Chapter 2

Human Motor Control

This Chapter is a survey of some relevant topics from the current state of the art in the engineering research of human motor control. It briefly describes the anatomy and physiology of human motor control, and then describes some analytic and engineering tools for modeling the biological motor control system.

2.1. The Biological System

There is a widespread research trying to model and understand the biological motor control system. The motivation and applications of this research come from various disciplines such as medical study, sports research, engineering, and pure intellectual curiosity. In this section, we briefly describe the anatomy and physiology of the biological motor control system from an engineering point of view. The aim of this section is to give a general view for the layman. For a comprehensive overview, the reader is referred to Kandel et al. (1991), Rosenbaum (1991), Keifer and Houk (1994), Kandel et al. (1995), Arbib (1995) and many references in each of these sources.

2.1.1. The muscles and the joints

The muscles convert chemical energy into mechanical energy in order to produce force and movement. Skeletal muscles are connected to the bones by means of
muscle tendons and they produce moments and movements in the joints. The geometry of the skeleton determines a mathematical relation between the angle of each joint and the position of each bone in space. This relation is called forward kinematics. The forward dynamics is the mathematical expression that enables us to calculate the force that the bones exert upon the environment as a function of the angle and angular velocity of each joint, and of the moment that is produced by the muscles in each joint. The kinematics and dynamics relationships are calculated with methods that are well described in the robotics literature (e.g., Asada and Slotin 1986). The relationship between the force in the muscles and the moment in the joint depends on the geometric location of the connection between the tendons and the bones (see, e.g., Giat et al. 1994).

The force is produced in the muscles by means of molecular interaction between proteins (see Murray and Weber 1974). This mechanism is energized by ATP (Adenosine Triphosphate) and enabled by the presence of Calcium. The flow of calcium into the muscles is determined by the electric potential in the muscle, which is controlled by the command from the nervous system.

2.1.2. Muscles, motor neurons, and motor units

The motor neurons are located in the spinal cord and they are the main output of the nervous system (see the schematic description in Figure 2). The motor neuron receives information (directly or by means of inter-neurons) from the central nervous system (CNS), from the spinal cord and from proprioceptive feedback. The motor neuron sends the motor command through its axon to the muscles, by means of action potential. Action potential is an electric signal that lasts about one millisecond and is the basic element of communication in-between nerves and between nerves and muscles. The intensity of the control command is generally measured by means of the rate of these action potentials. The motor neuron sends its command through its axon to a group of muscle cells in a specific muscle, the muscle cells that are activated by the motor neuron and the motor neuron itself are called motor unit. There are different sizes of motor units and they can produce different forces. The muscle consists of a group of motor units and the force in the muscle is the sum of the forces that are generated by the motor units. There are many types of mechanical models
that describe the relation between the neural command and the length, force and contraction velocity of the muscle (see, e.g., Hill 1938, Wilkie 1956, McRuer et al. 1968, Winters and Stark 1987). We will further describe the role of the muscle properties in Chapter 4.

**Figure 2:** The motor neuron and its main inputs and outputs. The output is an axon that carries the efferent control signal to the muscles. The inputs are from the CNS, from other interneurons in the spinal cord and from proprioceptors.

### 2.1.3. Proprioceptors

There are various kinds of proprioceptors that send sensory information to the spinal cord and to the CNS (see Matthews 1988). The muscle spindle senses the length and the extension velocity of the muscle. The Golgi tendon organ senses the force in the muscle, and there are receptors in the joint and on the skin that send further information about the status of the motor system. This information is collected in the spinal cord, and can be used there in order to perform simple reflex and regulate the control in real time. This sensory information is also used by the spinal cord and by the CNS for learning and adaptation. There are mathematical models for the function
of sensory organ and for the transmission of information through parallel pathways (e.g., McRuer et al. 1968, Milgram and Inbar 1976)

2.1.4. The spinal cord

The spinal cord contains the motor neurons that send the motor command and receive the sensory information from the proprioceptors. Therefore, its first role is as a simple feedback reflex loop that regulates the control command. See Figure 2 for a simple schematic description of the motor neuron and its main connections. However, the spinal cord is much more complex. It regulates the force in the muscles and the recruitment of the motor units (see DeLuca and Erim 1994). Beyond the motor units, there are other nervous cells in the spinal cord that can potentially manage other automatic tasks and complex reflexes. For example, Bizzi et al. (1991) suggests that the spinal cord can produce force fields that can be combined in order to perform various motor tasks (see also Mussa-Ivaldi et al. 1994).

The spinal cord is intensively connected to various regions in the CNS through sensory and motor pathways.

2.1.5. The central nervous system

The Central nervous system is a general term that virtually includes all the brain. In this work we do not wish to review all the massive literature about the structure of the CNS, we just mention the basic structures that have an important role in the control of movements. The primary motor cortex contains a detailed map of the muscles in the body and can send a specific command to small groups of muscles. A similar sensory map is located in the primary somatic sensory cortex. This notion of mapping was modeled in artificial neural networks for robotics control; see Ritter et al. (1992). The firing rate of some neurons in the motor cortex can also indicate the direction of the planned movement, see Georgopoulos et al. (1993). The cerebellum also has some internal maps, and it has a role in coordinating and timing of movements. The cerebellum is also hypothesized to contain the inverse and forward models of the musculoskeletal system that are supposed to assist in the generation of the motor command (we will further discuss these notions in the next section). The cerebellum has a regular structure and many computational models for its operation were
suggested (e.g., Marr 1969, Bloedel 1992, Houk et al. 1996, Kettner et al. 1997, Schweighofer et al. 1998ab, and Barto et al. 1999). There are many other areas such as the basal ganglia (see Beiser et al. 1997, for a review of recent models) that have an important role in the execution of human motor control; Figure 3 is a rough sketch that describes the main areas and their interconnections.

The specific role of each part of the CNS in motor control is still under research and although vast literature and information is available, we still do not clearly understand its operation.

Figure 3: A schematic description of the main neural structures that participate in the execution of human motor control. This sketch is based on Kandel et al. 1991, Rosenbaum 1991, and Houk and Wise 1992. It is definitely incomplete, however, it does give a general idea of the main pathways.

2.1.6. Limitations of our technical description

In our description of the biological system, it is sometimes unavoidable using engineering terms. However, it is well known that each generation describes the brain by means of its latest technology, e.g., cooling system, telephone exchange, digital computer, and artificial neural networks. We frequently see in the scientific literature phrases such as: "the nervous system has to calculate the inverse kinematics". It is important to remember that we are not allowed to assume that the biological system
works exactly in the same way our technological instruments do. Nevertheless, this kind of description and modeling is worthwhile for analyzing the possible properties of the biological motor control system, as long as we remember this reservation and do not extrapolate too far.

2.2. From Feedback to Adaptation

A salient feature in any biological system is the ability to adapt to the environment and to its own internal changes. The revolution in the beginning of modern cybernetics was the incorporation of feedback to artificial machines. In this section we describe the main engineering control architecture, that serves as a model for the biological motor control, from simple feedback through adaptive control and to artificial neural network control.

2.2.1. Feedback control

Feedback control is based on using the outcome of the process, or the controlled system, which is usually called the plant, in order to control it. In other words, using the error between the desired output, \( Y_d \), and the actual output, \( Y \), in order to reduce it (see Figure 4). See any basic control textbook, e.g., Franklin et al. (1986).

![Feedback control diagram](image)

**Figure 4: Feedback control**

The analogy of the feedback scheme, to the biological motor control is the following. The plant corresponds to the muscles, the bones and the dynamics of the environment, the feedback corresponds to the output of the sensory systems, and the controller corresponds to the nervous system. For biological examples, see Houk (1961), McRuer et al. (1968), and Inbar (1972a). The control problem is how to design a controller that suits the desired performance. In the linear case we can use the
Laplace transform and describe each block with a transfer function (1) where \( k \) is called the gain, \( z_i \) are the zeros and \( p_i \) are the poles.

\[
H(s) = \frac{\text{OUT}(s)}{\text{IN}(s)} = k \cdot \frac{\prod(s - z_i)}{\prod(s - p_j)}
\]

(1)

Let us denote the transfer functions of the blocks in Figure 4 as follows: \( P \) for the plant, \( C \) for the controller and \( F \) for the feedback (i.e., for the sensory system). In the Laplace domain, we can write the output as a function of the input in Figure 4 in terms of the blocks' transfer functions (the Laplace variable \( s \) is avoided for simplicity):

\[
\frac{Y}{Y_d} = \frac{P \cdot C \cdot F}{P \cdot C \cdot F + 1}
\]

One major advantage of the feedback control scheme is the reduced sensitivity to changes in the parameters of the plant, and to changes in the environment. The sensitivity of system \( H \) to changes in the parameter \( k \) is defined as follows:

\[
S^k_H \equiv \left| \frac{\partial H}{\partial k} \cdot \frac{k}{H} \right|
\]

When the value of the sensitivity function is zero, the system is insensitive to changes in the parameter. Let us look at the system without feedback, where \( H \) is the transfer function, and \( k \) is a gain parameter as in (1). The system in an open loop is, \( H = k \cdot P \), and the sensitivity of the system would be:

\[
S^k_H \equiv \left| \frac{\partial H}{\partial k} \cdot \frac{k}{H} \right| = P \cdot \frac{k}{k \cdot P} = 1
\]

The system with feedback, i.e., in close loop, is \( H = \frac{k \cdot P}{k \cdot P \cdot F + 1} \) and the sensitivity would be:

\[
S^k_H \equiv \left| \frac{\partial H}{\partial k} \cdot \frac{k}{H} \right| = \frac{P \cdot (k \cdot P \cdot F + 1) - k \cdot P^2 \cdot F \cdot k \cdot P \cdot F + 1}{(k \cdot P \cdot F + 1)^2} \cdot \frac{k \cdot P \cdot F + 1}{P} = \frac{1}{k \cdot P \cdot F + 1} < 1
\]

Therefore, when the loop gain, \( k \), is high the sensitivity to changes is low.
There is vast literature on the stability of such systems and on methods to choose a controller when the specifications of the desired performances are given (See for example Kwakernaak and Sivan 1972, and section III in Levine 1996).

