Technical University of Crete
Electronic & Computer
Engineering Department

Diploma Thesis

Subject:

USE OF A DIRECT ADAPTIVE METHOD FOR REGULATING ENZYME ACTIVITY

by Michael S. Stathopoulos

Committee: Prof. M. Christodoulou (Thesis Advisor)
Prof . M. Zervakis
Assoc. Prof. E. Georgiou

August 2006

Acknowledgements

I would like to express my thanks to my advisor Professor M. Christodoulou for his support and guidance in the preparation of this thesis.

ABSTRACT

Cell signaling plays an important role in many functions of the cell, as the development or proliferation. Therefore the regulation of signaling is essential and its malfunction causes various diseases. For instance the cancer where cells proliferate despite the death signals, or Parkinson disease where too many neurons die.

A signal transduction pathway, or a network of biochemical reactions, is a convenient model of the observed reality. These pathways represent various signals which are realized through changes of concentrations and help us to model the response of gene expression or protein concentrations to various stimulations. Scientists try to study the protein functions and interactions within cell. They are also interested in the regulation of protein interactions, which is prerequisite in order to achieve changes in a pathway. Many biological re-searches have been conducted in this field.

In this project our aim is the regulation of enzyme activity by changing the rate contants. We focus on the basic pathway of enzyme reactions, which is the Michaelis-Menten mechanism for basic one-substrate enzyme reactions. It consists of three elementary reactions steps and it is the basic scheme of an enzymatic reaction. The more complex signal transduction pathways can be represented as sequences of this basic reaction's scheme.

We implement a Direct Adaptive method using Recurrent High-Order Neural Networks (RHONNs). We take also into consideration destabilizing factors as modeling errors.

The organization of this project is the following. In Chapter 1 there is a general overview of Systems Biology, the scientific field which includes this type of applications. In Chapter 2 RHONNs are introduced. In the next Chapter Direct Adaptive methods are discussed. In Chapter 4 we focus on enzyme kinetics reactions and especially on the regulation of enzyme activity. In the

next Chapter a Direct Adaptive Method is implemented. Furthermore results are presented and analyzed. Finally, in Chapter 6 there are final conclusions and future work on the field is suggested.

CONTENTS

CHAPTER 1
A GENERAL OVERVIEW OF SYSTEMS BIOLOGY

1.1 Definition of Systems Biology

Biology has made important steps in the last decades. Biologists have obtained a clear image of the inner part of a living cell, which is the basic element of living matter.

Sequencing of DNA was a very significant achievement of Biology and revealed a huge number of genomes. The next step is the research about the interactions among genes, among proteins or between genes and proteins. Cell's metabolism is composed of various functions as division or growth, where reactions among proteins occur. These interactions are regulated by complex mechanisms. This interest in the research of functional activity is reasonable. The identification and analysis of intra- and inter-cellular processes will help scientists to understand the function of the cell, explore other issues as the mechanisms of various diseases and finally find a treatment.

Since all these functions and interactions are dynamic processes there is high interest in applications of System Theory in Biology. This field of science is known as Systems Biology.

Specifically: "*Systems Biology investigates the functioning and function of inter- and intra-cellular dynamic networks, using signal- and systems-oriented approaches"* [3].

1.2 Enzymes and Proteins

Most of proteins are enzymes. Enzymes are biological catalysts and take part in DNA replication, protein synthesis and generally the cell's metabolism. In cells and organisms most reactions are catalyzed by enzymes, which are regenerated during the course of a reaction. Enzymes are physiologically important because they speed up the rates of reactions that would otherwise be too slow to support life. Enzymes increase reaction rates, sometimes by as much as one millionfold, but more typically by about one thousand fold.

Although enzymes are responsible for the catalysis of almost all biochemical reactions, it is important to mention that rarely, if ever, do enzymatic reactions proceed in isolation. In almost all cases enzymes catalyze individual steps of multi-step pathways, as is the case with glycolysis, gluconeogenesis or the fatty acids. As a consequence enzymes are dependent on the activity of preceding reaction steps. Usually the step with the lowest rate-constant is the step which limits the rate of the overall pathway.

Generally catalysts speed up the forward and reverse reactions proportionnately so that, although the rate constants of the forward and reverse reactions increase, the ratio of the rate constants remains the same in the presence or absence of enzyme. Enzymes increase reaction rates by decreasing the activation energy of a reaction, as we can see in Figure 1.1.

Activation energy is the amount of energy required for a reaction to proceed. As we can see in Figure 1.1, some bonds need to be broken in order to form new bonds and finally energy is released. In this example the formation of new bonds releases more energy than the energy that was initially input to break some initial bonds.

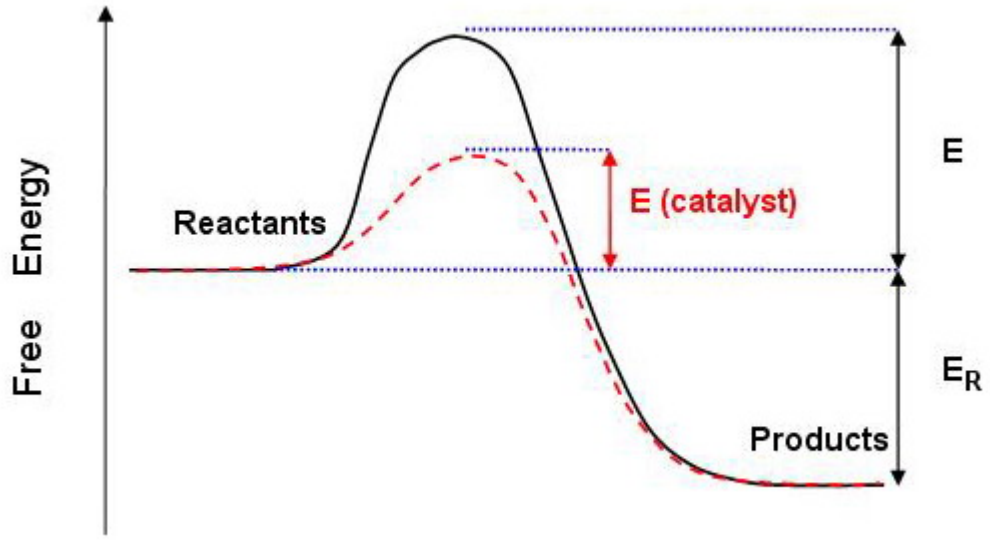

Fig. 1.1: Activation Energy of Uncatalyzed/Catalyzed reaction. The Catalyst increases reaction rate by decreasing the activation energy of the reaction.

Researchers in the 1960s proved that the most important processes of cell are dynamic. A significant indication about that was the process of protein syn-thesis and the control of its rate.

1.3 Signal transduction pathways

1.3.1 Definition of signal transduction

Cells take part in complex interactions which are influenced by various fa-ctors, as environmental conditions. These interactions are based on transfer of information between and within cells.

***Definition**: "Cell signaling or signal transduction is the study of the me-chanisms that enable the transfer and processing of biological infor-mation. "[1]*

1.3.2 The states of a signal transduction pathway

There are three possible states of a pathway:

- equilibrium
- steady-state
- transient state

Equilibrium state

In this state all the reaction rates of biochemical network are zero and there is balance. According to Biochemistry, only isolated systems can come to this state.

Steady state

In this state the quantities are time-invariant and reaction rates are not zero.

Theoretically steady state is never achieved, only approached asymptotically. In practice, one can regard that a system is in steady state. When for a certain set of initial conditions a steady-state is unique, this state will be approached independently from the initial concentrations. Furthermore it is possible more than one steady states exist for a given set of parameters. In this case, the steady-state depends on the initial conditions(the history of the system).

The stability of the pathway is connected with the steady state. Steady state cannot maintained or exist without the stability with respect to fluctuations or perturbations of parameters. Usually we consider that steady state is asymptotically stable. This means that if we regard a set of steady state concentrations, then initial concentrations in the neighbourhood of these concentrations cause the same steady state.

Transient state

A signal transduction pathway is in this state before the steady state for a set of initial conditions, or from one steady state to another after a fluctuation of a parameter.

1.3.3 Regulation of signal transduction pathways

Cell signaling plays an important role in many functions of the cell, as the development or proliferation. Therefore the regulation of signaling is essential and its malfunction causes various diseases. For instance the cancer where cells proliferate despite the death signals, or Parkinson disease where too many neurons die.

A signal transduction pathway is a network of biochemical reactions and is a convenient model of the observed reality. In the case of cells the term *signal*s refers to the processing or regulation of information. These pathways represent various signals which are realized through changes of concentrations and help us to model the response of gene expression or protein concentrations to various stimulations.

The regulation of protein interactions is prerequisite in order to achieve changes in a pathway. Many biological researches have been conducted in this field.

Figure 1.2 illustrates a basic signal transduction model.

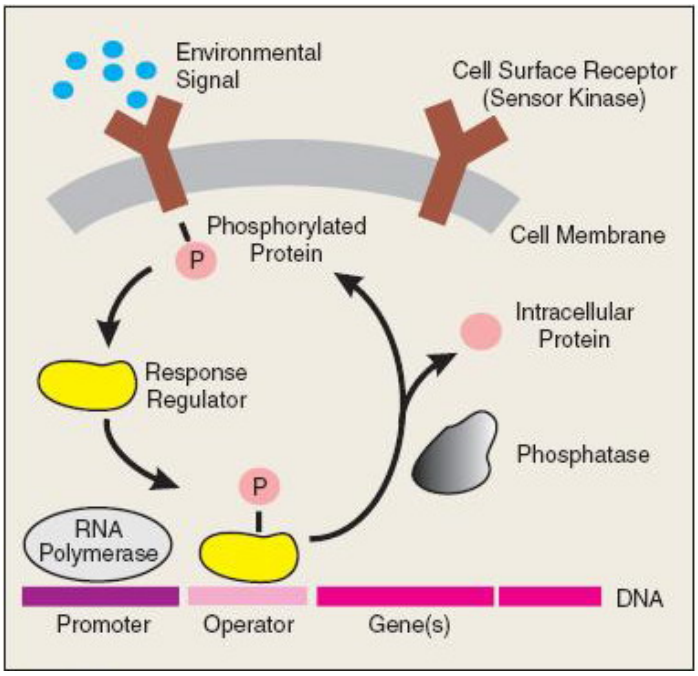

Fig. 1.2: A Signal Transduction Model.
Gene expression can be affected by environmental conditions. Receptors, which are proteins, span the cell membrane. Then another protein, the response regulator, is activated by phosphorylation [1].
(Reproduced by permission from Mr O. Wolkenhauer)

As it is shown in Figure 1.2, gene expression can be affected by environmental conditions. The cells have the appropriate mechanisms which regulate a specific target. Receptors, which are proteins, span the cell membrane. Then another protein, the response regulator, is activated by phosphorylation. Generally a pathway consists of many more intermediate steps until the signal transduction come to an end.

1.4 Modeling

1.4.1 Introduction to models

A model is a precise representation of a system's dynamics. Models allow us to study systems and predict their behaviour. Real world has high comple-

xity. Consequently modeling and simulation of systems, as cell-biological, is necessary.

The creation of a model requires some assumptions or simplifications which do not affect the ability of prediction. The main difficulty is the knowledge about the prediction of a complex hypothesis with many processes. Mathematical modeling is a common technique for this[4]. Simulation can reveal us what a model predicts precisely for a given experimental situation. Alternatively, in the case of the model in Figure 1.2 it validates the signal transduction model for various initial conditions. Consequently modeling and simulation are basic steps of the validation of hypothesis.

A common class of mathematical models for dynamical systems, and consequently for applications in cell biology, is ordinary differential equations (ODEs), like the following:

$$\dot{x} = f(x,u)$$
$$y = g(x,u)$$

where *u* represents external influences or input. The input is the factor which help us to analyze the influence external disturbances have on the trajectories of a system. Or, in the case when the input is something that can be modulated in a controlled way, we can analyze whether it is possible to control the direction of the system from one point in the state space to another through proper choice of the input. Furthermore model uncertainty and disturbances are important aspects of control.

In addition to performing simulations, models can also be used to answer other types of questions. For example, stability at an equilibrium point using Lyapunov stability analysis.

Schematically, the basic method which is used in Systems biology is shown in Figure 1.3 [3]. As we can see modelling and simulation are very useful in biological study because help scientists validate and generate hypotheses. Moreover modeling supports the design of experiments because contributes to the choice of the proper variables to measure.

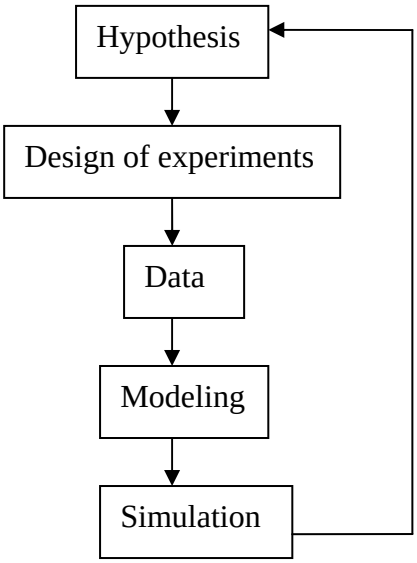

Fig. 1.3: The basic scheme of method of Systems Biology. Modeling and Simulation validate and generate hypotheses. Modeling, also supports the design of experiments. [3]

1.4.2 State-space models

State-space models are used in order to predict the evolution of the system state from a given initial condition. Alternatively one can also use state-space models to analyze the overall behavior of the system.

A common form of a model is differential equations. In this form the state is a collection of variables that summarize the past of a system for the purpose of prediction of the future. The state variables are gathered in a vector $x \in R^n$, called the state vector. The control variables are represented by another vector $u \in R^p$ and the measured signal by the vector $y \in R^q$.

A system can then be represented by the differential equation:

$$\dot{x} = f(x,u)$$
$$y = g(x,u)$$

The dimension of the state vector is called the order of the system.

The system is called time-invariant because the functions f and g do not depend explicitly on time *t*. It is possible to have more general time-varying systems where the functions depend on time. The model thus consists of two functions.

State-space models can represent the signal transduction pathways. The concentrations of proteins are the states of the system. Each step of the pathway has a protein as a member and a system of differential equations can give us the dependencies of these steps.

CHAPTER 2
RECURRENT HIGH-ORDER NEURAL NETWORKS

2.1 Introduction

Artificial neural networks have been extensively studied in the last decades, especially in the fields of speech and image processing. The aim is the approximation of human ability to solve problems in these fields. Furthermore new network topologies have attracted the interest of scientists mainly about control problems.

There are several types of neural networks used in control systems. The choice of the appropriate network and training method depends on the application. For instance, feedforward multilayer neural network, where no information is feedback during operation. However, there is feedback information during training. Also, supervised learning methods, where the neural network is trained to learn input-output patterns presented to it, are used. This process is slow and time consuming, because the algorithm takes a long time to converge. Moreover other methods as Backpropagation (BP) algorithm, which adjusts the weights during training, or recurrent networks are used.

Theoretical studies have proved that multilayer neural networks with one hidden layer can approximate any continuous function uniformly over a compact domain by adjusting synaptic weights in order to minimize the error between the network output and the output of the unknown map [27], [28], [29], [30].

Forward modeling is the training of a neural network to model the forward dynamics of a plant. The neural network model is placed in parallel with the plant and the error between the plant and the network outputs is the training

signal. The training procedure may needs discrete samples of the plant inputs and outputs.

If we consider that the system output at time *k+1* depends on the past *n* output values and the past *m* values of input *u,* we have:

$$y^p(k+1) = f(y^p(k),...,y^p(k-n+1);u(k),...,u(k-m+1))$$

Thus, the output of the neural network is:

$$y^m(k+1) = f_{apr}(y^p(k),...,y^p(k-n+1);u(k),...,u(k-m+1))$$

f_{apr} represents the nonlinear input-output map of the network, or the approximation of f . It is clear that the network's input includes the past values of the real system's output. In other words the system has not feedback. After the training, the network approximates the plant or $y^m \approx y^p$. When this is true, the network's output and the delay values can be fed back and be part of the network's input. In this way, the network can be used independently of the plant and the model function can be written:

$$y^m(k+1) = f_{apr}(y^m(k),...,y^m(k-n+1);u(k),...,u(k-m+1))$$

The information about the plant can be in the form of an input-output table. In this case the training of the network necessitates current and previous inputs or outputs of the plant. Alternatively the states of the plant or their derivatives can be used. Consequently, for the case of feedforward multilayer neural network and BP training algorithm we consider discrete or discretized continuous plant, as is described in Figure 2.1.

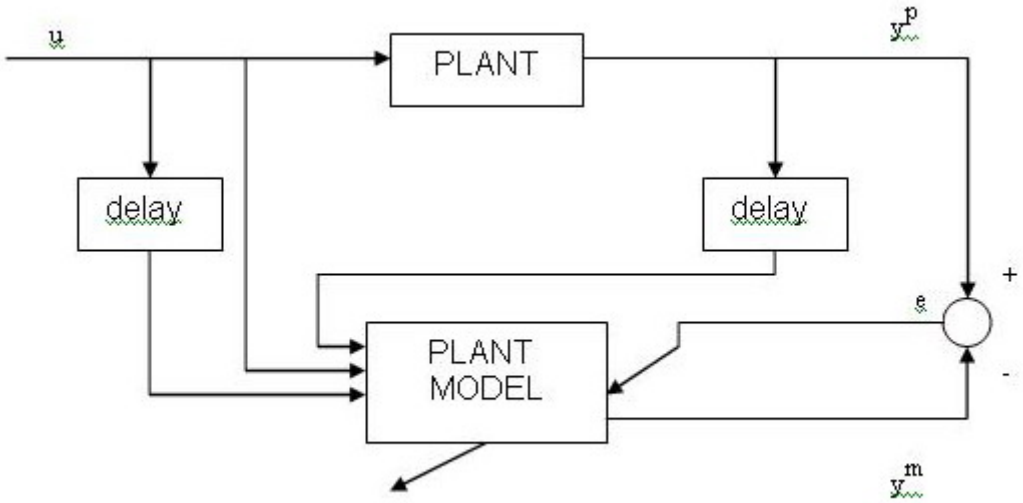

Fig. 2.1: Plant identification with a multilayer neural network[5 (chapter 1)]

Another approach of training aims to identify the inverse dynamics of the plant. In this case the network's input is the plant's output and the plant's input is the network's output. The training signal is the error between the actual input of the plant and the network's output. The current input of the plant is the de- sired output of the network. We have assumed that the inverse of the plant is unique. If the inverse is not unique, we must restrict the ranges of the input to the network.

The architecture of the network is chosen appropriately according to the case. The first step is the identification of the plant. Then, a controller can be designed. In Figure 2.2 the training of a neural network as open loop con- troller is described. The error $e = y - y_d$ is used in order to train the net-work. As we can see the error is backpropagated through the plant.

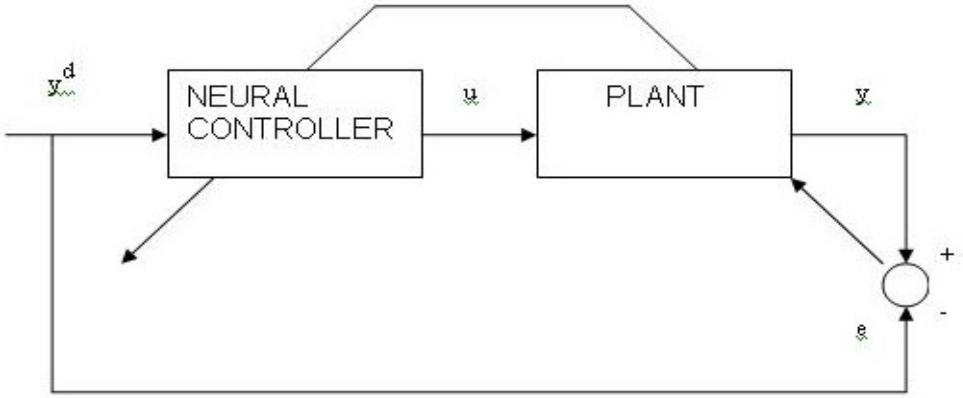

Fig. 2.2: A neural network controller [5 (chapter 1)]

We must add that the neural model of a controller can include mean squared error between the reference output and model output or other terms as the error between the reference input and real output and the input *u*.

In most applications we meet dynamical systems which necessitate the use of feedback connections in order to approximate them. Such networks are known as recurrent. A static neural network can also be transformed to a dynamic one, by simply connecting the past neural outputs as inputs to the neural network, thus making the neural network a very complicated and highly nonlinear dynamical system.

The main difficulty with the dynamic neural networks that are based on static multilayer networks is that the synaptic weights appear nonlinearly in their mathematical representation. This creates a number of significant hurdles. Firstly, the learning laws that are used require a high computational time. Secondly, since the synaptic weights are adjusted to minimize a functional of the approximation error and the weights appear nonlinearly, the functional has many local minima so there is no way to ensure the convergence of the weights to the global minimum. Moreover, due to the highly nonlinearity of the neural network architecture, basic properties like stability, convergence and robustness are very difficult to verify. On the other hand for the recurrent neural networks that possess a linear-in-the weights property, is feasible to prove the stability and convergence properties.

In this chapter Recurrent High-Order Neural Networks' (RHONN) structure is introduced and their approximation capabilities are analyzed. This network scheme approximates nonlinear systems whose vector fields satisfy a local Lipschitz condition.

2.2 Identification of dynamical systems using Recurrent High-Order Neural Networks

Recently there is renewed interest in the usage of neural networks for modeling and identification of dynamical systems in the form of feedback connections, most known as recurrent neural networks (RNN).

Several training methods are used. For example, recurrent backpropagation [6], backpropagation-through-time algorithms[7], real-time recurrent learning algorithm[8] and the dynamic backpropagation[9] algorithm. The last method is based on the computation of sensitivity models for generalized neural networks. Generalized neural networks combine feedforward neural networks and dynamical components of stable rational transfer functions. All these training methods have been widely used in empirical studies and had many drawbacks. First of all they rely on the approximation of computing a partial derivative. Moreover these methods need much computational time. Also, it is very difficult to produce analytical results for the convergence and stability.

An interesting effort is the design of training methods based on the Lyapunov stability theory [10], [11], [12], [13], [14], [15], [16], [17], [18]. These me-thods have the advantage of stability, convergence and robustness proofs which promotes control theory.

RHONNs are appropriate for identification models. High-order networks are expansions of the first-order Hopfield [19] and Cohen-Grossberg [36] models which allow higher-order interaction between neurons. Their superior storage capacity has been demonstrated in [20], [21], and their stability properties for fixed-weight values have been studied in [37], [22]. Furthermore, several authors have demonstrated the feasibility of using these architectures in applications such as grammatical inference [23] and target detection [24].

In [18] was introduced the idea of recurrent neural networks with dynamical components distributed throughout the network in the form of dynamical neurons and their application for identification of dynamical systems. In this chapter, we combine distributed recurrent networks with high-order connections between neurons. The next section shows that recurrent high-order neural networks can model a large class of dynamical systems. Specifically, it is proven that if enough higher-order connections are allowed in the network then there exist weight values such that the input-output behavior of the RHONN model approximates that of an arbitrary dynamical system whose state trajectory remains in a compact set.

2.3 The RHONN Model

Recurrent neural network models have two way connectivity between units (or neurons). On the other hand, feedforward neural networks have the output of one unit connected only to units of the next layer. In the most simple case, the state history of each neuron is represented by the following form:

$$\dot{x}_i = -a_i x_i + b_i \sum_j w_{ij} y_j \qquad (2.1)$$

where x_i, is the state of the *i-th* neuron, a_i, b_i are constants, w_{ij} are synaptic weights connecting the *j-th* input to the *i-th* neuron and y_j is either an external input or the state of a neuron passed through a sigmoid function (y_j = $s(x_j)$), where $s(.)$ is the sigmoidal.

With respect to the order of a RNN (*k*) the input to the neuron contains the product:

$$\underbrace{y_i \cdots y_j}_{ktimes}$$

For instance, in a second order RNN the input to the neuron is a linear combination of not only y_j but also of $y_j y_k$.

Now we consider a RHONN consisting of *n* neurons and *m* inputs. The state of each neuron is represented by:

$$\dot{x}_i = -a_i x_i + b_i \left[\sum_{k=1}^{L} w_{ik} \prod_{j \in Ik} y_j^{d_j(k)} \right] \tag{2.2}$$

where $I_1, I_2, \ldots, I_L$ is a collection of L not-ordered subsets of $\{1,2,\ldots,m+n\}$, a_i, b_i are real coefficients, w_{ik} are synaptic weights of the neural network and $d_j(k)$ are non-negative integers. The state of the *i-th* neuron is again represented by x_i and $y = [y_1, y_2, \ldots, y_{m+n}]^T$ is the input vector to each neuron defined by:

$$y = \begin{bmatrix} y_1 \\ y_2 \\ . \\ . \\ y_n \\ . \\ . \\ . \\ y_{n+m} \end{bmatrix} = \begin{bmatrix} s(x_1) \\ s(x_2) \\ . \\ . \\ s(x_n) \\ u_1 \\ u_2 \\ . \\ . \\ u_m \end{bmatrix} \tag{2.3}$$

where $u = [u_1, u_2, \ldots, u_m]^T$ is the external input vector to the network. The function $s(.)$ is monotone-increasing, differentiable usually represented by sigmoidals of the form:

$$s(x) = \frac{a}{1 + e^{-\beta x}} - \gamma \tag{2.4}$$

where α, β are the bound and slope of sigmoid's curvature and γ a bias constant. When $\alpha = \beta = 1$, $\gamma = 0$, we obtain the logistic function and by setting $\alpha = \beta = 2$, $\gamma = 1$, we obtain the hyperbolic tangent function; these are the sigmoid activation functions most commonly used in neural network applications.

We now introduce the *L*-dimensional vector z, which is defined as

$$z = \begin{bmatrix} z_1 \\ z_2 \\ . \\ . \\ . \\ z_L \end{bmatrix} = \begin{bmatrix} \prod_{j \in I1} y_j^{d_j(1)} \\ \prod_{j \in I2} y_j^{d_j(2)} \\ . \\ . \\ . \\ \prod_{j \in IL} y_j^{d_j(L)} \end{bmatrix} \tag{2.5}$$

Therefore, the RHONN model (2.2) becomes

$$\dot{x}_i = -a_i x_i + b_i \sum_{k=1}^{L} w_{ik} z_k \tag{2.6}$$

Moreover, if we define the adjustable parameter vector as

$w_i = b_i [w_{i1}, w_{i2}, ..., w_{iL}]^T$ then (2.6) becomes

$$\dot{x}_i = -a_i x_i + w_i^T z \tag{2.7}$$

where vectors $\{w_i : i = 1,2,...,n\}$ represent the adjustable weights of the network and the coefficients $\{a_i : i = 1,2,...,n\}$ are fixed during training.

In order to guarantee that each neuron x_i is bounded-input bounded-output (BIBO) stable, we shall assume that $a_i > 0, \forall i = 1,2,...,n$. In the special case of a continuous time Hopfield model [19], we have $a_i = 1/(R_i C_i)$, where $R_i > 0$ and $C_i > 0$ are the resistance and capacitance connected at the i-th node of the network respectively.

In vector form the dynamic behavior of the overall network is described by:

$$\dot{x}_i = Ax + W^T z \tag{2.8}$$

where $x = [x_1, x_2, ..., x_n]^T \in R^n$, $\underline{W} = [w_1, w_2, ..., w_n]^T \in R^{Lxn}$ and

$A = diag(-a_1,-a_2,\ldots,-a_n)$ a *n x* n diagonal matrix. Since $a_i > 0 \quad \forall i = 1,2,...,n$, *A* is a stability matrix. Vector z is a function of the state *x* and the external input *u*.

2.3.1 Approximation properties

We consider a dynamical nonlinear system of the form:

$$\dot{\chi} = F(\chi, u) \tag{2.9}$$

where $x \in R^n$ is the system state, $u \in R^m$ is the system input and *F* : $R^{n+m} \to R^n$ is a smooth vector field defined on a compact set $Y \subset R^{n+m}$.

Our aim is the approximation of this system. This means that we want to determine weights *W*, such that the RHONN model approximates the input-output behavior of an arbitrary dynamical system of the form (2.9). For this purpose we need enough higher-order connections.

We assume that *F* is continuous and satisfies a local Lipschitz condition such that (2.9) has a unique solution- in the sense of Caratheodory [25]- and (*x*(*t*), *u*(*t*)) $\in Y$ for all *t* in some time interval $J_T = \{t : 0 \le t \le T\}$. The interval J_T represents the time period over which the approximation is to be performed. The next theorem proves that if sufficiently large number of connections is allowed in the RHONN model, then any dynamical system can be approximated to any degree of accuracy.

Theorem 2.1.1

Suppose that the system (2.9) and the model (2.8) are initially at the same state *x(0)* = *χ(0),* then for any *ε*> *0* and any finite *T* > 0, there exists an integer *L* and a matrix $W^* \in R^{Lxn}$ such that the state *x(t)* of the RHONN model (2.8) with *L* high-order connections and weight values *W* = *W** satisfies

$$sup\left|x(t) - \chi(t)\right| \le \varepsilon$$

Proof [5 (chapter 2)]

From (2.8), the dynamic behavior of the RHONN model is described by

$$\dot{x} = Ax + W^T z(x,u) \tag{2.10}$$

Adding and subtracting $A\chi$, (2.9) is rewritten as

$$\dot{\chi} = A\chi + G(x,u) \tag{2.11}$$

where $G(x, u) = F(x, u)-A\chi$. Since $x(0) = \chi(0)$, the state error $e = x-\chi$ satisfies the differential equation

$$\dot{e} = Ae + W^T z(x,u) - G(x,u)\,,\ e(0) = 0 \tag{2.12}$$

By assumption, $(x(t), u(t)) \in Y$ for all $t \in [0,T]$ where Y is a compact subset of R^{n+m}. Let

$$Y_e = \left\{(\chi,u) \in R^{n+m} : \left|(\chi,u) - (\chi_Y, u_Y)\right| \le \varepsilon, (\chi_Y, u_Y) \in \mathrm{Y}\right\} \tag{2.13}$$

It is also clear that Y_e is also a compact subset of R^{n+m} and $Y \subset Y_e$. In simple words Y_e is ε larger than Y, where ε is the required degree of approximation. Since z is a continuous function, it satisfies a Lipschitz condition in Y_e, i.e., there exists a constant l such that for all (x_1, u); $(x_2, u) \in Y_e$:

$$\left|z(x_1,u) - z(x_2,u)\right| \le l\left|x_1 - x_2\right| \tag{2.14}$$

We will show that the function $W^T z(x, u)$ satisfies the conditions of the Stone-Weierstrass Theorem and can approximate any continuous function over a compact domain, therefore.

We infer from (2.2), (2.3) that $z(x, u)$ is in the standard polynomial expansion with the exception that each component of the vector x is pre-processed by a sigmoid function $s(.)$. As shown in [26], preprocessing of input via a continuous invertible function does not affect the ability of a network to approximate continuous functions: therefore, it can be shown readily that if L is sufficiently large, then there exist weight values $W = W^*$ such that $W^{*\mathrm{T}} z(\chi,u)$ can approximate $G(x, u)$ to any degree of accuracy, for all $(x; u)$ in a compact domain.

Hence, there exists $W = W^*$ such that

$$\sup\left|W^* z(\chi,u) - G(\chi,u)\right| \le \delta \tag{2.15}$$

where δ is a constant to be designed in the sequel.

The solution of (2.12) is:

$$e(t) = \int_0^t e^{A(t-\tau)}\left[W^{*T}\ z(x(\tau),u(\tau)) - G(\chi(\tau),u(\tau))\right]d\tau$$
$$= \int_0^t e^{A(t-\tau)}\left[W^{*T}\ z(x(\tau),u(\tau)) - W^{*T}\ z(\chi(\tau),u(\tau))\right]d\tau + \qquad (2.16)$$
$$+ \int_0^t e^{A(t-\tau)}\left[W^{*T}\ z(\chi(\tau),u(\tau)) - G(\chi(\tau),u(\tau))\right]d\tau$$

Since A is a diagonal stability matrix, there exists a positive constant α such that $\left\|e^{At}\right\| \le e^{-\alpha t}$ for all $t \ge 0$. Also, let $L = l\|W^*\|$. Based on the aforementioned definitions of the constants α, L, ε let δ in (2.15) be chosen as

$$\delta = \frac{\varepsilon\alpha}{2} e^{-\frac{L}{\alpha}} > 0 \qquad (2.17)$$

First consider the case where $(x(t),u(t)) \in Y_e$ for all $t \in [0,T]$. Starting from (2.16), taking norms on both sides and using (2.14), (2.15), and (2.17), the following inequalities hold for all $t \in [0,T]$:

$$e(t) \le \int_0^t \left\|e^{A(t-\tau)}\right\|\left[W^{*T}\ z(x(\tau),u(\tau)) - W^{*T}\ z(\chi(\tau),u(\tau))\right]d\tau +$$
$$+ \int_0^t \left\|e^{A(t-\tau)}\right\|\left[W^{*T}\ z(\chi(\tau),u(\tau)) - G(\chi(\tau),u(\tau))\right]d\tau$$
$$\le \int_0^t e^{-\alpha(t-\tau)} L|e(\tau)|d\tau + \int_0^t \delta e^{-\alpha(t-\tau)}d\tau$$
$$\le \int_0^t \frac{\varepsilon}{2} e^{\frac{-L}{\alpha}} + L\int_0^t e^{-\alpha(t-\tau)}|e(\tau)|d\tau$$

From Bellman-Gronwall Lemma [25], we have:

$$|e(\tau)| \le \frac{\varepsilon}{2} e^{\frac{-L}{\alpha}} + e^L \int_0^t e^{-\alpha(t-\tau)}d\tau \qquad (2.18)$$
$$\le \frac{\varepsilon}{2}$$

Now suppose (for the sake of contradiction) that (x, u) does not belong to Y_e for all $t \in [0, T]$; then, by continuity of $x(t)$, there exist a T^*, where

0 <T^*< T, such that ($x(T^*)$, $u(T^*)$) $\in \partial Y_e$ where ∂Y_e denotes the boundary of Y_e. If we carry out the same analysis for $t \in [0, T^*]$ we obtain in this interval $|x(t) - \chi(t)| \leq \frac{\varepsilon}{2}$, which is clearly a contradiction. Hence, (2.18) holds for all $t \in [0, T]$. ■

This theorem proves the existence of solution, but it does not provide any constructive method for obtaining the optimal weights W^*.

CHAPTER 3
DIRECT ADAPTIVE CONTROL

3.1 Introduction

This chapter introduces direct adaptive control for affine in the control nonlinear dynamical systems. In adaptive control there is estimation of unknown parameters at each instant and a control law is used. The objective is the approximation of the actual system by the model system. There are two basic approaches: direct and indirect.

In indirect adaptive method there is on-line estimation of the actual system parameters and then the controller parameters are calculated. In direct adaptive method the model system's parameters are estimated according to the controller parameters which are estimated directly without estimation of plant parameters.

In this chapter we use RHONNs. Also destabilizing factors as modeling errors are discussed. In this case the appropriate changes to update and control laws guarantee robustness and the uniform ultimate boundness property. We focus on regulation issues and consider the more general case where the number of states is different from the number of control inputs.

3.2 Modeling errors with unknown coefficients

We consider the more general case where the modeling error has polynomial growth of first order with unknown growth magnitude. The update and

control laws are modified appropriately in order to guarantee the robustness property.

We assume that the actual system can be modeled by a RHONN plus a modeling error term:

$$\dot{x} = -Ax + W^* S(x) + W_1^* S'(x)u + \omega(x,u) \tag{3.1}$$

Furthermore we assume that the modeling error satisfies:

$$|\omega(x,u)| \le k_0 + k_1(|x| + |u|) \tag{3.2}$$

where k_1 >0 is unknown and k_0 >0 known positive constants.

3.3 Adaptive regulation

3.3.1 Modeling error at zero case

Here we examine more general systems where *n!=m* (*n* the states and *m* the inputs). We consider the case where the actual system can be modeled by RHONN plus a modeling error:

$$\dot{x} = -Ax + W^* S(x) + W_1^* S'(x)u + \omega_0(x,u) \tag{3.3}$$

where $x \in R^n$ is the state vector, $u \in R^m$ is the control input, *A* is *n x n* matrix of positive eigenvalues which can be considered diagonal, *W** is a *n x L* matrix of synaptic weights, W_1^* is *n x* L_0 matrix of synaptic weights.

S(x) is an L-dia- mensional vector with elements:

$$s_i(x) = \prod_{j \in Ii} \left[s(x_j) \right]^{d_j(i)} \tag{3.4}$$

where I_i , *i=1,2,…,L* are a collection of *L* subsets of $\{1,2,...,n\}$ and d_j are non-negative integers.

S'(x) is L_0*xm* matrix with elements:

$$s'_{lk}(x) = \prod_{j \in I_{lk}} \left[s(x_j) \right]^{d_j(l,k)} \tag{3.5}$$

where l=1,2,…, L_0 , k=1,2,…,m , I_{lk} are collections of mL_0 subsets of $\{1,2,...,n\}$ and d_j are non-negative integers.

$s(x_j)$ is a sigmoid function:

$$s(x_j) = \frac{\mu}{1+e^{-l_0 x_i}} + \lambda$$

where μ, l_0 represent the bound and slope of sigmoid's and λ is a bias constant.

We consider h as a function of class C^2 from $R^n x R^m x R^{nxL} to R^+$ whose derivative is:

$\dot{h} = \frac{\partial h^T}{\partial x}\left[-Ax + W^* S(x) + W_1^* S'(x)u + \omega_0(x,u)\right] + \frac{\partial h^T}{\partial u}\dot{u} + \frac{\partial h^T}{\partial W}\dot{W}$. This equation can also be written as:

$$\dot{h} + \frac{\partial h^T}{\partial x}Ax - \frac{\partial h^T}{\partial x}\omega_0(x,u) = \frac{\partial h^T}{\partial x}W^* S(x) + \frac{\partial h^T}{\partial x}W_1^* S'(x)u + \frac{\partial h^T}{\partial u}\dot{u} + \frac{\partial h^T}{\partial W}\dot{W} \qquad (3.6)$$

We define:

$$\upsilon = \frac{\partial h^T}{\partial x}WS(x) + \frac{\partial h^T}{\partial x}W_1^* S'(x)u - \dot{h} - \frac{\partial h^T}{\partial x}Ax + \frac{\partial h^T}{\partial u}\dot{u} + \frac{\partial h^T}{\partial W}\dot{W} - I_n tr\{\Lambda^T W\},$$

where $\dot{h}$ is unknown, thus we use the filtered version of υ:

$$\dot{\xi} + k\xi = \upsilon,$$

$$= -\dot{h} + \frac{\partial h^T}{\partial x}\left[-Ax + WS(x) + W_1\ S'(x)u\right] + \frac{\partial h^T}{\partial u}\dot{u} + \frac{\partial h^T}{\partial W}\dot{W} - I_n tr\{\Lambda^T W\} \qquad (3.7)$$

Lemma 3.3.1

The control law is the following:

$$\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x) \qquad (3.8)$$

together with the update law:

$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in W or \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \geq 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases} \quad (3.9)$$

and

$$\frac{1}{2}kI = aI = A,$$

guarantees that

$\zeta(t) \leq 0, \forall t \geq 0$

$\lim_{t\to\infty} \zeta(t) = 0$ exponentially fast provided that *ζ(0)* <0.

Furthermore $\|W\| \leq w_m, \forall t > 0$ provided that $W(0) \in W$ and $W^* \in W$.

Proof [5 (chapter 4)]

For the first relation we have that: $\dot{\zeta} = -k\zeta, \forall t \geq 0$, consequently:

$$\zeta(t) = \zeta(0)e^{-kt}, \forall t \geq 0.$$

Since *k*>0, if we take $\zeta(0) < 0$, we have $\zeta(t) \leq 0, \forall t \geq 0$ and $\zeta(t)$ converges to zero exponentially fast with rate *k*.

For the second relation we will prove that $\frac{d}{dt}\left(\|W(t)\|\right)^2 \leq 0$ whenever $\|W(t)\| = w_m$, which means that the weights *W* are directed towards the inside of the ball $\{W : \|W\| \leq w_m\}$.

It is true that:

$$\frac{d}{dt}\left(\|W(t)\|\right)^2 = \frac{d}{dt}\left(tr\{W^T W\}\right) = tr\{\dot{W}W^T\}$$

Using the adaptive law (3.9) we have:

$$W^T\dot{W} = -e\frac{1}{2}W^T kW - eW^T xS^T(x) + tr\left\{e\frac{1}{2}\ kW + exS^T(x)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W^T W$$

Consequently:

$$tr\{W^T\dot{W}\} = tr\left\{-e\frac{1}{2}W^TkW - eW^TxS^T(x) + tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W^TW\right\},$$

$$= tr\left\{-e\frac{1}{2}W^TkW - eW^TxS^T(x)\right\} + tr\left\{tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W^TW\right\},$$

$$= tr\left\{-e\frac{1}{2}W^TkW - eW^TxS^T(x)\right\} + tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 tr\{W^TW\}$$

Also it is true that:

$$tr\left\{-e\frac{1}{2}W^TkW - eW^TxS^T(x)\right\} = tr\left\{\left(-e\frac{1}{2}kW - exS^T(x)\right)W^T\right\}.$$

Therefore:

$$tr\{W^T\dot{W}\} \le tr\left\{\left(-e\frac{1}{2}kW - exS^T(x)\right)W^T\right\} + tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2\|W\|^2,$$

$$\le tr\left\{\left(-\left(e\frac{1}{2}kW + exS^T(x)\right)\right)W^T\right\} + tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\}(1+w_m)^2$$

because we have considered $\|W\| = w_m$.

Finally, because $tr\left\{\left(e\frac{1}{2}\ kW + exS^T(x)\right)W^T\right\} < 0$ and $(1+w_m)^2 > 1$ we have:

$$\frac{d}{dt}\left(\|W(t)\|\right)^2 \le 0.$$

We consider the Lyapunov-like function:

$L = \frac{1}{2}\xi^2 + \frac{1}{2}tr\{\tilde{W}_1^T\tilde{W}_1\}$, where $\tilde{W}_1 = W_1 - W_1^*$.

Taking the derivative of *L* we have:

$$\dot{L} = \xi\dot{\xi} + tr\{\dot{W}_1\tilde{W}_1\},$$

or after using (3.7):

$$\dot{L} = -k\xi^2 + \xi\left[-\dot{h} - x^TAx + x^TWS(x) + x^TW_1S'(x)u + \dot{u}^Tu\right] +$$
$$+ \xi\left[tr\{(\dot{W}^T - I_n\Lambda^T)W\}\right] + tr\{\dot{W}_1\tilde{W}_1\},$$

which combined with (3.6) is:

$$\dot{L} = -k\xi^2 + \xi\left[-x^T W^* S(x) - x^T W_1^* S'(x)u + x^T W S(x)\right] +$$
$$+ \xi\left[x^T W_1\ S'(x)u - x^T \omega_0(x,u) + tr\left\{-I_n \Lambda^{\mathrm{T}} W\right\}\right] + tr\left\{\dot{W}_1 \tilde{W}_1\right\},$$

or

$\dot{L} = -k\xi^2 + \xi x^T \tilde{W} S(x) + \xi x^T \tilde{W}_1 S'(x)u - \xi x^T \omega_0(x,u) + \xi tr\left\{-I_n \Lambda^{\mathrm{T}} W\right\} + tr\left\{\dot{W}_1 \tilde{W}_1\right\}$, where $\tilde{W}_1 = W_1 - W_1^*$.

Moreover from the update law $\dot{W}_1 = -\xi x\left(S'(x)u\right)^T$ we have:

$$\dot{L} = -k\xi^2 + \xi x^T \tilde{W} S(x) - \xi x^T \omega_0(x,u) - \xi tr\left\{I_n \Lambda^{\mathrm{T}} W\right\}.$$

We need the following lemma:

Lemma 3.3.2

The update law (3.9) guarantee that: $tr\left\{I_n \Lambda^{\mathrm{T}} W\right\} \le 0$.

Proof[5 (chapter 4)]

It is true that:

$I_n \Lambda^{\mathrm{T}} W = I_n tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W^T W$, hence

$tr\left\{I_n \Lambda^{\mathrm{T}} W\right\} \le I_n tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 tr\left\{W^T W\right\},$

$= I_n tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 \|W\|^2,$

$\le 0,$

because if $I_n = 1$, we have that $tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0$ when

$\left(\frac{1+\|W\|}{w_m}\right)^2 \|W\|^2 \ge 0, \forall t \ge 0$. ■

Lemma 3.3.3

For the system under examination there are the following properties:

$$\xi(t) \le 0, \forall t \ge 0$$
$$|x(t)|^2 \le 4|\xi(t)|, \forall t \ge 0$$
$$|u(t)|^2 \le 4|\xi(t)|, \forall t \ge 0$$

Proof [5 (chapter 4)]

We have that: $h = \zeta - \xi$. $h \ge 0, \forall t \ge 0$ consequently: $\zeta(t) \ge \xi(t), \forall t \ge 0.$

Moreover from Lemma 3.3.1: $\zeta(t) \le 0, \forall t \ge 0$, thus $\xi(t) \le 0, \forall t \ge 0.$ Equivalently:

$|\zeta(t)| \le |\xi(t)|, \forall t \ge 0.$

We have chosen $h = \frac{1}{2}\left(|x|^2 + |u|^2 + tr\{W^T W\}\right) \ge \frac{1}{2}|x|^2$ (3.10)

Furthermore:

$$h \le |\zeta(t)| + |\xi(t)|,$$
$$h \le 2|\xi(t)| \quad (3.11)$$

From (3.10) and (3.11) we have:

$|x(t)|^2 \le 4|\xi(t)|, \forall t \ge 0$.

Moreover since: $h \ge \frac{1}{2}|u|^2$, it is true that: $|u(t)|^2 \le 4|\xi(t)|, \forall t \ge 0$. ■

Lemmas 3.3.2 and 3.3.3 prove that $I_n \xi tr\{\Lambda^T W\} \ge 0, \forall t \ge 0$.

Thus: $\dot{L} \le -k\xi^2 + \xi x^T \tilde{W} S(x) - \xi x^T \omega_0(x,u)$ (3.12)

Assumption 3.3.1

The modeling error at zero case term satisfies:

$$|\omega_0(x,u)| \le k_1 + k_2|x| \quad (3.13)$$

where k_1, k_2 are known positive constants.

The relation (3.13) means that the modelling error term satisfies the Lipschitz condition which guarantees the existence and uniqueness of solutions of (3.3). Furthermore k_1, k_2 it is not suggested take large values because the error in Theorem 2.1.1 can be considered relatively small.

Generally the uncertainties are considered to be bounded by known functions. The design of robust methods which can model systems of unknown uncertainties is not examined here.

It is true that:

$$\left\|\tilde{W}\right\| \le \overline{w}_m, \tag{3.14}$$

by using the projection algorithm (3.9) and also

$$\left|S(x)\right| \le k_0, \tag{3.15}$$

by definition of *S(x),* where $\overline{w}_m, k_0$ are known positive constants. From (3.14), (3.15) and (3.12) we have: $\dot{L} \le -k\left|\xi\right|^2 + \left|\xi\right|\left|x\right| \ \overline{w}_m k_0 - \xi x^T \omega_0(x,u)$.

According to Assumption 3.3.1 and Lemma 3.3.3 we have that:

$$\dot{L} \le -k\left|\xi\right|^2 + 2\overline{w}_m k_0 \left|\xi\right|\sqrt{\left|\xi\right|} \ + \left|\xi\right|\left|x\right|\left|\omega_0(x,u)\right|,$$

$$\le -k\left|\xi\right|^2 + 2\overline{w}_m k_0 \left|\xi\right|\sqrt{\left|\xi\right|} \ + 2k_1\left|\xi\right|\sqrt{\left|\xi\right|} + 4k_2\left|\xi\right|^2,$$

$$-\left[\left(k-4k_2\right)\sqrt{\left|\xi\right|} - 2k_1 - 2\overline{w}_m k_0\right]\left|\xi\right|\sqrt{\left|\xi\right|} \ ,$$

$$\le 0$$

provided that $\sqrt{\left|\xi\right|} \ > \frac{2\left(k_1 + k_0\overline{w}_m\right)}{k-4k_2}$ and $k > 4k_2 > 0$.

The aforementioned analysis combined with Lemma 3.3.3 proves that the signals *ξ(t)* and *x(t)* are uniformly ultimately bounded with respect to the sets:

$$\Xi = \left\{\xi(t) : \left|\xi(t)\right| \le \frac{4\left(k_1 + k_0\overline{w}_m\right)^2}{\left(k-4k_2\right)^2}\right\}$$

$$X = \left\{x(t) : \left|x(t)\right| \le \frac{4\left(k_1 + k_0\overline{w}_m\right)}{k-4k_2}\right\}$$

These sets can be confined because k can be chosen large. If *ξ(t)* is inside *Ξ* then it is bounded by $\frac{4\left(k_1 + k_0\overline{w}_m\right)^2}{\left(k-4k_2\right)^2}$. Alternatively, if it is outside *Ξ,* then $\dot{L} \le 0$, thus *ξ* is driven to the boundary of *Ξ*.

Theorem 3.3.1

Consider the system (3.3) with the modeling error term satisfying Assumption 3.3.1. Then the dynamic compensator

$$\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x)$$

together with

$$\dot{\zeta} = -k\zeta$$
$$h = \zeta - \xi$$
$$h = \frac{1}{2}\left(|x|^2 + |u|^2 + tr\{W^T W\}\right)$$
$$\Lambda = tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W$$

and the update laws

$$\dot{W}_1 = -\xi x\left(S'(x)u\right)^T,$$
$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in W or \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \geq 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases}$$

guarantee the uniform ultimate boundness of the trajectories of *ξ(t)* and *x(t)* with respect to the arbitrarily small sets *Ξ* and *X*

$$\Xi = \left\{\xi(t) : |\xi(t)| \leq \frac{4\left(k_1 + k_0\overline{w}_m\right)^2}{\left(k - 4k_2\right)^2}\right\}$$

$$X = \left\{x(t) : |x(t)| \leq \frac{4\left(k_1 + k_0\overline{w}_m\right)}{k - 4k_2}\right\}$$

provided that $k>4k_2>0$.

The afore-mentioned theorem guarantees the boundness of weights *W*. About W_1, it can cause instability, thus we can use the following projection algorithm in order to guarantee its boundness:

$$\dot{W}_1 = \begin{cases} -\xi x\left(S'(x)u\right)^T, if W_1 \in W_1 or \|W_1\| = \overline{w} and tr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} \geq 0 \\ -\xi x\left(S'(x)u\right)^T + P, if \|W_1\| = \overline{w} and tr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} < 0 \end{cases} \quad (3.16)$$

where $P = tr\left\{\xi x(S'(x)u)^T W_1^T\right\}\left(\frac{1+\|W_1\|}{\overline{w}}\right)^2 W_1$.

This projection algorithm restricts W_1 in the set: $W_1 = \{W_1 : \|W_1\| \leq \overline{w}\}$. Consequently $\|W_1\| \leq \overline{w}$, provided that we take $\|W_1(0)\| \leq \overline{w}$ and $W_1^* \in W_1$. Particularly, if W_1 is in the interior of W_1 or tends to move inward from the bound, then $\dot{W}_1$ is unchanged. Else there is modification of $\dot{W}_1$.

As we have already discussed, there are various modification algorithms which can be used. These algorithms ensure the boundness of the weights estimates and guarantee the stability in the presence of modeling errors or disturbances.

Another important matter is the appropriate choice of the design constant *k*. A high design constant causes high model inaccuracy. In addition, *k* is a gain in the construction of ζ. Therefore there is a compromise and an appropriate value of *k* is chosen.

3.3.2 No modeling error at zero case

In this paragraph we consider the case of no modeling error at zero case, in order to prove the convergence of the state to zero.

Assumption 3.3.2

The modelling error term satisfies the relation:

$$|\omega_0(x,u)| \leq k_2|x| \quad (3.17)$$

Theorem 3.3.2

Consider the system:

$$\dot{x} = -Ax + W^* S(x) + W_1^* S'(x)u + \omega_0(x,u)$$

$$\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x)$$

$$\dot{\zeta} = -k\zeta$$
$$h = \zeta - \xi$$
$$h = \frac{1}{2}\left(|x|^2 + |u|^2 + tr\{W^T W\}\right)$$
$$\Lambda = tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W$$

and the update laws

$$\dot{W}_1 = -\xi x\left(S'(x)u\right)^T,$$
$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in W or \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \geq 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases}$$

guarantee the following properties

$$\xi, \dot{\xi}, |x|, |u|, W, W_1, \zeta \in L_\infty, |\xi| \in L_2$$
$$\lim_{t\to\infty} \xi(t) = 0, \lim_{t\to\infty} |x(t)| = 0,$$
$$\lim_{t\to\infty} \|W(t)\| = 0, \lim_{t\to\infty} |u(t)| = 0,$$
$$\lim_{t\to\infty} \dot{W}_1(t) = 0$$

provided that $k > 4\left(k_2 + k_0\overline{w}_m\right)$ and $S(x) \leq k_0|x|$.

Proof [5 (chapter 4)]

With respect to the previous analysis about *L* we have:

$$\begin{aligned} \dot{L} &\leq -k|\xi|^2 + 4\overline{w}_m k_0 |\xi|^2 + 4k_2 |\xi|^2, \\ &- \left[\left(k - 4k_2 - 4\overline{w}_m k_0\right)\sqrt{|\xi|}\right]\xi|^2, \\ &\leq 0, \end{aligned} \qquad (3.18)$$

given that the design constant k satisfies: $k > 4(k_2 + k_0\overline{w}_m)$ and $S(x) \leq k_0|x|$.

From (3.18) we infer that $L \in L_\infty$. Consequently $\xi, \tilde{W}_1 \in L_\infty$.

Since $W_1 = \tilde{W}_1 + W_1^*$, $W_1 \in L_\infty$.

Furthermore $\zeta(t) \le 0, \forall t \ge 0$, which combined with $\xi = \zeta - h$ proves that $\zeta, h \in L_\infty$. Therefore $|x|, |u|, \|W\| \in L_\infty$.

Since *L* is a monotone decreasing function of time and bounded from below, $\lim_{t\to\infty} L(t) = L_\infty$ exists. Consequently we have:

$$\int_0^\infty |\xi|^2 dt \le \frac{1}{k - 4(k_2 + k_0 \overline{w}_m)} [L(0) - L_\infty] < \infty$$

which means that $|\xi| \in L_2$.

It is also true that:

$$\dot{\xi} = -k\xi + x^T \tilde{W} S(x) + x^T \tilde{W}_1 S'(x) u - x^T \omega_0(x,u) - I_n tr\{\Lambda^T W\}$$

which means that $\dot{\xi} \in L_\infty$ because $|x|, u, \tilde{W}, \tilde{W}_1 \in L_\infty$, *S,S'* are bounded by definition and (3.17) is true. Applying Barbalat's Lemma we can prove that $\lim_{t\to\infty} \xi(t) = 0$.

As we know *h=ζ-ξ*, and since *ζ,ξ* converge we infer that also *h* converges because:

$$\lim_{t\to\infty} h(x,u,W) = \lim_{t\to\infty} \zeta(t) - \lim_{t\to\infty} \xi(t) = 0 \qquad (3.19)$$

From (3.10) and (3.19) we finally infer that:

$$\lim_{t\to\infty} |x(t)| = 0,$$
$$\lim_{t\to\infty} |u(t)| = 0,$$
$$\lim_{t\to\infty} \|W(t)\| = 0$$

Respectively, $\lim_{t\to\infty} \dot{W}_1 = 0$ since *ξ* converges to zero and *x,S',u* are bounded.

■

According to this method RHONN and actual system have the same origin, as Assumption 3.3.2 requires. The aforementioned Theorem 3.3.2 proves that in the particular case the control scheme maintains its stability properties.

3.3.3 No modeling error case

In this paragraph we examine the case of no modeling error.

Assumption 3.3.3

The modelling error term satisfies: $|\omega_0(x,u)| = 0$.

Theorem 3.3.3

Consider the system:

$$\dot{x} = -Ax + W^* S(x) + W_1^* S'(x)u$$

$$\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x)$$

$$\dot{\zeta} = -k\zeta$$
$$h = \zeta - \xi$$
$$h = \frac{1}{2}\left(|x|^2 + |u|^2 + tr\{W^T W\}\right)$$
$$\Lambda = tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W$$

and the update laws

$$\dot{W}_1 = -\xi x\left(S'(x)u\right)^T,$$
$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in W or \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \geq 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases}$$

guarantee the following properties

$$\xi, \dot{\xi}, |x|, |u|, W, W_1, \zeta \in L_\infty, |\xi| \in L_2$$
$$\lim_{t\to\infty} \xi(t) = 0, \lim_{t\to\infty} |x(t)| = 0,$$
$$\lim_{t\to\infty} \|W(t)\| = 0, \lim_{t\to\infty} |u(t)| = 0,$$
$$\lim_{t\to\infty} \dot{W}_1(t) = 0$$

provided that $k > 4k_0\overline{w}_m$ and $S(x) \leq k_0|x|$.

Proof[5 (chapter 4)]

With respect to the previous analysis about *L* and Assumption 3.3.3 we have:

$$\begin{aligned}\dot{L} &\le -k|\xi|^2 + 4\overline{w}_m k_0 |\xi|^2 \ , \\ &\le -\left[(k - 4\overline{w}_m k_0)\sqrt{|\xi|}\right]\xi|^2 \ , \\ &\le 0\end{aligned} \tag{3.20}$$

given that *k* satisfies: $k > 4k_0\overline{w}_m$ and $S(x) \le k_0|x|$.

From (3.20) we infer that $L \in L_\infty$. Consequently $\xi, \tilde{W}_1 \in L_\infty$. Moreover $W_1 = \tilde{W}_1 + W_1^*$, therefore $W_1 \in L_\infty$.

Furthermore $\zeta(t) \le 0, \forall t \ge 0$, which combined with $\xi = \zeta - h$ proves that $\zeta, h \in L_\infty$. Therefore $|x|, |u|, \|W\| \in L_\infty$.

Since *L* is a monotone decreasing function of time and bounded from below, $\lim_{t\to\infty} L(t) = L_\infty$ exists. Consequently we have:

$$\int_0^\infty |\xi|^2 dt \le \frac{1}{k - 4(k_2 + k_0\overline{w}_m)}\left[L(0) - L_\infty\right] < \infty$$

which means that $|\xi| \in L_2$.

It is also true that:

$$\dot{\xi} = -k\xi + x^T\tilde{W}S(x) + x^T\tilde{W}_1 S'(x)u - I_n tr\left\{\Lambda^T W\right\}$$

which means that $\dot{\xi} \in L_\infty$ because $|x|, u, \tilde{W}, \tilde{W}_1 \in L_\infty$ and *S,S'* are bounded by definition. Applying Barbalat's Lemma we can prove that $\lim_{t\to\infty} \xi(t) = 0$. As we know $h=\zeta-\xi$, and since ζ,ξ converge we infer that also *h* converges because:

$$\begin{aligned}&\lim_{t\to\infty} h(x,u,W) = \lim_{t\to\infty} \zeta(t) - \lim_{t\to\infty} \xi(t) \\ &= 0\end{aligned} \tag{3.21}$$

From (3.10) and (3.21) we finally infer that:

$$\lim_{t\to\infty}|x(t)| = 0,$$
$$\lim_{t\to\infty}|u(t)| = 0,$$
$$\lim_{t\to\infty}\|W(t)\| = 0$$

Respectively, $\lim_{t\to\infty}\dot{W}_1 = 0$ since ξ converges to zero and x, S', u are bounded.

■

In conclusion, the direct adaptive regulation method under modeling errors *(n!=m)* is described at Figure 3.1.

Actual system: $\dot{x} = f(x) + G(x)u, x \in R^n, u \in R^m$

Model: $\dot{x} = -Ax + W^* S(x) + W_1^* S'(x)u + \omega_0(x,u)$

Control Law: $\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x)$

Update Laws:

$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in Wor \|W\| = w_m andtr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \geq 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m andtr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases}$$

$$\dot{W}_1 = \begin{cases} -\xi x\left(S'(x)u\right)^T, if W_1 \in W_1 or \|W_1\| = \overline{w} andtr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} \geq 0 \\ -\xi x\left(S'(x)u\right)^T + P, if \|W_1\| = \overline{w} andtr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} < 0 \end{cases}$$

$$\Lambda = tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W$$

$$P = tr\left\{\xi x(S'(x)u)^T W_1^T\right\}\left(\frac{1+\|W_1\|}{\overline{w}}\right)^2 W_1$$

Filter:

$$\dot{\zeta} = -k\zeta$$
$$\xi = \zeta - h$$
$$h = \frac{1}{2}\left(|x|^2 + |u|^2 + tr\left\{W^T W\right\}\right)$$

Results:

$$W_1 \in W', u, \dot{\xi} \in L_\infty,$$
$$W \in W, |\zeta(t)| \leq |\zeta(0)|,$$

ξ, x are u.u.b. with respect to

$$\Xi = \left\{\xi(t) : |\xi(t)| \leq \frac{4\left(k_1 + k_0 \overline{w}_m\right)^2}{\left(k - 4k_2\right)^2}\right\},$$
$$X = \left\{x(t) : |x(t)| \leq \frac{4\left(k_1 + k_0 \overline{w}_m\right)}{k - 4k_2}\right\}$$

Requirements:

$$W_1^* \in W', W_1(0) \in W'$$
$$W^* \in W, W(0) \in W,$$
$$|\omega(x,u)| \leq k_1 + k_2|x|, \zeta(0) \leq 0$$

Fig. 3.1: Direct Adaptive Regulation under modeling errors(n!=m)

CHAPTER 4
ENZYME KINETICS REACTIONS

4.1 Michaelis-Menten mechanism

The Michaelis-Menten mechanism for basic one-substrate enzyme reactions consists of three elementary reactions steps, as we can see in the scheme (4.1). This is the basic scheme of an enzymatic reaction. Subsequently, signal transduction pathways can be represented as sequences of this basic reaction's scheme. These enzyme kinetics reactions turn a substrate *S* into a product *P* via an intermediate complex *ES* and are regulated by an enzyme *E*.

$$S + E \underset{k_2}{\overset{k_1}{\Leftrightarrow}} ES \xrightarrow{k_3} E + P \tag{4.1}$$

k_1 is the rate by which the enzyme-substrate complex *ES* is formed. The complex *ES* can be dissociated into *E* and *S* with a rate k_2 or it can proceed to form a product *P* with a rate k_3.

Generally the velocity of reaction is the following:

$$V = k_3[ES] \tag{4.2}$$

Under steady-state conditions, where concentration of intermediate complex is constant, we have:

$$\frac{d[ES]}{dt} = k_1[S][E] - k_2[ES] - k_3[ES] = 0,$$
$$k_1[S][E] = k_2[ES] + k_3[ES],$$
$$k_1[S]([E]_0 - [ES]) = (k_2 + k_3)[ES],$$
$$k_1[S][E]_0 = k_1[S][ES] + (k_2 + k_3)[ES],$$
$$k_1[S][E]_0 = [ES][k_1[S] + (k_2 + k_3)],$$
$$[ES] = \frac{k_1[S][E]_0}{k_1[S] + (k_2 + k_3)},$$
$$[ES] = \frac{[S][E]_0}{[S] + \frac{k_2 + k_3}{k_1}}$$

(4.3)

where according to the mass balance:

$$[E]_0 = [E] + [ES] \tag{4.4}$$

Substitute equation (4.3) in (4.2) gives:

$$V = k_3 \frac{[S][E]_0}{[S] + \frac{k_2 + k_3}{k_1}},$$
$$V = \frac{k_3[S][E]_0}{[S] + k_m},$$
$$V = \frac{V_{\max}[S]}{[S] + k_m}$$

where V is the velocity of reaction, V_{max} is the velocity of reaction when the enzyme is completely saturated with substrate and K_m is the Michaelis-Menten constant [2].

A steady state situation develops with regard to [*ES*] and remains until almost all of the substrate is consumed, as we can see in Figure 4.1. Furthermore, the algebraic condition (4.4) is true for the total enzyme concentration.

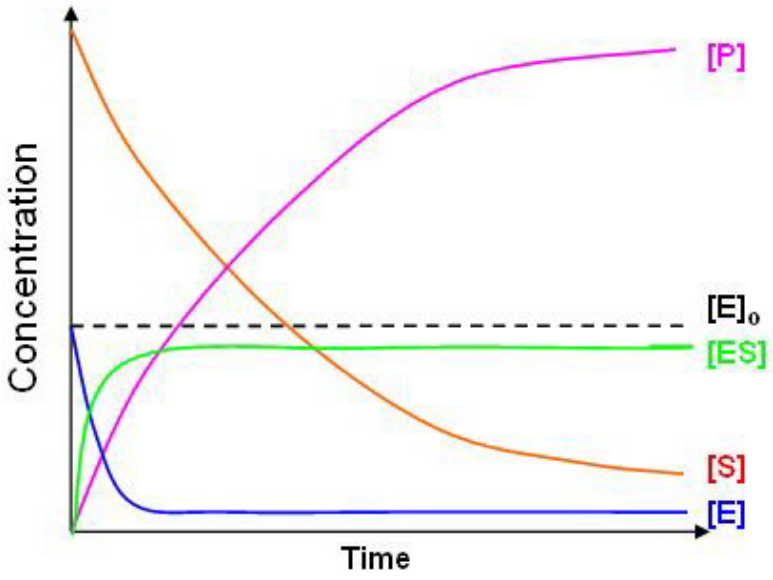

Fig. 4.1: Graphs of concentrations of Substrate (S), Enzyme (E), Enzyme-Substrate (ES)and Product (P) of enzymatic reaction (Michaelis-Menten)

In this project our aim is the regulation of enzyme activity by changing the rate constants. We concentrate on the basic pathway of enzyme reactions (4.1).

4.2 Regulating enzyme activity

There are various methods of regulation of enzyme activity. The most known methods are:

- Change of temperature
- Change of pH
- Allosteric regulation
- Enzyme availability (synthesis, degradation, localization)
- Substrate availability (synthesis, degradation, localization)
- Inhibition:
  - By specific metabolites within the cell
  - By drugs, toxins, etc.

– By specific analogues in study of reaction mechanism

4.3 Mathematical description of enzyme kinetics reactions

The enzyme kinetics reactions are based on the scheme (4.1). According to this scheme we have the signal transduction pathway of Figure 4.2, where $m_1 = [S], m_2 = [E], m_3 = [ES], m_4 = [P]$. This pathway can be mathematically described by the following set of nonlinear ordinary differential equations:

$$\frac{dm_1(t)}{dt} = -k_1 m_1(t) m_2(t) + k_2 m_3(t)$$
$$\frac{dm_2(t)}{dt} = -k_1 m_1(t) m_2(t) + k_2 m_3(t) + k_3 m_3(t)$$
$$\frac{dm_3(t)}{dt} = k_1 m_1(t) m_2(t) - k_2 m_3(t) - k_3 m_3(t)$$
$$\frac{dm_4(t)}{dt} = k_3 m_3(t)$$

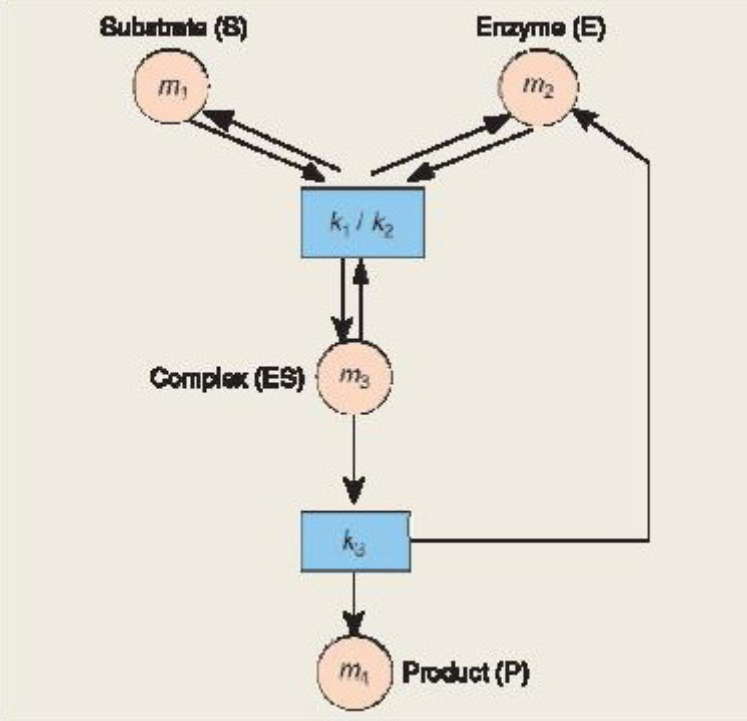

Fig. 4.2: Signal transduction pathway of basic enzyme kinetics reaction[1]
(Reproduced by permission from Mr O. Wolkenhauer)

The system of these equations describes transient time courses for the concentrations of substrate *[S],* Enzyme *[E]* , substrate-enzyme complex *[ES]* and product *[P].* These time courses are initiated by a set of initial concentrations.

4.4 Regulation of enzyme activity by temperature

4.4.1 Temperature effects on the activity of enzymes

In section 4.2 we referred to various factors which regulate enzyme activity. Temperature affects the activity of enzymes because it changes the three-dimensional structure of the enzyme. In an uncatalyzed reaction the rate increases in direct proportion to the temperature. But, in catalyzed reactions

the effect of temperature is quite different. Starting at a low temperature, and increasing the temperature initially causes increase of the rate. However, once the peak temperature has been reached the rate of the reaction begins to decrease. Generally, enzymes become inactive in high temperatures (>40- 50 ^{o}C). Above those temperatures there is a critical point where the enzyme becomes inactive and undergo denaturation. In the Figure 4.3 we can see the reactivity of an enzyme reaction against temperature.

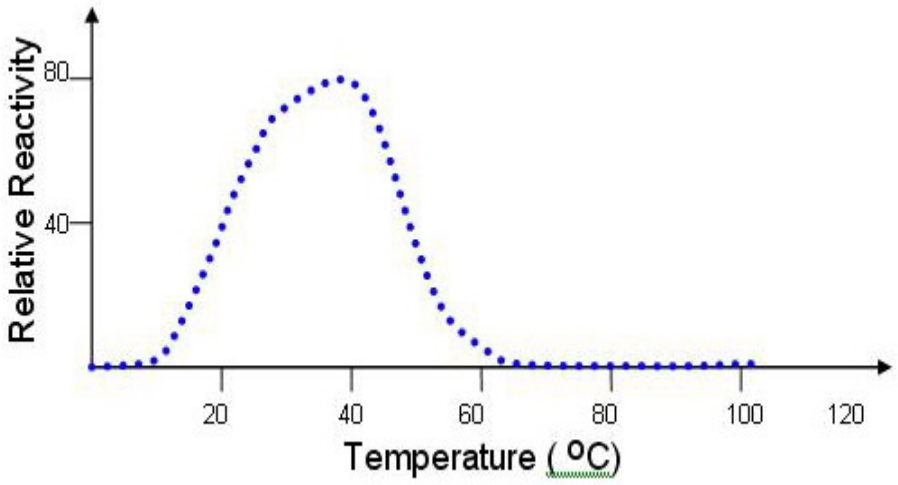

Fig. 4.3: Reactivity of an enzyme(Human Amylase Starch Glucose) against temperature. There is a critical point(almost 40 ^{o}C) where the enzyme becomes denaturated.

4.4.2 Arrhenius equation

The thermal variation of rate constants follows the following equation, known as Arrhenius equation:

$$K=Ae^{-E/RT} \quad (4.5)$$

where A is the frequency factor(or Arrhenius constant), E the activation energy, R the gas constant and T the absolute temperature. Arrhenius equation shows the effect of a change of temperature on the rate constant and therefore on the rate of the reaction. Generally, rates of reactions increase with the temperature.

In the case of our signal transduction pathway (Michaelis-Menten mechanism) each of the three rate constants is described by an Arrhenius equation:

$$K_1=A_1e^{-E1/RT} \tag{4.6}$$

$$K_2=A_2e^{-E2/RT} \tag{4.7}$$

$$K_3=A_3e^{-E3/RT} \tag{4.8}$$

CHAPTER 5
IMPLEMENTATION OF DIRECT ADAPTIVE METHOD

5.1 Introduction of the method

In this project we implement direct adaptive regulation using RHONN (Recurrent High-Order Neural Networks). RHONNs are used as models of the unknown plant, transforming the original system into a RHONN model which is of known structure, but contains a number of unknown constant-value parameters, known as synaptic weights. RHONNs were introduced in the previous Chapters. As we have already discussed the particular method refers to affine in the control nonlinear dynamical systems possessing unknown non-linearities.

A system affine in the control has the form:

$$\dot{x}(t) = f(x) + G(x)u$$

The control input, which is the absolute temperature in our case, has exponential dependence. As a result, a linear in control system is adopted to approximate the nonlinear system. Based on this equivalent linear in control system, a controller for the original nonlinear system can be constructed.

Taylor series expansion method is appropriate and is employed to transfer our system into the standard affine form in a region of input *T*.

5.2 Taylor series expansion

For a function f, we can write the Taylor series about a point a as

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \ldots \qquad (5.1)$$

This series is exact if we use an infinite number of terms. If we use only the first 3 terms we get a Taylor series of order 2 which is accurate near the point $x = a$.

The order of Taylor series depends on the magnitude of the area *(x-α)* and the accuracy required.

Generally the error is small when the area of expansion *(x-α)* is small and only a few terms are needed for good accuracy. The area of interest is defined by the requirements.

In our case we need a linear approximation so we use the first two terms and omit the remaining higher order terms. This is a linearization around a particular point. We choose that point to be in the middle of the area of interest.

5.3 Description of the System

Our system is a nonlinear dynamical system and has 4 states according to the state equations. Input is the absolute temperature (*T*). This system is of the general form:

$$\dot{x} = f(x,u)$$

Furthermore there is the following algebraic condition about enzyme concentration according to (4.4):

$$m_2 + m_3 = c_1 \tag{5.2}$$

Consequently the third state equation can be omitted and the system has 3 states and 1 input:

$$\frac{dm_1(t)}{dt} = -k_1 m_1(t) m_2(t) + k_2(c_1 - m_2(t))$$
$$\frac{dm_2(t)}{dt} = -k_1 m_1(t) m_2(t) + k_2(c_1 - m_2(t)) + k_3(c_1 - m_2(t))$$
$$\frac{dm_4(t)}{dt} = k_3(c_1 - m_2(t))$$

If we apply Arrhenius equation for k_1, k_2, k_3 ($K=Ae^{-E/RT}$) we have the dependence of the rate constants on the control input T.

Our system is, as we know, nonaffine in control. The linear approximation is implemented by the use of Taylor series expansion as was introduced in the previous subsection.

The range of input is chosen to be 8K and the area of interest is 296-304K. This range is adequate for our purpose and the mean error of this approximation is about 1-2% which is considered tolerable.

The expansion point, as we said, is in the middle of the space: Te=300.

Finally, the system in affine form for an area of input *T* is the following:

$$\frac{dm_1}{dt} = -A_1 e^{\frac{-E1}{RTe}}(1-\frac{E_1}{RTe})m_1 m_2 + A_2 e^{\frac{-E2}{RTe}}(1-\frac{E_2}{RTe})(c_1 - m_2) +$$
$$+\left[-A_1 e^{\frac{-E1}{RTe}}\frac{E_1}{RTe^2}m_1 m_2 + A_2 e^{\frac{-E2}{RTe}}\frac{E_2}{RTe^2}(c_1 - m_2)\right]T$$
$$\frac{dm_2}{dt} = -A_1 e^{\frac{-E1}{RTe}}(1-\frac{E_1}{RTe})m_1 m_2 + \left[A_2 e^{\frac{-E2}{RTe}}(1-\frac{E_2}{RTe}) + A_3 e^{\frac{-E3}{RTe}}(1-\frac{E_3}{RTe})\right](c_1 - m_2) +$$
$$+\left[-A_1 e^{\frac{-E1}{RTe}}\frac{E_1}{RTe^2}m_1 m_2 + \left(A_2 e^{\frac{-E2}{RTe}}\frac{E_2}{RTe^2} + A_3 e^{\frac{-E3}{RTe}}\frac{E_3}{RTe^2}\right)(c_1 - m_2)\right]T$$
$$\frac{dm_4}{dt} = A_3 e^{\frac{-E3}{RTe}}(1-\frac{E_3}{RTe})(c_1 - m_2) + \left[A_3 e^{\frac{-E3}{RTe}}\frac{E_3}{RTe^2}(c_1 - m_2)\right]T$$

The state equations can be written in the form $\dot{x}(t) = f(x) + G(x)u$, where the states are the concentrations of the substrate m_1, enzyme m_2 and product m_4 , therefore the state vector: $x=[m_1\ m_2\ m_4]^T$.

As control input we have the absolute temperature $u=T$.

Thus, the matrices $f(x)$ and $G(x)$ are:

$$f(x) = \begin{bmatrix} -A_1 e^{\frac{-E_1}{RTe}}(1-\frac{E_1}{RTe})m_1 m_2 + A_2 e^{\frac{-E_2}{RTe}}(1-\frac{E_2}{RTe})(c_1 - m_2) \\ -A_1 e^{\frac{-E_1}{RTe}}(1-\frac{E_1}{RTe})m_1 m_2 + \left[A_2 e^{\frac{-E_2}{RTe}}(1-\frac{E_2}{RTe}) + A_3 e^{\frac{-E_3}{RTe}}(1-\frac{E_3}{RTe})\right](c_1 - m_2) \\ A_3 e^{\frac{-E_3}{RTe}}(1-\frac{E_3}{RTe})(c_1 - m_2) \end{bmatrix}$$

$$G(x)=\begin{bmatrix} -A_1 e^{\frac{-E_1}{RTe}}\frac{E_1}{RTe^2}m_1m_2 + A_2 e^{\frac{-E_2}{RTe}}\frac{E_2}{RTe^2}(c_1-m_2) \\ -A_1 e^{\frac{-E_1}{RTe}}\frac{E_1}{RTe^2}m_1m_2 + \left(A_2 e^{\frac{-E_2}{RTe}}\frac{E_2}{RTe^2} + A_3 e^{\frac{-E_3}{RTe}}\frac{E_3}{RTe^2}\right)(c_1-m_2) \\ A_3 e^{\frac{-E_3}{RTe}}\frac{E_3}{RTe^2}(c_1-m_2) \end{bmatrix}$$

5.4 Direct Adaptive Regulation under modeling error method

Direct adaptive regulation under modeling error at zero case is described in Figure 3.1 and has been introduced in section 3.3. This method has control and update laws which guarantee uniform ultimate boundedness property of signals *x(t)* and *ξ(t)*. Therefore the true plant can be modelled by the recurrent neural network plus a modelling error term $\omega_0(x,u)$:

$$\dot{x} = -Ax + \mathrm{W}^* S(x) + S'(x)\mathrm{W}_1^* u + \omega_0(x,u) \tag{5.3}$$

where *u* is the control input.

The control law is the following:

$$\dot{u}^T = -\frac{1}{2}ku^T - x^T W_1 S'(x) \tag{5.4}$$

There are appropriate update laws of matrices of the weights *W, W_1,* which ensure the boundness of the weights estimates and guarantee the stability in the presence of modeling errors or disturbances. Specifically, we use proje-

ction algorithms which confine the weights and the following relations are valid:

$$\|W\| \le w_m$$
$$\|W_1\| \le \overline{w}$$

The update laws are the following:

$$\dot{W} = \begin{cases} -\frac{1}{2}kW - xS^T(x), if W \in W or \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} \ge 0 \\ -\frac{1}{2}kW - xS^T(x) + \Lambda, if \|W\| = w_m and tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\} < 0 \end{cases}$$

$$\dot{W}_1 = \begin{cases} -\xi x\left(S'(x)u\right)^T, if W_1 \in W_1 or \|W_1\| = \overline{w} and tr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} \ge 0 \\ -\xi x\left(S'(x)u\right)^T + P, if \|W_1\| = \overline{w} and tr\left\{\left(\xi x\left(S'(x)u\right)^T\right)W_1^T\right\} < 0 \end{cases}$$

where $\dot{\zeta} = -k\zeta$ and $\xi = \zeta - h$, $h = \frac{1}{2}(|x|^2 + |u|^2 + tr\{W^T W\})$,

$$\Lambda = tr\left\{\left(\frac{1}{2}kW + xS^T(x)\right)W^T\right\}\left(\frac{1+\|W\|}{w_m}\right)^2 W \ ,$$

$$P = tr\left\{\xi x(S'(x)u)^T W_1^T\right\}\left(\frac{1+\|W_1\|}{\overline{w}}\right)^2 W_1 \ .$$

The modelling error term satisfies:

$$|\omega_0(x,u)| \le k_1 + k_2|x| \qquad (5.5)$$

where k_1 and k_2 are known positive constants.

5.5 Results

We apply the method with the next dimensions of vectors and matrices:

$x \in R^3$,
$u \in R^1$,
W 3x3,
W_1 3x3,
A 3x3,
$S(x) \in R^3$,
$S'(x) \in R^3$,

We choose a relatively small value of the design constant:

k=1.5

We use the control law (5.4) of the method, which gives the dynamic feedback.
Moreover we choose:

$$S(x_i) = S'(x_i)$$

This choice gives a simpler model with simpler adaptive laws.

The initial values of the states, control input and the values of parameters of our system (Frequency factors and Activation energy) are in the following tables:

Variable	Initial value
m_1	2 (Mole)
m_2	1 (Mole)
m_4	0
T	296 (K)

Parameter	Value
A_1	$3*10^7$ (s^{-1})
A_2	$5.8*10^8$ (s^{-1})
A_3	$1.2*10^9$ (s^{-1})
E_1	$40*10^3$ (Joule/Mole)
E_2	$50*10^3$ (Joule/Mole)
E_3	$50*10^3$ (Joule/Mole)

The parameters of the sigmoid functions were set to:

μ=1

ℓ_0=1

λ=-0.5

The input is: y=296+270*sin(0.0035*t).

The following Figures give the state errors e1,e2,e3 of Substrate, Enzyme and Product respectively:

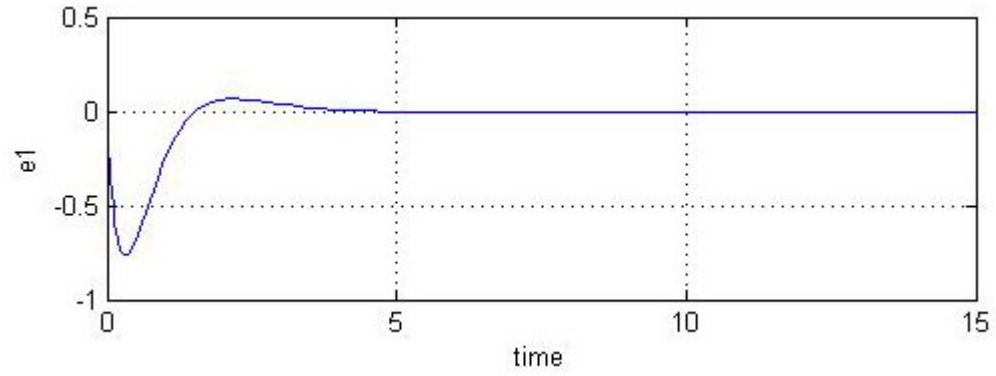

Fig. 5.1: Evolution of Substrate error (e1)

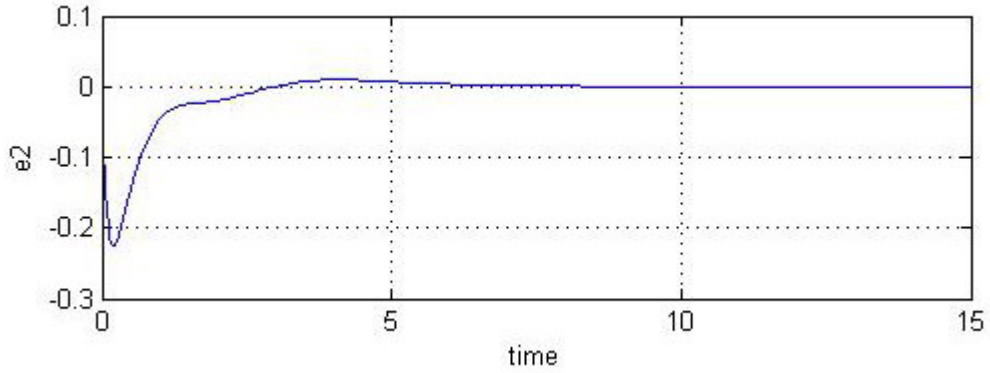

Fig. 5.2: Evolution of Enzyme error (e2)

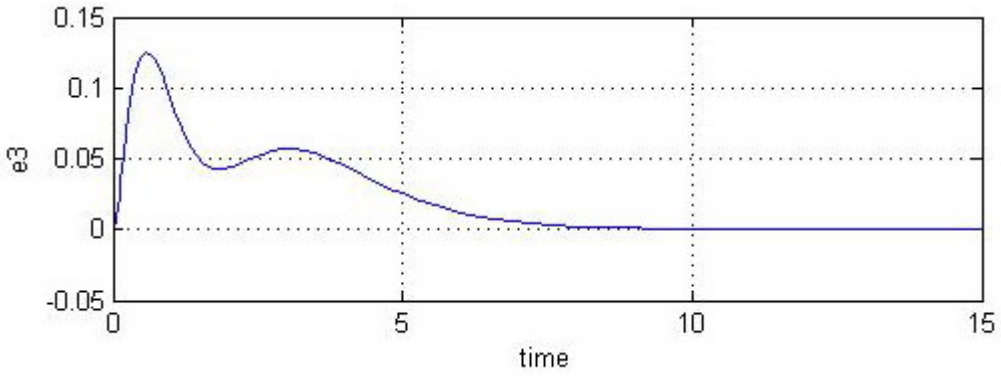

Fig. 5.3: Evolution of Product error (e3)

The error of Enzyme-Substrate: $m_3 = c_1 - m_2$ is the following:

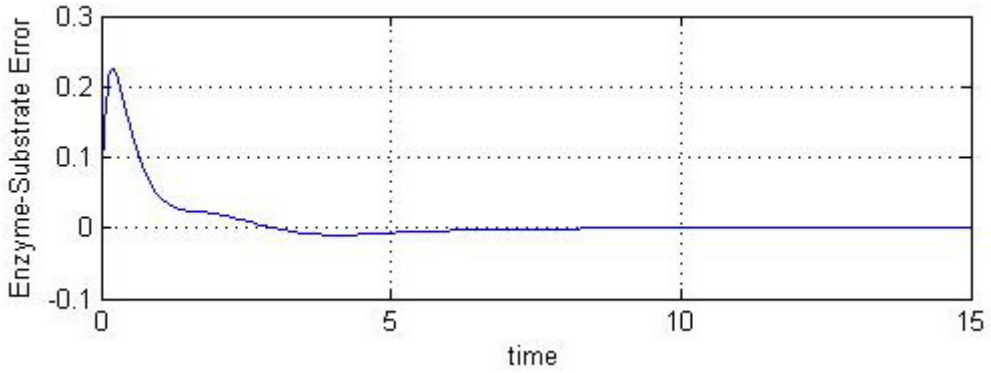

Fig. 5.4: Evolution of Enzyme-Substrate error

The following diagram gives the total state error:

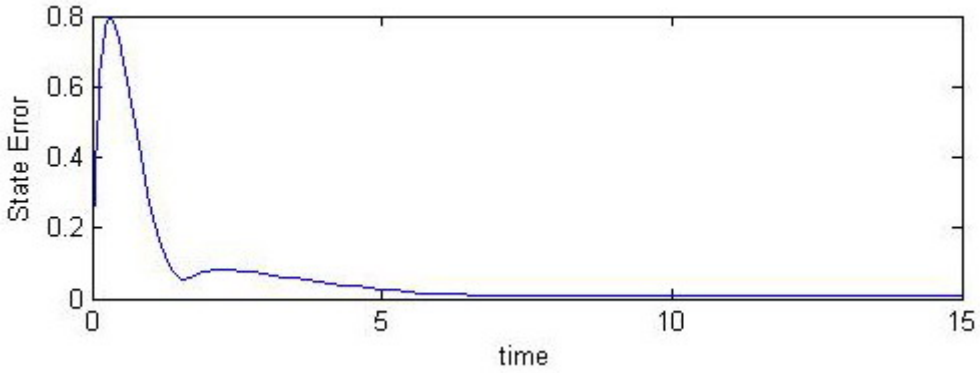

Fig. 5.5: Evolution of total State error

The evolution of Substrate, Enzyme and Product is the following, where the red line represents the actual system's state:

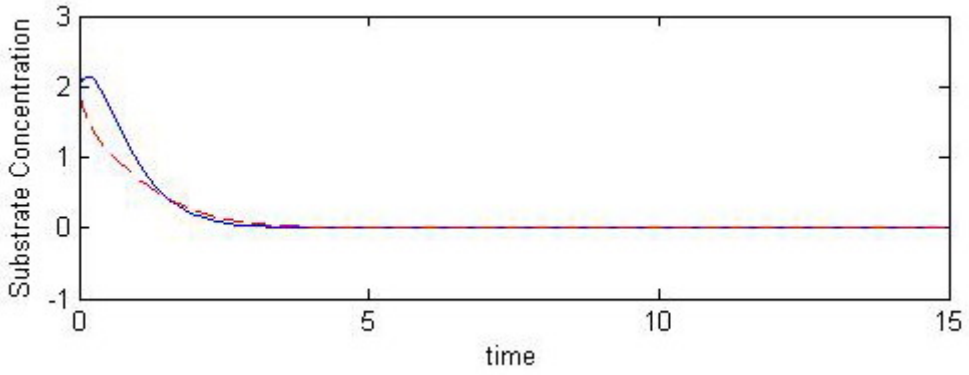

Fig. 5.6: Convergence of Substrate to actual state

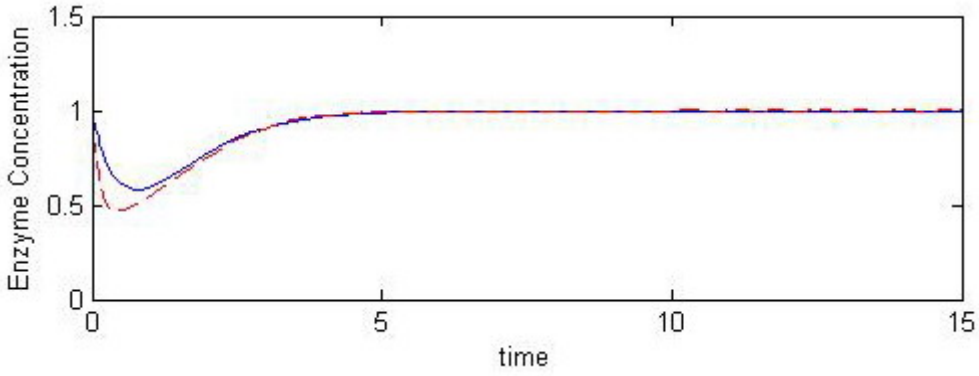

Fig. 5.7: Convergence of Enzyme to actual state

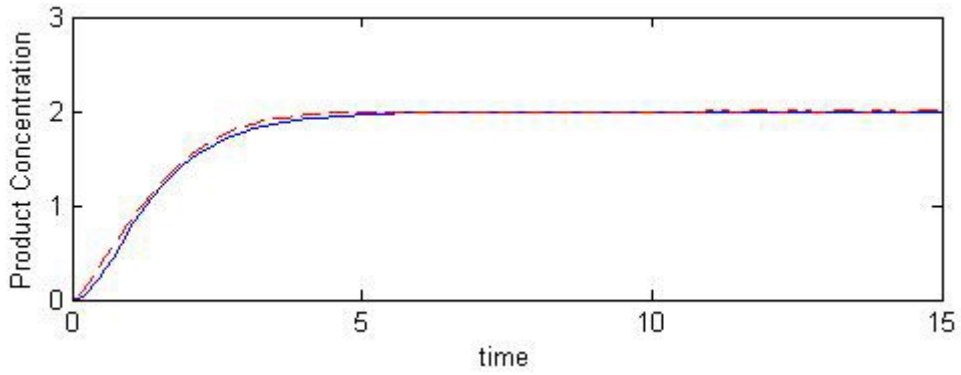

Fig. 5.8: Convergence of Product to actual state

Respectively, the Enzyme-Substrate is the following:

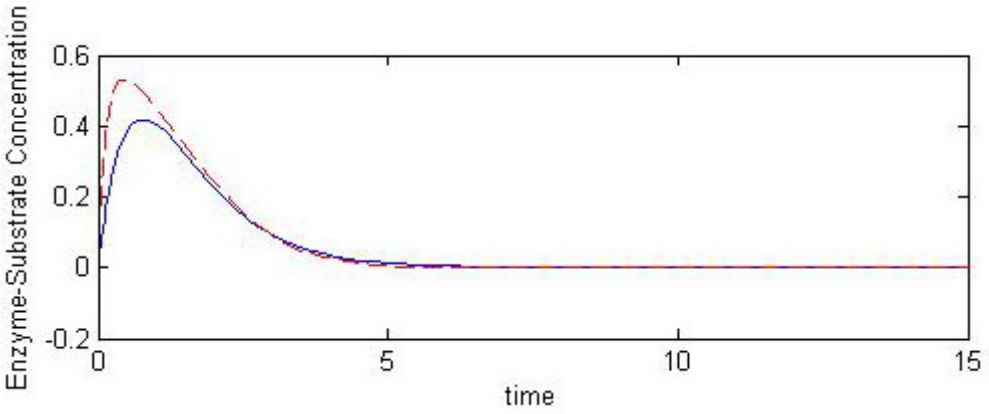

Fig. 5.9: Convergence of Enzyme-Substrate to actual state

The control action is the following:

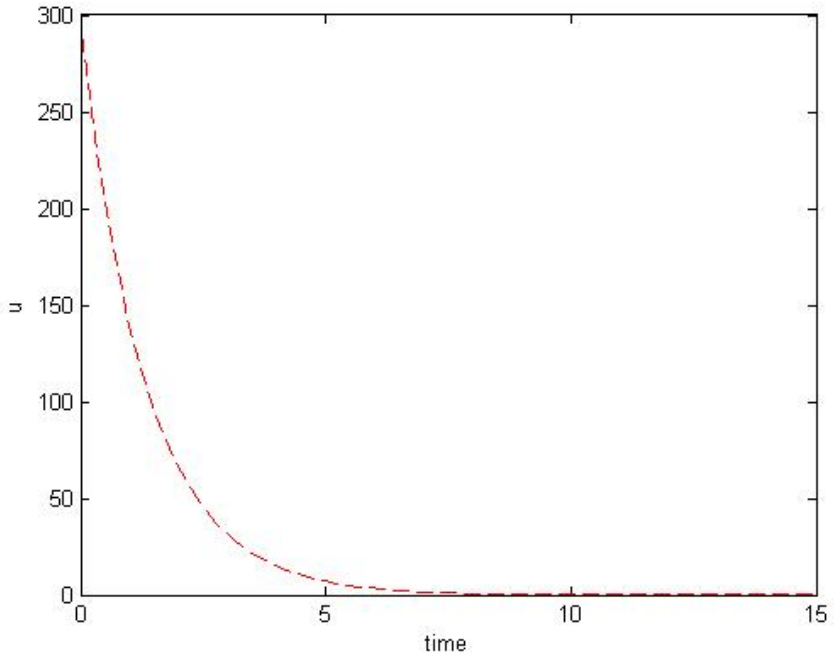

Fig. 5.10: Evolution of control action u for design constant k=1.5

Now, we take as input a white noise signal and apply the method using the same initial values of states and parameters as previously. The following Fi-

gures give the state errors e1,e2,e3 of Substrate, Enzyme and Product respectively:

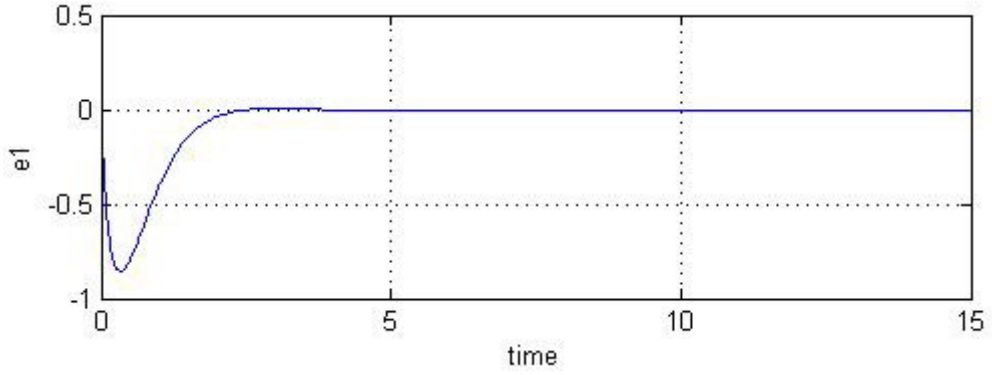

Fig. 5.11: Evolution of Substrate error e1

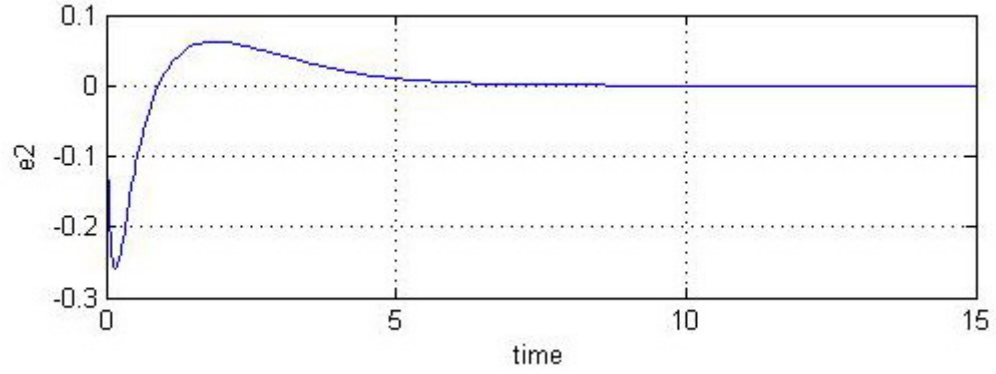

Fig. 5.12: Evolution of Enzyme error e2

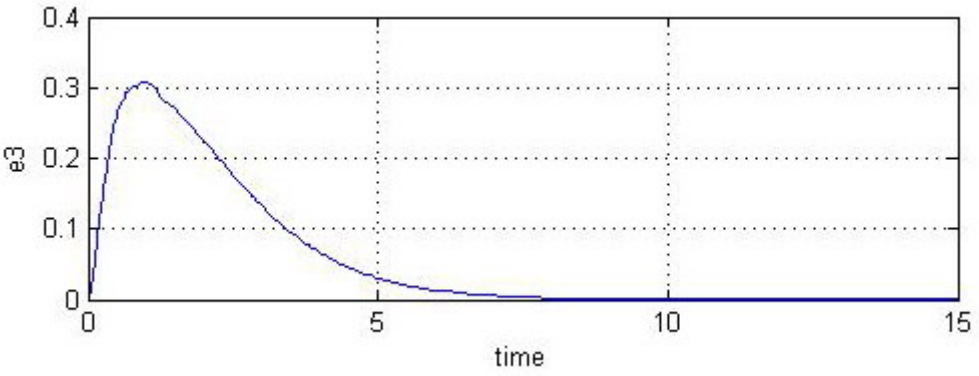

Fig. 5.13: Evolution of Product error e3

Respectively, the error of Enzyme-Substrate: $m_3 = c_1 - m_2$ is the following:

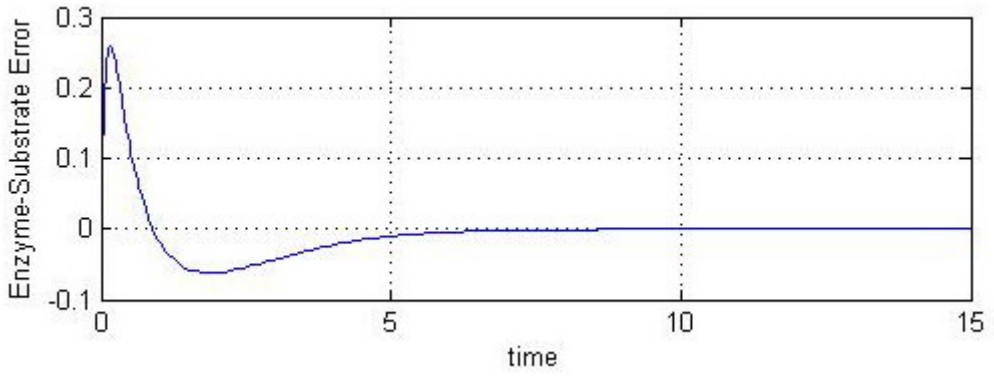

Fig. 5.14: Evolution of Enzyme-Substrate error

The following diagram gives the total state error:

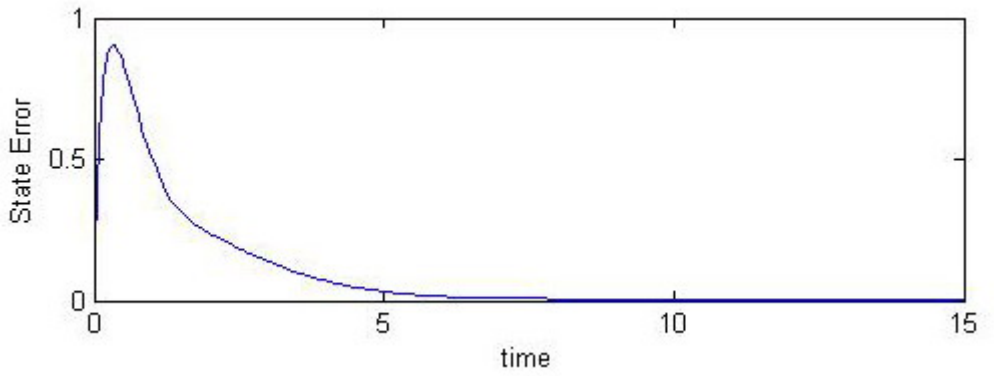

Fig. 5.15: Evolution of total State error

The evolution of Substrate, Enzyme and Product is the following, where the red line represents the actual system's state:

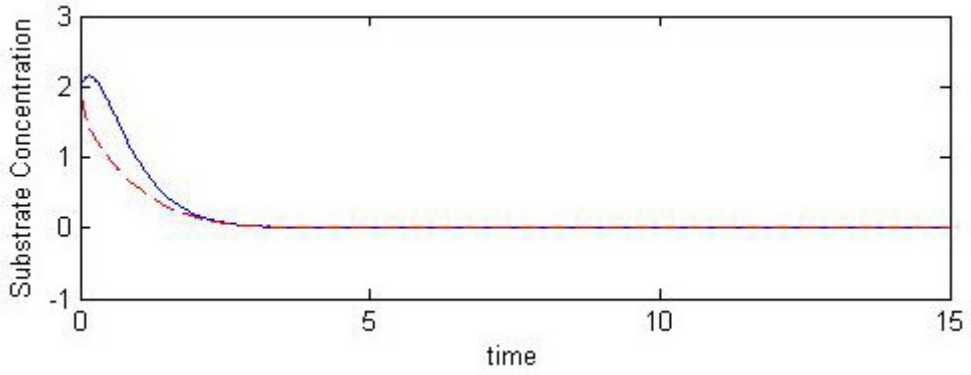

Fig. 5.16: Convergence of Substrate to actual state

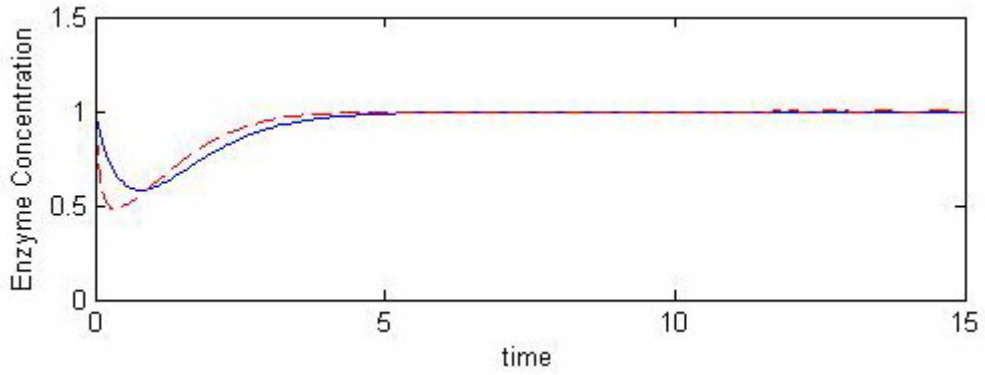

Fig. 5.17: Convergence of Enzyme to actual state

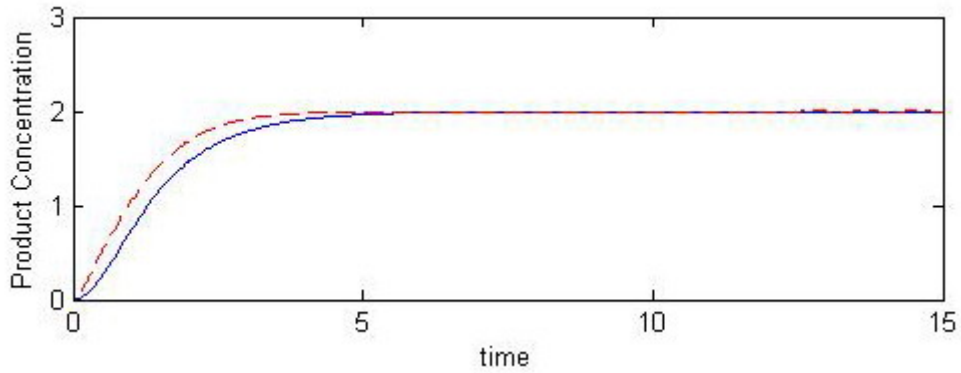

Fig. 5.18: Convergence of Product to actual state

The Enzyme-Substrate is the following:

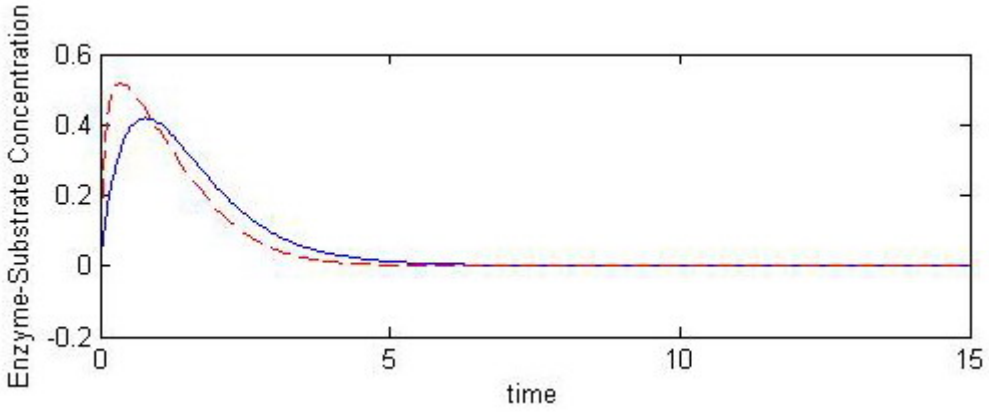

Fig. 5.19: Convergence of Enzyme-Substrate to actual state

The control action u is the following:

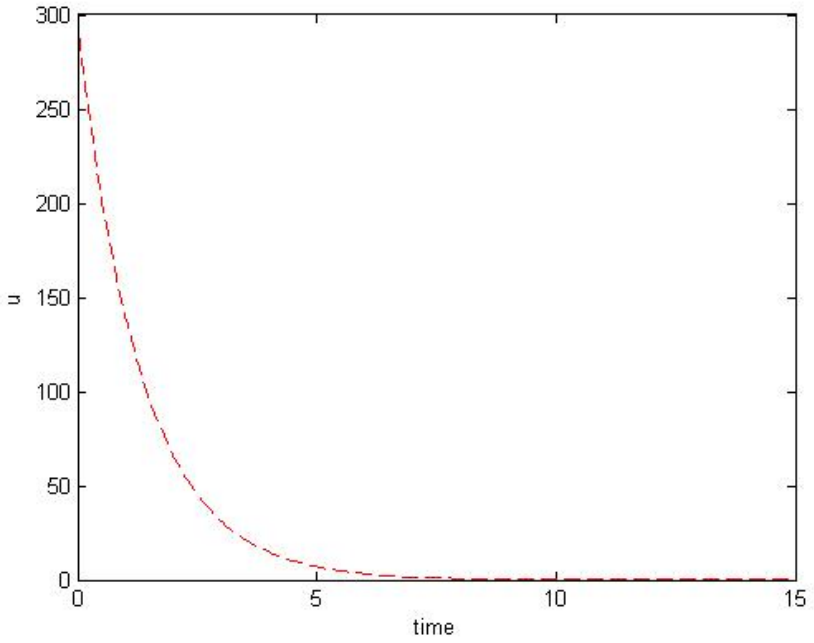

Fig. 5.20: Evolution of control action u for design constant k=1.5

5.6 Analysis of results

As we see at Figures (5.6), (5.7), (5.8) of states' evolution, all the three states of the system converge fast. Furthermore the convergence is achieved before the end of enzyme reaction. From Figures (5.1), (5.2) and (5.3) of state errors e1,e2 and e3, we infer that especially the states of substrate and enzyme converge more fast. The Figure (5.5) shows that the state error decreases dramatically before the first 2 sec, which proves the successful performance of the method.

In the second case with white noise as input we have similar results. All the states converge and again the state error decreases fast.

The control action to achieve the convergence is shown at Figures (5.10) and (5.20). We can see that the control action is smooth, because of the small value of design constant.

The design constant k is chosen appropriately in order to guarantee the uniform ultimate boundness of the state x. A large value of k creates large modeling errors. Moreover k is gain in $\zeta(t)$ and $u(t)$, consequently there is a compromise between k and the modelling error terms. In our case k has a relatively small value and from the results we infer that it is appropriate.

Another aspect is the appropriate choice of the initial weights W, W_1 and the matrix A which is a significant factor of good performance of the method.

CHAPTER 6
FINAL CONCLUSIONS & FUTURE WORK

6.1 Final Conclusions

The regulation of enzyme activity is crucial to the regulation of protein interactions within signal transdunction pathways. The aim of this project was the the implementation of Direct Adaptive method for regulating enzyme kinetics reactions and specifically the Michaelis-Menten mechanism for basic one-substrate enzyme reactions which consists of three elementary reactions steps. This is the basic scheme of an enzyme reaction. For this purpose Direct Adaptive Control method using RHONNs for affine in the control non-linear dynamical systems with $n!=m$ (n – the number of states, m – the number of control inputs) was implemented. We have also considered the more general case of modeling error at zero case, which is a usual instability mechanism.

In the non-linear system there were the rate equations of enzyme kinetics reactions and we used the absolute temperature T as control input. This system is non-affine in control and its linear approximation was implemented by the use of Taylor series expansion. The range of input was chosen to be 8K and specifically the area of interest was 296-304K. This area was adequate for the purpose of this project.

The Direct Adaptive Control method which was implemented produced satisfactory results, as presented in the previous Chapter. From these results we infer that the state error converged to zero and all signals in the closed loop were uniform ultimate bounded, as it was desirable. The appropriate values of

parameters, and especially the design constant *k,* played an important role in the performance of the closed loop system.

It is also remarkable the fact that the method showed stable behaviour for both of the inputs which were used.

6.2 Future Work

The design of adaptive controllers with certain robustness properties with respect to modeling errors or external disturbances can be further improved. In this project we have assumed that the modeling error term $\omega_0(x,u)$ satisfies a Lipschitz condition. This condition guarantees the existence and uniqueness of solutions of $\dot{x} = -Ax + \mathrm{W}^* S(x) + S'(x)\mathrm{W}_1^* u + \omega_0(x,u)$, which is necessary according to Theorem 2.1.1 for the actual system. Furthermore, larger values of k_1, k_2 cause larger modeling error, but we can take small k_1, k_2 because the approximation error ε can be considered arbitrarily small, according to Theorem 2.1.1

Recently, results about a special class of single-input- single-output systems, transformable via a global state-space diffeomorfism into a non-linear system the non-linearities of which depend only on the output, have been produced [31]. The design of controllers for uncertain systems started almost a decade ago, from [32] with the strict condition that only input-matched uncertainties were present. Moreover Barmish and Leitmann[33] allowed small mismatched uncertainties. Chen's work [34] was in an adaptive environ-ment. In [35] the conditions were less strict and n-dimensional single-input systems in the form of a perturbed chain of integrators were considered [5]. Generally, the design of robust controllers for more general classes of mo-deling errors and uncertainties is a field of further research.

RHONNs can perform simulations of the effects caused to signal transdunction pathways by factors, as temperature. Since cellular processes are very complex, RHONNs can be used in order to approximate dynamical

systems of the form: $\dot{x} = f(x,u)$. Furthermore these methods can help biologists to handle the huge number of experimental data.

Another significant matter is the use of Recurrent High Order Neural Networks in order to find the appropriate control input for pathways which do not have a desirable behaviour. RHONNs is an effective method and can calculate the control input in order the system obtain the desirable state. This can bring about benefits as the treatment of a disease.

In conclusion, biologists recognize system approaches as necessary to understand the complex mechanisms or interactions among cells. There are various tools and methods from system engineering which can help them handle the experimental data or make simulations. Consequently, the application of Control Theory to biological research is necessary and can lead to important innovations in Biology and Medicine.

BIBLIOGRAPHY

[1] Olaf Wolkenhauer, Hiroaki Kitano, and Kwang-Hyun Cho, *Systems Biology: Looking at opportunities and challenges in applying systems theory to molecular and cell biology*, IEEE CSS Magazine, 38-48, Aug. 2003

[2] Seng-Xiang Lin, Kuo-Chen Chou, J. Tze-Fei Wong, *A thermal-variation method for analyzing the rate-constants of the Michaelis-Menten mechanism,* Biochem. J. 207, 179-181 (1982)

[3] Olaf Wolkenhauer, *Systems Biology: Dynamic Pathway Modeling* (2004)

[4] Robert D. Phair, Tom Misteli, *Kinetic Modeling Approaches to In-Vivo Imaging,* Macmillan Magazines, Volume 2, 898-907 (2001)

[5] George A. Rovithakis, Manolis A. Christodoulou, *Adaptive Control with Recurrent High-Order Neural Networks: Theory and Industrial Applications*

[6] Pineda F. J., *Generalization of back propagation to recurrent networks*, Phys. Rev. Lett., volume 59, no. 19, 2229-2232 (1988)

[7] Werbos P. J., *Backpropagation through time: what it does and how to do it*, Proc. IEEE, volume 78, 1550-1560 (1990)

[8] Williams R. J., Zipser D., *A learning algorithm for continually running fully recurrent neural networks*, Neural Computation, volume 1, 270-280 (1989)

[9] Narendra K. S., Parthasarathy K., *Gradient methods for the optimization of dynamical systems containing neural networks*, IEEE Transactions on Neural Networks, Volume 2, no. 2, 252-262 (1991)

[10] Polycarpou M. M., Ioannou P. A.*, Identification and control of nonlinear systems using neural network models: design and stability analysis*, Tech. Rep. 91-09-01, Univ. of Southern Cal., Los Angeles (1991)

[11] Sanner R. M., Slotine J. -J. E., *Gaussian neural networks for direct adaptive control*, IEEE Transactions on Neural Networks, Volume 3, no. 6, 837-863 (1992)

[12] Chen F. -C., Khalil H. K., *Adaptive control of a class of nonlinear discrete time systems using neural networks*, IEEE Trans. Automatic Control, Volume 40, no. 5, 791-801 (1995)

[13] Chen F. -C., Liu C. -C., *Adaptively controlling nonlinear continuous-time systems using multilayer neural networks*, IEEE Trans. Automatic Control, Volume 39, no. 6, 1306-1310 (1994)

[14] Lewis F. L., Liu K., Yesildirek A., *Neural net robot controller with guaranteed tracking performance*, IEEE Transactions on Neural Networks, Volume 6, no. 3, 703-715 (1995)

[15] Sadegh N., *Nonlinear identification and control via neural networks*, Control Systems with Inexact Dynamic Models, DSC-Volume 33, ASME Winter Annual Meeting (1991)

[16] Sadegh N., *A perceptron network for functional identification using radial gaussian networks*, IEEE Transactions on Neural Networks, Volume 4, no. 6, 982-988 (1993)

[17] George A. Rovithakis, Manolis A. Christodoulou, *Adaptive Control of Unknown Plants using Dynamical Neural Networks*, IEEE Trans. Systems Man Cybernetics, Volume 24, no. 3, 400-412 (1994)

[18] Kosmatopoulos E. B., Polycarpou M., Christodoulou M. A., Ioannou P. A., *High-order neural network structures for identification of dynamical systems*, IEEE Transactions on Neural Networks, Volume 6, no. 2, 422-431 (1995)

[19] Hopfield J. J., *Neurons with graded response have collective computational properties like those of two-state neurons*, Proc. Natl. Acad. Sci., Volume 81, 3088-3092 (1984)

[20] Paretto P., Niez J. J., *Long term memory storage capacity of multiconnected neural networks*, Biol. Cybern., Volume 54, 53-63

[21] Baldi P., *Neural Networks, orientations of the hypercube and algebraic threshold functions*, IEEE Transactions on Information Theory, Volume IT-34, 523-530 (1988)

[22] Kamp Y., Hasler M., *Recursive Neural Networks for Associative Memory*, J. Wiley &Sons (1990)

[23] Giles C. L., Chen D., Miller C. B., Chen H. H., Sun G. Z., Lee Y. C., *Second-order recurrent neural networks for grammatical inference*, Proc. of Inter. Joint Conf. on Neural Networks, IJCNN91, