, Kantor C, Brown, DL. **Resource Guide on Functional Electrical Stimulation for Persons with Spinal Cord Injury or Multiple Sclerosis**, FES Information Center, Case Western Reserve University, 1995.
- <sup>117</sup> S.S. Galen, M.H. Granat, **Study of the Effect of Functional Electrical Stimulation (FES) on walking in children undergoing Botulinum Toxin A therapy**, *Proc. 1st FESnet Conference FESnet2002,Glasgow,UK*.
- <sup>118</sup> Liberson WT, Hotmquest HJ, Dow M. **Functional electrotherapy: stimulation of the peroneal nerve synchronized with the swing phase of the gait of hemiparetic patients**. *Arch Phys Med Rehabil.* (1961), 42:101–105.
- <sup>119</sup> Bogataj U, Gros N, Kljajic M, Acimovic R, Malezic M. **The rehabilitation of gait in patients with hemiplegia: a comparison between conventional therapy and multichannel functional electrical stimulation therapy**. *Phys Ther.* (1995), 75:490 –502.
- <sup>120</sup> Wang RY, Yang YR, Tsai MW, Wang WT, Chan RC. **Effects of functional electric stimulation on upper limb motor function and shoulder range of motion in hemiparetic patients**. *Am J Phys Med Rehabil.* (2002) Apr;81(4):283-90.
- <sup>121</sup> Gritsenko V, Prochazka A, **A functional electric stimulation—assisted exercise therapy system for hemiparetic hand function** *Arch. Phys.Med. and Rehabil.* (2004), 881-885
- <sup>122</sup> Furuse N., Watanabe T., Ohba S., Futami R., Hoshimiya N., Handa Y., **Control-Command Detection for FES using Residual Specific Movements**, *IFESS99, 4th Annual Conference of the International Functional Electrical Stimulation Society, August 23-27 1999, Japan*.

- 
- <sup>123</sup> Giuffrida, J.P.; Crago, P.E., **Reciprocal EMG control of elbow extension by FES**, IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol.9, no.4pp.338-345, Dec 2001.
- <sup>124</sup> Prochazka A, Gauthier M, Wieler M, Kenwell Z. **The bionic glove: an electrical stimulator garment that provides controlled grasp and hand opening in quadriplegia**. Arch Phys Med Rehabil (1997), 608–14.
- <sup>125</sup> Popovic D, Stojanovic A, Pjanovic A, Radosavljevic S, Popovic M, Jovic S, et al. **Clinical evaluation of the bionic glove**. Arch Phys Med Rehabil (1999), 299–304.
- <sup>126</sup> Kurosawa K, Futami R, Watanabe T, Hoshimiya N. **Joint angle control by FES using a feedback error learning controller**. IEEE Trans Neural Syst Rehabil Eng. (2005), Sep. 13 (3):359-71.
- <sup>127</sup> Gomi H. and Kawato M., **Neural network control for a closed-loop system using feedback-error learning**, Neural Networks, vol. 6, pp. 933--946, 1993.
- <sup>128</sup> Bernabucci I., D'Alessio T., Conforto S., Schmid M., **Controlling planar ballistic movements by means of neural system**. X Mediterranean Conf. on Med. and Biol. Eng. and Comp. IFMBE Proceedings, MEDICON 2004 Ischia, Italy (2004)
- <sup>129</sup> Popović M. B., Popović D. B., Tomović R., **Control of Arm Movement: Reaching Synergies for Neuroprosthesis with Life-Like Control**, Journal of Automatic Control, vol. 12:9-15, 2002.
- <sup>130</sup> Goffredo M., Carli M., Conforto S., Bibbo D., Neri A., D'Alessio T., **Evaluation of Skin and Muscular Deformations in a non-rigid motion analysis**, Proceedings ISandT/SPIE's International Symposium on Medical Imaging San Diego, California, USA. (2005)
- <sup>131</sup> Maa W., Maa X., Tsoa S., Panb Z., **A direct approach for subdivision surface fitting from a dense triangle mesh**, Computer-Aided Design 36 (2004) 525–536
- <sup>132</sup> Goffredo M., Carli M., Schmid M., Neri A., **Study of muscular deformation based on surface slope estimation**, Image Processing: Algorithms and Systems V - Electronic Imaging 2006 San Jose, California, USA (2006)
- <sup>133</sup> Blake A. and Isard M., **Active Contours**, (London, U.K.: SpringerVerlag, 1998)
- <sup>134</sup> Kass M., Witkin A., Terzopoulos D.: **Snakes: Active contour models**. Proc. 1st Int. Conf. on Computer Vision (1987) 259–268
- <sup>135</sup> Lee A., webpage www.virtualdub.org, 2002.
- <sup>136</sup> Canny J., **A Computational Approach to Edge Detection**. IEEE Trans PAMI, 1986, 679-698.
- <sup>137</sup> Goffredo M., Schmid M., Conforto S., D'Alessio T., **A neural approach to the tracking of human body silhouette**, IEEE ICSIP Conf., Hubli, Karnataka, India, 2006
- <sup>138</sup> Caselli P, Conforto S, Schmid M, Accornero N, D'Alessio T: **Difference in sensorimotor adaptation to horizontal and vertical mirror distortions during ballistic arm movements**. Hum Mov Sci, doi:10.1016/j.humov.2005.12.003.
- <sup>139</sup> Boessenkool J. J., Nijhof E. J., Erkelens C. J., **A comparison of curvatures of left and right hand movements in a simple pointing task**, Exp. Brain Res., vol. 120, pp. 369 – 376, 1998.