The first problem in using this simple feedback to model biological systems occurs when one tries to measure the loop-gain. In the biological system, one finds a very low loop-gain in the order of one, therefore the above reason, for reduced sensitivity to changes in the parameters, is not valid for the biological system. The second problem is the delays in the biological system, which can cause instability and oscillation in such a simple feedback control. Therefore, we can conclude that the simple feedback control, despite its advantages, is insufficient to describe the biological motor control. Let us continue with the development of modern control to a more complicated control scheme, the adaptive control.

2.2.2. Adaptive control

Adaptive control can be regarded as an extension of feedback control. In adaptive control, the controller can adapt itself to changes in the plant or even learn to control a new plant. A simple description of adaptive control scheme is given in Figure 5. In the case where the controller is changing, the linear description is not valid even if the plant and the controller are linear. The mathematical questions in such a control scheme are generally related to the ability to learn the plant and to the convergence properties of the adaptation algorithm. For a short summary of adaptive control see Astrom (1995b), for textbooks, see Astrom (1995a) and Goodwin and Sin (1984) and for the first attempt to model a biological system with an adaptive control scheme, see Inbar (1972b).

![Diagram](image.png)

**Figure 5: Adaptive control**
The literature separates between direct and indirect adaptive control. In the first case, the adaptation algorithm operates on the controller and in the second case there are two phases: first identification of the plant and then adaptation of the controller. The second case raises the problem of identification or learning of the plant and if we have a parametric group of possible plants, it brings us to the problem of parameter estimation, which is the subject of the next section. A difficult problem with adaptive controllers is the requirement for persistent excitation of the plant, which is needed in order to continuously estimate its time varying parameters (for conditions on the inputs that ensure persistence excitation, see, Shimkin and Feuer 1987). Without excitation, the estimated value of the parameters tends to drift. This can be stopped in engineering systems by turning off the adaptation algorithm when the system operates with fixed control signals.

2.2.3. Feed forward control and the inverse controller problem

The opposite of feedback or closed loop control is the feed-forward or open loop control where the sensory information is not used during the control and execution of the task. The delays in the biological system are relatively large and in such cases a feed-forward control model must be considered. This notion of feed-forward adaptive control was suggested as a model for the biological motor control by Inbar and Yafe (1976) where the term 'Signal Adaptation' was used to describe the act of learning the inverse controller in order to generate the desired control signals. A similar notion was introduced by Albus (1975) where a model for manipulator control with a lookup table was suggested. Widrow et al. (1979) proposed a method to identify the inverse of linear systems. Figure 6 describes the controller and the plant in a feed-forward control. The control problem is: Given a desired goal $Y_d$, how to generate $U$, such that $F(U)$ will be close to $Y_d$?

![Figure 6: Feed-forward control and the inverse controller problem](image)
This is a simple inverse problem, but the solutions to inverse problems, in many cases, is not simple, since the problem may be ill posed as defined in the next sub-section. Another problem is to learn the inverse of the system. Most biological systems are time-varying systems and therefore the inverse of the system must be learned from examples and adjusted to changes in the system as they occur. The field of artificial neural networks has grown rapidly in the last decade, and it provides many candidate solutions to this problem. Some of these solutions are described after the next subsection, which defines the notion of well and ill posed problems.

2.2.4. Well and ill posed problems

This subsection is dedicated to the definition of a well or an ill posed problem. The data and the possible solutions for the problem are all in a closed subspace in a linear normed space. (For the detailed definition and for further information, see, e.g., Lavrentev and Savelev 1995). A well-posed problem should satisfy the following conditions:

(i) The solution of the problem exists for any data.

(ii) The solution is unique.

(iii) Infinitesimal variations of the data produce infinitesimal variations in the solution.

An ill-posed problem is a problem that does not obey one (or more) of the conditions above.

The inverse of a many-to-one function is not unique and therefore this problem is ill posed. We define a redundant system as a many-to-one function and therefore finding the inverse controller of a redundant system is an ill-posed problem. This problem and a proposed solution are discussed in chapters 5 and 6 of this thesis.

2.2.5. Artificial neural network control

Artificial neural network (ANN) is a field that draws its ideas from the structure of biological neural networks. For a comprehensive textbook about ANN see Haykin (1999), and for a review on the use of ANN for control see Hunt et al. (1992),
Narendra (1995), Levin and Narendra (1996), and a collection of articles in Miller et al. (1990). One can look at an ANN as a black box that operates as a function approximation, with a learning algorithm that can change the structure of the ANN in order to reduce some error between the network output and the desired one, given by training examples. Three results in the study of ANN paved the way for implementing this architecture in the control area: (i) The introduction of the Backpropagation algorithm, which is a gradient based learning algorithm, by Rumelhart et al. (1986) enabled the use of multi-layered ANN. (Actually, Werbos developed a similar algorithm in the 1970's, see Werbos 1989, however this algorithm became popular only after the work of Rumelhart et al. 1986). (ii) The representation theorem shows the ability of a one-hidden-layer ANN to approximate continuous functions. This property was presented and proved by numerous researchers (e.g. Cybenko 1989, Funahashi 1989, Homik et al. 1989). (iii) The ability of a two-hidden-layers ANN to approximate an inverse of continuous functions, which was proved by Sontag (1992). The first result establishes a learning algorithm and the second two establish the capability of the ANN. Unfortunately, there is no guarantee that the backpropagation algorithm will converge to the best approximation. However there are many improved versions that can practically achieve any desired approximation, see one of the textbooks and references above.

Let us describe the use of ANN as a controller in order to solve the feed-forward control problem in Figure 6. The simplest solution is to train an ANN to become an inverse of the Plant, see for example Levin et al. (1991). This solution is shown in Figure 7, where the training phase is described.

![Figure 7: Direct learning of an inverse model with ANN](image)

After the training is done, one can put the ANN before the Plant and it will function as a controller. Such a control scheme has a few drawbacks: First, there is a need for
two phases and therefore two copies of the ANN, one connected as a learner and the other as a controller. There is no evidence in the biological nervous system for such a dual system, therefore other solutions must be considered. However, lack of existence in the biological system is not the only drawback of the direct learning scheme. One problem in this type of learning is that we minimize the error in the control space rather than in the output space. This problem is described in detail and analyzed in section 6.2 of this dissertation. Another problem appears in trying to learn the inverse of a redundant system, i.e., the problem of mapping a many-to-one system. Most of the learning algorithm will converge to the average of all the possible solutions, but the average of correct solutions is not always a correct solution (see Jordan 1996). A second problem in inverting a many-to-one system is how to represent all the solutions and which solution to choose. One major problem in trying to train a controller that is attached to the controlled system is how to transform the error from the output coordinates of the system to the coordinates of the control signal. Following are two solutions to the training problem that were proposed as models for the biological motor control system. The first is the distal teacher in Figure 8 that was proposed by Jordan and Rumelhart (1992) and the second is the feedback learning in Figure 9 that was proposed by Kawato et al., see Kawato et al. (1987), and Kawato and Gomi (1992).

![Image of a block diagram for ANN control with distal teacher.](image)

**Figure 8: ANN control with distal teacher.**

In the distal teacher approach (see Figure 8) the problem of transforming the error from the output of the process to the output of the controller, i.e. the output of the ANN, is solved by an ANN forward model of the plant and by using the learning
algorithm in order to propagate the error through the model to the ANN controller. The forward model is trained by the prediction error and the controller is trained by the performance error that is propagated through the forward model. In the feedback learning scheme (see Figure 9) the error is transformed by a simple feedback controller and the ANN is trained by the motor error. In this way, the control is reasonable even in case of big changes in the plant, since in such a case, there is an error, and then the feedback controller adds its contribution to the control signal in the correct direction.

![Figure 9: ANN feedback learning control scheme](image)

Kawato et al. (1992) showed that in some simple cases, the distal teacher approach converges as the steepest decent method, and the feedback error method converges like Newton's method. This analysis is correct only if the controller has the correct parameters. However, for nonlinear systems, the ANN has more capability for approximating the system's gradient than a conventional feedback controller has to approximate the system's Jacobian. Another method to learn an inverse controller includes indirect learning, where the forward model is learned first and then the ANN is inverted, for examples of methods to invert ANN, see Lee and Oh (1997), Lu et al. (1997), Behera et al. (1996), and Miao and Hua (1995). There are also other architectures and approaches to describe the biological motor control by means of ANN. Stroeve (1998) and Bhushan and Shadmehr (1999) examine the integration of feedback and feed-forward control. Grossberg and colleagues suggest another approach and other architectures, adaptive resonance theory, vector-integration-to-endpoint model, and the Factorization-of-length-and-tension model, which describe the operation of the nervous system for both perception and motor tasks (see, Bullock and Grossberg 1991, Bullock et al. 1993, and Grossberg 1995, 1999). Yet another
approach is in learning sensory and motor maps, by means of an ANN architecture that is called self-organizing map, see Ritter et al. (1992). Which architecture is better, and which is more biologically plausible are still open questions.

2.3. Parameter Estimation

In many control schemes and biological modeling it is desirable to construct a model of the controlled system, for examples, see McRuer et al. (1968), Allin and Inbar (1986), Wolpert and Kawato (1998). A few examples for such a need were demonstrated in the previous section in the context of adaptive control schemes. A Parametric model is a model that belongs to a family of models with a finite number of parameters. The modeler's task is first to choose a proper family of models and then to estimate the values of the parameters. In this section, we describe the estimation problem, and then we concentrate on a commonly used family of models, which is the family of linear models. We give an example for building a linear model of the muscle and estimating its parameters. Then we discuss the problem of choosing the order of the model. The examples are of linear systems. However, the general ideas and issues of concern are similar to nonlinear models.

The purpose of this section is to introduce the basic terminology and problems in the simplest framework of linear systems. The same terminology and problems appear also in complex ANN learning algorithms since ANN can be viewed as a parametric model, and its learning algorithm is equivalent to the classical parameters estimation procedure.

2.3.1. The estimation problem

The general problem of parameters estimation can be formalized as followed: Let $\Theta(x, a)$ be a family of parametric functions, that is, for each parameter vector $a_0$, $y = \Theta(x, a_0)$ is a static input/output function or a transfer function in the Laplace domain, where $x$ is the input and $y$ is the output. Suppose that we have an unknown system $F(x)$ that is assumed to belong to the above functions family, that is, $F(x) = \Theta(x, a_0)$ for a specific but unknown parameter vector $a_0$. As a result of an experiment on this unknown system, we acquired a group of measurements of
inputs/outputs pairs \( \{x_i, y_i\} \) that naturally satisfies \( y_i = F(x_i) \). In the presence of measurement noise or uncertainty in the generating function, that is, if we are not positive about the assumption that the unknown system belongs to the family of parametric functions, we can relax the requirements from the data to be 
\[ |y_i - F(x_i)| < n, \]
where \( n \) represents the noise or the uncertainty in the fitness of the model to the system. The problem is to find the vector of parameters \( a \) that will best fit the measurement pairs, according to a given criterion. If one uses the least-squares criterion, the problem is to solve the following minimization:

\[
\hat{a} = \arg \min_a \sum_i (y_i - \Theta(x_i, a))^2
\]

There are many methods to solve this problem and to formalize parametric groups of functions, for a recent review, see Sjoberg et al (1995). In the next section, we will concentrate on the linear group of functions and on an example from the field of human motor control.

2.3.2. Linear models

A linear model can be described by a linear differential equation, and therefore can be transformed to the Laplace domain as a transfer function in the form of (1). In the discrete case, the same can be done with a difference equation and the Z transform. Since this formalization is easy to analyze, there are many names for all kinds of such systems. If there are only poles, the system is called auto recursive (AR), if there are only zeros, the system is called moving average (MA), and the general case is called an auto recursive moving average (ARMA) system. For a comprehensive description of adaptive control, see Goodwin and Sin (1984). For an example of parameter identification of a discrete non-linear muscle dynamics, see Inbar et al. (1970).

2.3.3. An example: parameters estimation of a linear muscle model

Let us look at the mechanical second order model in Figure 10. This is a linear lumped model, which is an approximation of the muscle behavior for a small signal (see McRuer et al. 1968). In this model, \( P \) represents the internal force in the muscle.
that is the result of the neural excitation. $K$ and $B$ are the elastic and the viscose-damping elements that represent the mechanical properties of the muscle tissue, and $M$ is the mass of the muscles and the joint.

![Second order mechanical model of the muscles and the joint.]

**Figure 10:** Second order mechanical model of the muscles and the joint.

The output to input transfer function of this model is:

$$\frac{X(s)}{P(s)} = \frac{-1}{M \cdot s^2 + B \cdot s + K}$$

One can derive a similar relation for external force and its relation to the position, or any other desired relationship, for the muscle model. Our goal is to estimate the value of the model parameters ($M$, $B$ and $K$ in this example). Since we usually use a computer and discrete measurements, our first step will be discretization. There are different procedures for discretization just as there are many ways for numerical integration. Let us use here the simplest method, the Euler’s forward method, which is to move to the $Z$ transform by replacing each $s$ by $(z-1)/T$, where $z$ is the $Z$ transform variable and $T$ is the sampling interval,

$$\frac{X(z)}{P(z)} = \frac{-1}{M \cdot z^2 + B \cdot z + K} \bigg|_{s = \frac{z-1}{T}} = -\frac{T^2}{M \cdot z^2 + (B \cdot T - 2 \cdot M) \cdot z + M - B \cdot T + K \cdot T^2},$$

see Santina et al. (1996) for more details about discretization methods.

From the $Z$ transform we can move directly to discrete time:

$$X(n) = -\frac{T^2}{M} \cdot P(n-2) - \frac{(B \cdot T - 2 \cdot M)}{M} \cdot X(n-1) - \frac{(M - B \cdot T + K \cdot T^2)}{M} \cdot X(n-2),$$

this equation can be formulated as follows:
\[ X(n) = w_1 \cdot P(n-2) + w_2 \cdot X(n-1) + w_3 \cdot X(n-2), \]

where \( w_i \) are the parameters. If the sampling interval \( T \) is given, it is equivalent to know \( w_i \) or \( M, B, K \). This functional description is illustrated in Figure 11.

\[ \begin{aligned}
&\text{\( P(n-2) \)} \\
&\text{\( X(n-1) \)} \\
&\text{\( X(n-2) \)} \\
&\downarrow W_3 \quad \downarrow W_2 \quad \downarrow W_1 \\
&\sum \quad \text{\( \vdots \)} \\
&\text{\( X(n) \)} \\
\end{aligned} \]

**Figure 11**: The length of the muscle as a linear function of the excitation and the previous lengths.

The description in Figure 11 reminds us of a very large family of parametric models which can be presented by an Artificial Neural Networks (ANN). We use this description for two reasons, first since these models were inspired by the structure of the biological nervous system, and second since this description prepares the ground for handling much more complex nonlinear models. Let us leave this general case and continue with our special case which is the linear model, we have presented our model as a weighted sum of the inputs and our problem is to find the optimal weights. We can combine our input components \( X(n-2), X(n-1), P(n-2) \) to form an input vector \( U \), and denote the output vector, which has in our case just one element, \( X(n) \), by the letter \( Y \). Now, if we use the Minimum Mean Square Error (MMSE) criterion, we can use the optimal solution, which is:

\[
W_{opt} = \Phi^{-1} \cdot P \quad \Phi = \mathbb{E}[Y \cdot U] \quad \Phi = \mathbb{E}[U \cdot U^T]
\]

Where \( \mathbb{E} \) stands for expectation, see Porat (1994). In practice, numerical average over the measurements is used.

Let us illustrate this estimation scheme with a simulation example. A random sequence of inputs \( P \) was drawn from a normal distributed noise with standard deviation (STD) equal one, and zero mean. The length \( X \) was calculated with the following nominal value of the parameters: \( M = 5, B = 3, K = 2, T = 0.1 \), that is
\( W_1 = -0.002, \ W_2 = 1.94, \ W_3 = -0.944 \). Figure 12 shows the results of the simulation. The first graph is the random input \( P \), the second is the calculated \( X \). An additional random noise was added to simulate measurement noise or uncertainties in the model (Normal distributed noise with STD=0.01 and zero mean), this sequence appears in the third graph. Then the optimal MMSE parameters were calculated and the results were: \( W_1 = -0.0035, \ W_2 = 2.088, \ W_3 = -1.09 \) which are close to the nominal parameters, as expected. Finally, the output of the estimated model was simulated and it appears in the forth graph to be similar to the second graph, which is the actual model output.

![Graphs of simulation results](image)

**Figure 12:** An example of parameter estimation of the linear (ARMA) muscle model.

This example is synthetic and has many drawbacks that will help us illustrating the problems in parameter estimation procedures. This example regards the discrete data as a set of independent examples of a static model and the optimal model is checked for each couple of input-output independently. In practice, the error combines from one time step to the other, since the model may use its own output to estimate the next
time step, and not the real system outputs. This problem can be severe when the system has some unstable poles, then the error might grow very fast. The calculated model, that is, the estimated parameters should be checked on a new data set and not only on the data that was used for the parameter estimation. This check is called generalization check and it can assist in avoiding over-fitting the data. We discuss this method of validation in the following subsection. One should remember that the biological system is generally a time varying system, for example the muscles can change their properties due to fatigue, therefore the duration of the experiment must be short in order to justify the assumption that the system is a time invariant system. We must mention here that the simple optimal MMSE parameter calculation above is not always stable numerically and there are many improvements and practical methods that can be found in modern numerical software, such as MATLAB (see Ljung 1994).

2.3.4. The order of the model

In the example above, the structure of the model was known and the only problem was to estimate the parameters, but in most biological cases and in many other applications the model is unknown. For the sake of simplicity and since there are many algorithms and software tools for linear systems; we restrict our discussion here to linear model. The general structure of a linear model is the ARMA model, which can be written in the discrete form as follows:

\[ y(n) = \sum_{i=0}^{N} a_i \cdot x(n-i) - \sum_{j=1}^{M} b_j \cdot y(n-j) \]

As an immediate example of an ARMA model, we can mention the example in the previous subsection.

In order to estimate the parameters we need to establish first the order of the model. In the ARMA case, we have to choose \( N \) and \( M \) in the equation above. At a first look, one can suggest that the more parameters the model has, the better it will fit the actual system, but actually, this is not the case. Too many parameters are not only a computational burden but they may cause errors in the model (see Paiss and Inbar 1987 for extensive treatment of the model order selection problem for the case of
surface electromyography). Let us describe the pitfalls in choosing the wrong number of parameters. One can be wrong by either choosing too many or too few parameters.

*Under-fit* is the situation where the model is less complex than the actual system; in this case, the model is unable to fit the data. See Figure 13 on the left.

*Over-fit* is the situation where the model is more complex than the actual system, in this case, the model will fit the observations, but if there is a noise or insufficient observations (i.e., less independent observations than the number of parameters) the model will not fit the actual system, and in the validation process it may fail to predict the outcome of the system. (In the validation process, we check the generalization, that is, the ability of the identified model to deal with cases that were not seen before). See Figure 13 on the right.

![Figure 13: Fitting a model to data. In this illustration, the three stars are the data taken from an underlying unknown function. In the left a linear function was fitted to the data, in the middle quadratic function was fitted and in the right a third order polynomial function was fitted. After the fitting was completed, two more examples were taken from the same underlying function (the two circles). One can see that the left model is too simple, i.e. under-fit the data, and the right model is too complex, i.e. over-fit the data but unfortunately, does not fit the underlying system.](image)

Many approaches have been suggested for choosing the proper order. For linear models, a commonly used approach is the Akaike information criterion (AIC), which is based on a discrepancy measure. For the ARMA model, it will take the following form:
\[ N^{-1} \cdot AIC(n,m) = \hat{\sigma}^2_x + \frac{2 \cdot (n + m + 1)}{N} \]

Since the first term, the estimation of the error, \( \hat{\sigma}^2_x \), is monotonically decreasing with increasing model size, and the second term is increasing, one can find an optimal model size by finding the minimal value of the AIC. Another method to choose the order of the model is by empirical validation. This method is commonly used in pattern recognition and classification where part of the data is kept from the learning phase (in our case this will be the fitting phase) and then the model is chosen for its generalization capabilities checked on the kept data. For more information about parameters estimation, and system identification see Porat (1994), Sjoberg et al. (1995), Ljung (1994,1996).

2.4. Redundancy, Degeneracy and Parallelism

Redundancy, parallelism and degeneracy have a major role in the superb flexibility, reliability and robustness of the biological system. However, these properties frequently jeopardize our attempts to model and understand this system. In this dissertation, we define redundancy in a functional view, as the ability to perform the same task in more than one possible way, see also Latash and Turvey (1996), and Zatsiorsky et al (1998). Degeneracy is defined in a structural view as the ability of elements that are structurally different to perform the same function. Classical illustration of degeneracy includes different proteins that catalyze the same enzymatic reaction, and different genotypes that produce the same phenotype (see Edelman 1987). Redundancy in a structural view is sometimes defined as the situation where the same function is performed by identical elements (see Tononi et al 1999, for recent attempt to measure these properties with tools from information theory). Both structurally degenerate and structurally redundant systems are functionally redundant, and in this dissertation, we will always refer to systems in the functional sense and use just the term redundant. Parallelism relates to the architecture of the biological system, which includes frequently, many pathways that execute similar or related task. For example, nerve fiber contain many axons that transport similar information, a muscle contains many parallel similar fibers, there are many muscle spindles and Golgi tendon organs in each muscle that measure length, speed and force in the muscles, and so forth. In many cases the parallel architecture implies redundancy, but
redundancy can exist without parallelism, as is in the inverse kinematics problem. Parallelism can also exist without redundancy, as is in some distributed systems, see for example in multiple model control (Wolpert and Kawato 1998, and Narendra and Balakrishnan 1997). In the following subsections, we describe two interesting aspects of these properties. The issue of multiple feedback loops which is most common in biological systems and can jeopardize classical attempts to measure the loop-gain, and one possible function of the parallelism in nerve fibers, which is the spatial filtering. The proposed model for learning to control redundant system is described in the next chapters.

2.4.1. Multiple feedback loops

In section 2.2.1, where feedback control was discussed (see Figure 4), the importance of the loop-gain in reducing the sensitivity to parameter changes was mentioned. The loop-gain can also be a major factor in establishing the stability of the system. In order to measure the loop-gain, one should break the loop, introduce input in one place and measure the output. However, in the biological system there are typically multiple feedback loops in parallel. Figure 14 describes a typical situation of many sensory systems that work together for the same purpose. For example in the temperature regulation system, there are sensors in the skin, in the core of the body and in the hypothalamus and they all influence the temperature regulation mechanisms (see Brown and Brengelmann 1970). In movement control, there are feedback loops from sensors in the muscles, joints, and skin, i.e. muscle spindles, Golgi tendon organs, joint pressure transducers etc. and there are many of each type of sensor all operating in parallel. The first advantage of such multiple loops and of any redundancy is in robustness, i.e. if one sub-system fails, there are other options to activate the system. More advantages that are interesting are spatial filtering and the flexibility given to the system in the ability to choose different solution in different situations, as will be described in the following subsections.
Figure 14: Multiple feedback loops (in the motor control system, in addition to these multiple feedback loops, there is excessive nesting of feedback loops and in this case the transfer function of each block in this diagram contains the expressions of these internal loops).

There is a great danger in trying to estimate the loop-gain in such a system, because there may be loops that we cannot open, or are unaware of their existence. In such a case, we may underestimate the loop-gain. For example, if we open the first two loops in Figure 14 and leave \( F_3 \) connected, the transfer function from \( Y_d \) to \( Y \) will be \( P \cdot C/(P \cdot C \cdot F_3 + 1) \) instead of \( P \cdot C \) when there is no additional loop. Therefore, one should be aware of these multiple loops.

2.4.2. Spatial filtering

Another interesting property of parallelism in the biological system is the spatial filtering in a multi-path transmission mechanism. Milgram and Inbar (1976) demonstrate that the fidelity of transmission of certain types of information through a single neurons acting alone is comparatively poor, within typical physiological ranges of operation, and that the quality of reception of these messages increases in proportion to the number of neural channels involved in transmitting them simultaneously, or in parallel. The main idea behind these observations is that there is a distortion in the neuronal code, which adds high frequencies to the transmitted information. The parallel transmission produces low pass filtering (which in this case is spatial filtering) and therefore improves the transmission fidelity by eliminating the distortion components. In order to explain this spatial filtering, let us examine a multi-channel transmission line, where each channel has different delay. The input to all the channels is \( f(t) \) and the output will be:
\[ P_o(t) = \sum_{k=0}^{M} f(t - a_k). \]

If one assumes a large number of channels, and the distribution of the delays \( a_k \) is given as a density function \( g(a) \), the output can be estimated as the following integral:

\[ P_o(t) \approx M \cdot \int_0^\infty f(t - a_b) \cdot g(a) \cdot da. \]

Let us use the Laplace transform in both sides and then remember that the input is \( M \) times \( f(t) \) which is \( Mf(t) \) in the Laplace domain:

\[ P_o(s) \approx M \cdot \int_0^\infty F(s) \cdot e^{-st} \cdot a \cdot g(a) \cdot da = P_i(s) \cdot \int_0^\infty e^{-st} \cdot a \cdot g(a) \cdot da. \]

Now, one can see that the transfer function between the input and the output is the Laplace transform of the delay distribution:

\[ \frac{P_o(s)}{P_i(s)} \approx \int_0^\infty e^{-st} \cdot a \cdot g(a) \cdot da = G(s). \]

This result means that as the range of delays is larger, the filter is narrower. In the extreme case when all the channels are equal, there is no filtering and there is no advantage over a single transmission line.

There are other possible ways to achieve similar spatial filtering, such as distributed thresholds instead of delays, the latter option is more physiological plausible, but is less analyzable, and therefore is demonstrated in simulations (see Milgram and Inbar 1976).

### 2.4.3. Learning to exploit the redundancy

Learning and redundancy are the main issues of this dissertation. In this chapter, we have introduced the background and the salient research directions in the field of human motor control. The fact that it is an adaptive system with vast redundancy is well recognized. The main theme of this thesis is that the redundancy should be viewed as a virtue rather than a problem and that the biological system can exploit the
redundancy and use different solutions in different circumstances. When a single solution is desired frequently the long-term adaptation, which is the evolutionary process differs the responsibility for choosing this solution to the dynamics of the system in the lower level, that is, the muscles and the spinal cord dynamics. However even in these cases and definitely in many other cases the CNS can choose out of many possible solutions and multiple controller architecture is suggested here for this purpose. These subjects are introduced in the next chapter and then demonstrated and analyzed in the rest of this thesis. We defer the discussion of other related recent research to Chapter 7 where we discuss it in view of the current thesis results.
Chapter 3

The General Model

This chapter introduces the general model for learning redundant systems control. The control problem is described, a hierarchy of learning and adaptation is introduced, and formal definitions of redundancy types and of multiple controller are given. Finally, the general model is illustrated.

3.1. The Control Problem

Suppose that we are control engineers that are occupied in the classical task of designing a controller for a given system. We call the given system a plant and our job is to bring it to behave in a desired fashion. We typically build a model of the plant, find means to change its behavior and call them inputs and find measurements of the plant behavior and call them outputs. Then the control problem is "how to design a controller that will produce the proper control signals to the input of the plant, such that the output of the plant will indicate the desired behavior".

As engineers, that are trying to build a model of the biological motor control system, our problem is even greater. We wish to use the terminology and methods of the control engineering trade and impose them on the biological system. However, there is no clear distinction between plant and controller in the biological system, even after we describe it by means of a block diagram, see Figure 15. Many researchers relate
to the joints and muscles as a plant and to the nervous system as a controller. This description can be misleading. The biological system evolved together and all the organs on the motor control system share the responsibility to produce the desired behavior. There are evidences that the structure of the muscles can simplify the control scheme and that means that it is not just a plant. We will supply such evidence and discuss them in Chapter 4.

![Block diagram of the biological motor control](image)

**Figure 15:** A block diagram of the biological motor control. It is not clear where the controller ends and the plant begins.

With this comment in mind, we still use this distinction in order to enjoy the control theory terminology and tools. Once the distinction between the plant and the controller is made, we present the control problem as a feed forward control, as illustrated in Figure 6, the biological reason for preferring feed-forward over feedback is described in section 2.2.

At this point, we are looking for an inverse of the controlled system. In the presence of redundancy, the inverse problem is ill posed (see subsection 2.2.4). The controller has to act on a many-to-one (MTO) system and has to choose one of the many possible actions to obtain the same desired target. Most of the solutions to this problem use a single criterion, such as smoothness of the movement or minimum energy, and find a single optimal solution that optimized the chosen criterion (see, e.g., Jordan 1990). However, the biological system exhibits the ability to use different solutions at different occasions (see, e.g., Latash and Turvey 1996). The nervous system is known to be modular and there is enough room to have many solutions ready, and choose the desired solution in real time. The solution is chosen according with an appropriate criterion that can change under different circumstances. Beside this issue of redundancy, we recall that the biological system is an adaptive one and therefore has to learn and track in time, the environmental parameters and its own internal parameters from examples. An illustration of the proposed model, which deals with both redundancy and learning, is described in Figure 16. In our general
model, we will also address the first issue of the distinction between the plant and the controller and the role of the system dynamics in reducing the control complexity.

![Diagram](image)

**Figure 16:** The proposed scheme for learning the inverse control of redundant system. The plant is a many-to-one system; the controller parameters \( w \) are learned by means of previous input and output examples \( x, y \). The controller is able to choose one of the many possible control signals according with the parameter \( p \), which is selected by a redundancy exploitation system according to a changeable set of criteria.

In the next sections, we organize and clarify the notions and the definitions of various terms regarding adaptation and redundancy and then we illustrate the general model.

### 3.2. The Virtue of Adaptation

Adaptation in the wide sense is any process of change in the system that is aimed to fit itself to the environment. Adaptation is well recognized as one of the salient and most remarkable feature of the biological system, see Holland (1995). Adaptation is definitely a desired property and therefore we call it a virtue. Some examples for the presence of the virtue of adaptation were described in the previous chapter both for biological systems and for artificial systems in various architectures and techniques. The jargon used in this field seems to be mixed and unclear, therefore we suggest the following hierarchy in learning and adaptation terminology in order to clarify the scope of each term and demonstrate the large range of the adaptation phenomenon.
3.2.1. The hierarchy of learning and adaptation

In the following hierarchy, there is a tradeoff between two factors: the speed of adaptation, and the extent of plasticity. Feedback is the most rapid, but its changes do not last beyond the time scale of the process, while evolution is the slowest and its changes last for generations to come.

**Feedback:** A process of change in the control signals during the execution of the action, by means of sensory feedback signal from the plant output. This change is valid only for the current execution and its effects are ‘forgotten’ rapidly. The feedback scheme is used for tracking or regulation and its advantage is in noise rejection and low sensitivity to changes in the systems parameters.

**Adaptation (in the strict sense):** A process of change in the controller parameters by means of sensory information from the plant output, which is valid for the next execution of similar actions. This change is valid for longer period, as long as the plant parameters do not change. The adaptation rate is normally much slower than the time constants of the plant, but much faster than the rate of change in the plant parameters, therefore it may take a few repetitions of the action in order to achieve a significant adaptation and change in the controller parameters. The adaptive control scheme is used where the controlled system is time variant and then the controller can adjust to changes in the system.

**Learning:** A process of change in the control in order to produce new control signals to a new situation or to a new target. This change can involve a major change in the architecture and it may take many repetitions of the execution in order to achieve the desired change. The results can stay permanently in the same human, animal or machine. Learning takes place when major change in the system occurs, when a new task is given, and when a new skill is acquired. In humans, there are types of learning that are strict to specific time windows in the development (e.g. in children).

**Evolution:** A process of change in the architecture or basic property of the control by mutation and natural selection through a long period of trial and error and optimization. This change occurs only when a new species or a new version of the machine is evolved. This change is the result of many trials of many actions and controllers (By survival of the fittest), and it may last for generations.
3.3. The Virtue of Redundancy

The biological motor control system contains vast redundancy in all its hierarchical levels starting from the excess degrees of freedom in the skeletal structure, through the multiple muscles around each joint, the motor units and the multiple neural pathways. This fact is sometime considered as the Bernstein problem (Bernstrin 1967), and in other context even as the curse of dimensionality (see Bellman 1961). We prefer to relate to it as a virtue rather than a problem, as a blessing rather than a curse. Redundancy with a good controller can improve the reliability and flexibility of the system and is probably one of the main reasons for the superb dexterity of human motor control (see Bernstein 1967, and Latash and Turvey 1996). Before we present our general model, let us write the formal definition of redundancy and the notion of multiple controller which is an important part of the general model.

3.3.1. The formal definition of redundancy

We define redundant systems as being many-to-one functions and suggest a set of definitions for different types of redundancy. These definitions provide a solid ground for discussing learning issues and for suggesting and analyzing new architectures for the control of redundant systems.

Definition-1: A system is defined by a function $f : X \rightarrow Y$.

Remember that a definition of a function includes a definition of a mapping and definitions of input and output domains. In this definition, we do not restrict the input and output domains; they can be scalars, vectors, continuous or discrete functions, or Laplace or Z transform domain functions. Nevertheless, a specific system definition must include a mapping, and well-defined input and output domains.

Definition-2: A system $f : X \rightarrow Y$ is redundant if there exist $y \in Y$, $x_1 \in X$, $x_2 \in X$, such that $x_1 \neq x_2$ and $f(x_1) = f(x_2) = y$.

Thus, a system is redundant if and only if it is not injective (injective being a one-to-one mathematical function). Let us further differentiate between three types of redundancy, finite, countable and uncountable.
**Definition-3:** A system is said to possess *finite redundancy* if it is redundant and if for each $y \in Y$ there is a finite number $N$ and a finite set of input values $S = \{x_1, x_2, \ldots, x_N\}$ such that $f(x_i) = y$ for every $x_i \in S$; and $f(x) \neq f(x_j) = y$ for every $x \not\in S$, $x_j \in S$.

For example, $f(x) = |x|$, where $X = (-\infty, \infty)$, $Y = [0, \infty)$, is a redundant system with finite redundancy (see Figure 17a). A physical example for such a case can be a tap with a handle that enables the water flow when the handle is rotated to the right or to the left.

**Definition-4:** A system is said to possess *countable redundancy* if it is redundant and if for some $y \in Y$ there is a countable set of input values $S = \{x_1, x_2, \ldots\}$ such that $f(x_i) = y$ for every $x_i \in S$; and $f(x) \neq f(x_j) = y$ for every $x \not\in S$, $x_j \in S$.

For example, $f(x) = \sin(x)$ where $X = (-\infty, \infty)$, $Y = [-1, 1]$ is a redundant system with countable redundancy (see Figure 17b). A physical example can be a simple unconstrained rotation joint.

![Figure 17](image-url)

**Figure 17:** Types of redundancy: (a) finite redundancy as $f(x) = |x|$, (b) countable redundancy as $f(x) = \sin(10 \cdot x)$, and (c) uncountable redundancy as $f(x) = 0.5 \cdot \text{step}(x)$.

46
**Definition-5:** A system is said to possess *uncountable redundancy* if it is redundant and if for each \( y \in Y \) there is a set of input values \( S \), such that \( f(x) = y \) for every \( x \in S \); and \( f(x') \neq f(x) = y \) for every \( x' \notin S \), \( x \in S \), and if there is at least one value of output for which the cardinality of \( S \) is greater than countable.

For example, \( f(x) = 7 \) where \( X = (-\infty, \infty) \ Y = 7 \), is an uncountable redundant system (see Figure 17c). A physiological example can be two muscles that act on a single joint. There are uncountable number of combinations of forces in each muscle that will generate the same moment in the joint.

**Remarks**

1. In the literature, the finite and countable redundancies are not always considered as redundant systems (e.g., a manipulator without excess degrees of freedom can posses countable redundancy according to the definitions above, however it may not be considered as a *redundant manipulator*, see for example, DeMers 1998). For linear redundant systems, only uncountable redundancy is possible, and a redundant system is sometimes defined as a system with fewer outputs than inputs (see, e.g., Neilson 1993).

2. There is a crucial importance to the definition of the input and output domains, since different domain can imply different types of systems. For example, the system \( f(x) = \sin(x) \) where \( X = (-\pi, \pi) \ Y = [-1,1] \) is not redundant.

3. Any redundant system is either countable redundant or uncountable redundant, since any set \( S \) is either countable or not countable. (Note that a finite redundant system is also countable redundant.)

4. Note that a non-redundant system that is onto is invertible. The proof is by a theorem, which states that a bijective (=one-to-one and onto) function is invertible.

5. One should notice the difference between structural and functional redundancy. Tononi et al (1999) described the difference between redundancy and degeneracy: redundancy being the result of identical elements in the system and degeneracy of different elements that perform the same function. These definitions are in the
structural sense. The definitions in this thesis are in the functional sense and in this sense both redundancy and degeneracy are functionally redundant.

3.4. The Notion of Multiple Inverse Controller

In this section, we define the notion of multiple inverse controller. The main idea behind the concept of multiple controller is to learn all the possible control signals and choose one of them in real time according to a changeable criterion, see Figure 16.

When the system is redundant, finding an inverse is an ill-posed problem. We suggest to regulate the redundant system by expanding the output space $Y$ to $Y \oplus P$, where $\oplus$ stands for a direct sum (i.e., each element in the expended space consist of one element from $Y$ and one element from $P$). For a finite redundant system, $P$ is a finite set; for a countable redundant system, $P$ is a countable set; and for an uncountable redundant system, $P$ is a vector space. Then we suggest constructing a multiple inverse function $f_p^{MI}(y_d)$, where MI stands for Multiple Inverse and the parameter $p$ determine which of the many possible solutions is chosen. Let us state a formal definition of multiple inverse controller.

**Definition-6:** Let $f : X \to Y$ be a redundant system. The system $f_p^{MI}(y) : Y \oplus P \to X$ is called the *multiple inverse system* (or function or controller) if for every input value $x \in X$, there is a parameter $p \in P$, such that $f_p^{MI}(f(x)) = x$.

In practice, the system $f(x)$ is frequently unknown, and we are given a series of input and output vectors $\{x', y'\}$, which are input/output pairs of the unknown system. In this case the formal requirement is that for any given accuracy value $\varepsilon$, one can construct $\hat{f}_p^{MI}$, such that for any value of $y_d$, and for any value of the parameter $p$, the following inequality will hold.

$$|f(\hat{f}_p^{MI}(y_d)) - y_d| < \varepsilon$$

In order to construct a good approximation for multiple inverse controller, this requirement should be coupled with a requirement for completeness, i.e., that all the solutions are achievable. This issue will be described further in chapter 6.
In Chapter 5 we propose a new architecture, the polyhedral mixture of linear experts (PMLE). This architecture is based on the mixture-of-experts architecture of Jacobs et al (1991) and the hinging-hyperplanes algorithm of Breiman (1993). It produces a piecewise linear approximation of the system. Each area is governed by a linear function, which is called an expert, and one can invert each expert and get the multiple inverse. We then show that this architecture is capable to approximate inverse functions and to serve as a multiple inverse controller.

3.5. The General Model

Two competing views in the literature of motor control are the "dynamical" and the "hierarchical" view. The "dynamical" view asserts that the dynamics of the system finds the solution and therefore reduces the redundancy and simplifies the control scheme; the major advocate for this view is the equilibrium point hypothesis (EPH). The other view that is sometimes called "hierarchical", suggests that the CNS is aware of the details of the controlled system and calculates proper control signals, for example by means of an inverse model of the controlled system. We do not choose one of these views because we think that they both exist at different levels of the control (see also Latash and Anson 1996, in response to a commentary). Our general model contains both of these aspects in different levels.

A comprehensive model should permit the following three methods to exploit redundancy. (i) A simple fixed optimization that is dictated by the dynamics of the mechanical and neural systems, for example the size principle of motor unit recruitment and the smooth speed profile of reaching movements. (ii) Stochastic selection, that is an arbitrary variability, as demonstrated in the experiment of Bernstein with the blacksmith and his hammer (see e.g. Latash and Turvey 1996). Latash (1996) described this idea where he suggests, in a paraphrase on Einstein's famous quote, that the brain does play dice in choosing the motor commands. This method can be advantageously used in distributing the effort and stress evenly between joints and muscles. (iii) A systematic registration of all the solutions and a mechanism to choose the optimal one given a specific control.

Figure 18 is an illustration of our general model that includes these methods and their speculated place in the biological system. The first method, i.e., fixed optimization,
can be the result of the muscles' dynamics and of the spinal cord (SC) dynamics. We relate to the mechanism by which the dynamics of the system determine the single solution by the acronym DDSS. The third method can be implemented by a multiple controller (MC) such as the PMLE, which can be supplied with the redundancy reduction parameter from two possible ways, a specific criterion or a stochastic input. A stochastic input in this case can account for the second method above.

![Diagram](image)

*Figure 18: An illustration of a biologically plausible general model for learning redundant system control.*

To be more concrete, let us think about the control of rapid movements and see how it fits into our general model. Rapid movements have been intensively studies and were found to be generally performed according to a set of stereotyped features. See for examples, Robinson (1964) for invariant features of saccadic eye movements, and Flash and Hogan (1985) for the speed profile of reaching movements. Invariant
features of the speed profile were also found in the octopus arm movements, see Gutfreund et al. (1996). We suggest that these stereotyped features are the result of the dynamics of the lower level, i.e., the result of the DDSS part of our general model. We suggest that the CNS has to determine just a simple set of parameters and the trajectory of the movement is determined by the dynamics of the lower level. We will supply evidence for this view in chapter 4. However, the redundancy is still there even after this stage and one can choose many possible ways to perform the same task by means of these stereotyped rapid movements. For examples, one can look at the a desired target by mean of eye saccade, head movement or body movement, complex movements can be a combination of many possible sets of stereotyped arm reaching movements, and the octopus can reach for his target by means of many possible arms. We suggest that the motor control system can learn to use many possible ways to perform the same task, and to use different solutions in different circumstances, this is the idea of the MC in the higher level of our general model. We suggest a new architecture and theoretical framework for the purpose of analyzing the performance of MC in Chapters 5 and 6 of this thesis.

In our general model, we added direct arrows (dotted arrows in Figure 18) from the desired target block to the DDSS block and to the musculoskeletal block. These arrows represent the direct pathways from the motor cortex to specific muscles and to specific neural pools in the spinal cord. They can be used in order to bypass the MC for rapid execution of stereotyped simple movements by means of the DDSS or for specific concisely activation of specific group of muscles. The other path is through the MC, which can choose out of the many possible solutions to achieve the task. The MC can send a general command to the DDSS, e.g., parameters of excitation or of stereotypical movement, or alternatively a specific command to the musculoskeletal system. The distinctions made in this model are not strict, we are not positive as to location of each mechanism. It is possible that the spinal cord contains multiple controllers and alternatively that part of the CNS operates according with the DDSS notion. Nevertheless we believe that the proposed general model and the introduction of this twofold mechanism of DDSS and MC will prove to be useful in future development of the human motor control modeling research.
The next chapter describes the lower part of the model and then chapters 5 and 6 present a new architecture for a multiple controller and discuss some learning issues. The reader that is not interested in the dynamic part and only interested in the multiple controller and the computation aspects can skip chapter 4 and go ahead to chapter 5 without any loss in continuity.
Chapter 4

System Dynamics

This chapter describes and illustrates the role of the mechanical system, the muscles and the spinal cord dynamics in reducing the redundancy and simplifying the control signals from the central nervous system. This idea is described and related to many well-known theories in the field, and two models are studied by means of simulation. These models account for the bell shaped speed profile and the stereotyped relationships between duration amplitude and peak velocity of rapid movements.

4.1. Dynamics Determine Single Solution

In our general model for redundant system control, see Figure 18, we emphasize the fact that the dynamics of the mechanical and low level neuromuscular system has an important role in reducing the redundancy and simplifying the role of the higher neural system. We include various mechanisms of the lower level in one box and call it dynamics determine single solution (DDSS). There are many examples for apparent redundancies that are generally reduced in a single typical way. In this section we briefly describe some of the theories and experimental evidence that support the DDSS view, and then in the next sections we describe simulations of mechanical models that demonstrate the role of the nonlinear properties of the muscles.
4.1.1. Motor units' recruitment order

There are many motor units in each muscle that can potentially be activated in many
different combinations in order to produce a desired moment in the muscle. However,
many experiments showed a stereotypical order of the motor units recruitment. There
is a good correlation between the size of the motor neuron, the diameter of its axon
and the force that is generated by the motor unit. It was found that the small units are
activated first and then the larger units are activated. This empirical law is known as
the size principle (see, Hanneman et al. 1965, DeLuca and Erim 1994, Desmedt
1983). Hanneman proposed that the smallest motor neurons recruited first since their
threshold is smaller. See also Akazawa and Kato (1990) for a neural network model
that is based on this principle.

This stereotypical recruitment order is reasonable because it keeps the precision of the
control nearly constant in all the force levels. It is economic, since the largest units
that are rarely used have low maintenance cost. These large units are fast fatigable
units that are based mainly on anaerobic metabolism, and therefore their metabolic
cost is low. This stereotypical recruitment order is an example of the DDSS, since it
reduce the burden of the CNS that does not have to specify the activation of each
motor units but rather just the desired force of the whole muscle.

4.1.2. Central pattern generators

Central pattern generators in the nervous system can simplify the control commands
that are needed from the brain and therefore in some sense they assist it in reducing
the redundancy. Simple rhythmic walking pattems and many complex reflexes, can
be generated by the spinal cord dynamics without detailed command from the CNS
(see Chapter 38 in Kandel et al. 1991). See also Arshavsky et al. (1997) where the
redundant structure of the pattern generators is emphasized with its role in improving
their reliability and flexibility. This is a good example of the hierarchical structure of
the biological system and of the role of redundancy through all these hierarchical
levels. There are also models of pattern generators in the cerebellum (e.g., Houk et al
1990, and Barto et al. 1999) where the term adjustable pattern generator (APG) is
used. In all these models of pattern generators the main principle is to simplify the
control strategy by means of some neural dynamic mechanism that produce a typical behavior.

4.1.3. The equilibrium point hypothesis (EPH)

The EPH is a controversial theory with many versions, for examples of recent arguments see, Shadmehr (1995), Bizzi et al. (1992), Feldman and Levin (1995), Gomi and Kawato (1996), Gottlieb (1998), and Feldman (1998). The basic principles are spring-like muscles and musculo-spinal mechanisms that produce a force field that govern the movement and simplify the control commands from the brain. This theory deserves a long discussion in order to be described and fully account for. The interested reader is referred to the sources above and the many references therein. In the perspective of our general model, once the higher level issued its command, that is called an equilibrium point or an equilibrium trajectory, the lower level determines the single solution according to its dynamics. Therefore, the musculo-spinal mechanism, according with the EPH, perfectly serves the role of the DDSS in our general model.

The core of this chapter is in the following two sections were we demonstrate the important role of the nonlinear properties of the muscle dynamics in simplifying the control strategy for rapid movements, and producing the stereotypical features of these movements.

4.2. Bell Shaped Speed Profile

4.2.1. The phenomenon

The reaching movement is a fast movement of the arm towards a given target. This movement was studied experimentally and its main characteristics are roughly a straight-line path and a bell shaped speed profile (see Abend et al. 1982 for an experimental study, and Arbib and Hoff 1994 for modeling review). This speed profile was found to be consistent with minimization of the jerk during the movement, i.e., minimization of the following cost function, see Flash and Hogan (1985).
\[ C = \frac{1}{2} \int^T_0 \left( \left( \frac{d^3 x}{dt^3} \right)^2 + \left( \frac{d^3 y}{dt^3} \right)^2 \right) dt \]

This minimization was assumed to be carried out by the nervous system in several models, see Jordan et al. (1994), and Kawato and Gomi (1992). These two models were described in the Chapter 2. They are complex control schemes, which contain a forward and/or an inverse model of the system, and both have to deal with the time (i.e., have to calculate the control signal, and propagate the error throughout the trajectory). These and other studies use a linear model of the muscles. In the next subsections, we will show that when the non-linear muscle model is incorporated, the desired performance, i.e., a bell shaped speed profile with a smooth stop, can be achieved with a simple control strategy and without the need for a complex optimization procedure.

4.2.2. The hill type mechanical model

Let us look at the Hill type mechanical model of the muscle in Figure 19. This model is taken from Zangemeister (1981) with minor changes; see Kamiel and Inbar (1999b) for an introductory to the use of electrical and mechanical models, and Karniel and Inbar (1997) for a detailed description of this model and this example.

![Figure 19](image_url)

**Figure 19:** The mechanical model of the muscle. \( n_i \) is the neural input. The first order filter represents the activation contraction coupling. \( T_0 \) is the hypothetical force in the muscle. \( B \) represents the relation between force and velocity from Hill's model. The other elements represent the mechanical properties of the tendon and other connective tissues around the joint.

Following are the differential equations of this mechanical model:
\[
F_0 = \frac{1}{\tau_o} \cdot (n_i - F_0) \\
T_0 = F_0 \cdot F_{\text{max}} \\
\dot{X}_0 = \frac{K_s \cdot (X - X_0) - T_0}{B} \\
F_m = B \cdot \dot{X} + K_s \cdot (X - X_0)
\]

This model was derived from the Hill model (Hill 1938) and in the Hill model the value of the viscose-damping element, B, depends on the internal force and on the contraction velocity as follows:

\[
B = \begin{cases} 
  a \cdot T_0 / (b + \nu) & \nu \geq 0 \\
  a' \cdot T_0 & \nu < 0
\end{cases}
\]

(2)

The value of B was taken as a constant in several models, for the sake of simplicity in order to get a linear model of the muscle (e.g. Jordan et al. 1994, Massone and Myers 1996). This linear model is under-damped and therefore overshoot and oscillations are most likely to appear in the controlled movement. This problem is avoided by the use of the non-linear model and we demonstrate this for a very basic movement, the reaching movement.

4.2.3. The pulse excitation

The neural input to the muscles represents the envelope of the electromyography and it is modeled by rectangular pulses, see Figure 20.

![Flexor and Extensor pulses](Image)

**Figure 20:** The shape of the excitation pulses to the muscles

A similar description of the control command was used by Gottlieb (1993), and there are evidence that the nervous system can generate such characteristic pulses, see for
example the adjustable pattern generators in Houk and Wise (1992). We will not describe here the details of the pulses' parameters, and the control and learning scheme. The details of this model are described in Karniel (1996) and in Karniel and Inbar (1997).

4.2.4. The simulation result

The performance of an anthropomorphic arm with linear and non-linear muscle model in response to rectangular control pulses is shown in Figure 21. For the linear muscle model the value of the viscose-damping element, $B$, is constant instead of the Hill type relation in (2).

![Figure 21: A comparison between the simulated speed profile of the end point of a two degrees of freedom anthropomorphic arm with a linear muscle model (left) and with a nonlinear muscle model (right) in response to typical rectangular pulse activation of the muscles. Only the non-linear muscle model yields a bell shaped speed profile with a smooth stop. (For more details, see Karniel and Inbar 1997) ](image)

It can be seen that the arm with the linear muscle model, in response to pulses at the input, does not stop when the target is reached, and has an overshoot and an oscillatory behavior at the end of the movement. Under the same conditions, the nonlinear muscle can evoke a fast movement with a smooth stop. This is only a demonstration, however, this example is representative of the improved arm performance achieved with a nonlinear muscle model under the assumed conditions, see Karniel and Inbar (1997). The reason for this phenomenon can be explained by
observing the behavior of a simple second order system in its standard form. The transfer function in the Laplace transform domain is:

\[ \frac{w_n^2}{s^2 + 2 \cdot \xi \cdot w_n \cdot s + w_n^2} \]

\( w_n \) is the natural frequency, \( \xi \) is the damping coefficient, and \( a \) is the gain. Let us look at the overshoot (OS) and the time to reach the maximum, \( t_{\text{max}} \):

\[ t_{\text{max}} = \frac{\pi}{w_n \cdot \sqrt{1 - \xi^2}} \quad \text{OS} = \text{EXP} \left( \frac{\pi \cdot \xi}{\sqrt{1 - \xi^2}} \right) \]

For the human arm, the system is under-damped, i.e. \( \xi < 1 \) (see for example Inbar 1996). One can see from the equation above that as \( \xi \) gets smaller the movement is faster but the overshoot is bigger. In a linear system, a tradeoff exists between small overshoot and fast movement. In a nonlinear system, the parameter can change during the movement to achieve a fast movement without any overshoot, and we suggest that this is what happens in the muscles.

The same qualitative phenomenon can be observed in many other muscle models. Let us examine the second order model that is described by the following transfer function (see subsection 2.3.3 for the corresponding mechanical model)

\[ \frac{X(s)}{P(s)} = \frac{-1}{M \cdot s^2 + B \cdot s + K} \]

The parameters of the transfer function in the standard form are:

\[ a = \frac{1}{K}, \quad w_n = \sqrt{\frac{K}{M}}, \quad \xi = \frac{B}{\sqrt{4 \cdot M \cdot K}} \]

In trying to find the parameters of such a model, it was shown that \( K \) changes during the movement and it has a positive relation to the activation of the muscle, see, e.g., Inbar (1996), and Hunter and Kearney (1982). Changes in \( K \) have the same qualitatively effect as the changes in \( B \) in the Hill type model above. Another non-linearity that can produce a stop without oscillation is the one-fifth power law in the viscosity that was found in studies of human wrist movement (see Wu et al. 1990, and

59
the example in the next section). The inverse relation between the damping force and the contraction velocity in the model of Hill, and the increasing stiffness at the end of the movement also fits with the arm movement measurements by Gomi and Kawato (1996, 1997). These measurements were recently explained by means of another nonlinear 6 muscles model, with simple control signals, see Gribble et al. (1998). The main conclusion from these examples is that the nonlinear properties of the muscle may have a functional role in simplifying the control strategy. The nonlinearities make the system analysis process most difficult, however they are probably used advantageously in the biological system.

4.3. Stereotyped Relationships

4.3.1. The experimental phenomenon

Rapid human movements exhibit a quasilinear relationship between their amplitude and maximum velocity, and a log-like relationship between their amplitude and duration. (For saccadic eye movements, see Robinson 1964 and Bahill et al. 1975; for hand reaching movements, see e.g., Plamondon 1995 and references therein). These two features were found also in fragments of tracking movements; see Hanneton et al. (1997), and Figure 22. This finding suggests that the rapid movements are also the basis of other slower and more complex movements.

Various reasons, purposes, and optimization schemes might account for the stereotypical features of rapid movements. We assert that the origin of these features can be the electromechanical properties of the musculoskeletal system rather than a computation process in the nervous system. In this section, we show that the relationship between amplitude duration and maximum velocity can be obtained by a simple nonlinear musculoskeletal model with a simple pulse-step control scheme. First, the model is described, and then the simulations are shown and discussed.
Figure 22: An example of the relationships between the maximum velocity and the movement amplitude (upper graph), and between the movement duration and the movement amplitude (lower graph). Each dot represents one short, stereotyped period of arm movement. The lines are linear and logarithmic fits to the data. The methods and the details of this tracking movement experiment are described in Hanneton et al. (1997). This figure is adopted with permission from Figure 3 of Hanneton et al. (1997), © 1997, Springer-Verlag.

4.3.2. The mechanical model

There are many muscle models of various types and complexities (see for examples, Hill 1938, Winters and Stark 1987). In this work we choose the model of Wu et al. (1990) which is a very simple model that incorporates nonlinear properties that are essential in order to produce a fast and damped movement with a simple rectangular neural excitations.

Gielen and Houk (1984) found that human wrist movement could be modeled by the following differential equation that is referred to as the one-fifth power law. See also Gielen and Houk (1987).

\[ m \cdot \ddot{x} + b \cdot \dot{x}^{\frac{1}{5}} + k \cdot (x - x_{eq}) = 0 \]  

In this equation \( x \) is the position (in meters) of the controlled object and \( x_{eq} \) is the resting, or equilibrium position. The mass of the object is \( m \) (kg), the stiffness
The control signal $x_{eq}$, which represents the neural command, was chosen to be a pulse-step control signal, as modeled by Barto et al. (1995). See Figure 23, and a full paper of Barto et al. (1999).

This command produces a fast movement, which is sometimes called step-and-hold movement. There are three parameters that define the control command, the pulse amplitude $X_p$, the pulse duration $d$, and the step amplitude $X_s$. At the end of the rapid movement, the position of the object is not necessarily $X_s$, because the system tends to "stick" at non-equilibrium position and then to drift very slowly to the equilibrium position. Following Barto et al. (1995), we define the termination of the movement, that is, the position where the object sticks, as the position at which its absolute velocity falls and remains below 0.005 [m/s]. We denote by $X_n$ the value of $x$ at the termination of the movement, and by $T_n$ the termination time of the movement. We also observe the maximum velocity during the movement and denote it by $V_m$.

The simple model and the simple control strategy described above are used in the next section in order to simulate rapid human movements and examine the relationship between their parameters.
4.3.4. The simulation result

We begin this section with a simulation of a typical rapid movement (see Figure 24) and discussion of its properties; we then chart the relationships between the parameters of such movements.

\[ d=0.15 \quad X_\text{p}=0.1 \quad X_\text{s}=0.01 \]

![Graph](image)

**Figure 24: A typical simulated rapid movement**

The nonlinear viscosity in the model gives the system the potential to produce rapid movement that stops without oscillations. The same phenomenon appears in other nonlinear muscle models, such as the Hill model where the damping coefficient becomes larger at small velocities, and also in models with nonlinear stiffness. In the previous section, we used a Hill-type model and demonstrated the role of the nonlinear properties of the muscles. A nonlinear muscle model is essential in order to produce rapid movement with a smooth stop in response to simple pulses or pulse-step excitation regime. The model used in the previous section contains more elements and therefore produces smoother movements. The one-fifth power law was chosen for this section due to its simplicity.

The one-fifth power law (3) was validated for positive velocities, however at the end of the movement, small negative velocities might occur. Therefore, a reflection about the origin is assumed, that is, the following differential equation is solved:

\[ m \cdot \ddot{x} + b \cdot \text{sign}(\dot{x}) \cdot |\dot{x}|^{\frac{1}{5}} + k \cdot (x - x_{eq}) = 0 \]
The simulation was repeated for various values of the command parameters $X_p$, $d$, $X_p$, and the values of the movements’ parameters $V_m$, $X_s$, $T_n$ were examined. A salient tendency towards the stereotype relationships was observed over large part of the command parameter space. An example is illustrated below for a various pulse amplitude values, see Figure 25 and Figure 26.

![Graph](image)

**Figure 25:** Maximum velocity against movement amplitude of 17 simulated movements with different pulse magnitude. The solid line is a linear fit to the data.

![Graph](image)

**Figure 26:** Movement duration against movement amplitude of 17 simulated movements with different pulse magnitude. The solid line is a logarithmic fit to the data.

This simple control scheme of changing the pulse amplitude was described and studied extensively by Gottlieb et al. (1989), where it is named speed-sensitive.

We saw that in the amplitude modulation there is a salient tendency toward the stereotypical relationships, however for duration modulation of the control pulses this tendency is very limited; see Figure 27, and Figure 28. In this scheme, the stereotype
relationships are observed only for very small amplitude of the movements. However, for longer movements the results deviate from the stereotype relationships. This control strategy is referred to by Gottlieb et al. (1989) as speed insensitive.

![Graph](image.png)

**Figure 27:** Maximum velocity against movement amplitude of 15 simulated movements with different pulse duration. The solid line is a linear fit to data in the amplitude region [0-0.1].

![Graph](image.png)

**Figure 28:** Movement duration against movement amplitude of 15 simulated movements with different pulse duration. The solid line is a logarithmic fit to data in the amplitude region [0-0.1].

It will be interesting to check whether the nervous system avoids the control commands that derive movements that do not obey the stereotypical relationship. Another possibility that could be check experimentally is that there are movements that do not obey the stereotypical relationship, and they are buried in the variance of the experimental observations. Another possibility is this model being too simple to grasp the whole range of this phenomenon. A detailed physiological model might
produce the stereotypical relationships in even a larger extent of the control signals range.

We can conclude this section by observing Figure 25, and Figure 26, which suggests that a physiological plausible control strategy and nonlinear muscle model can reproduce the well-observed relationships between amplitude duration and maximum velocity without complicate computation in the central nervous system.

4.4. Conclusions

In this chapter, we described the DDSS part of the general model proposed in Chapter 3. We have described the role of the nonlinear properties of the musculoskeletal system in simplifying the control strategy and in facilitating the production of the stereotypical features of rapid movements. These issues are farther discussed in Chapter 7. The next chapter describes the higher level of our general model, that is the notion of multiple controller.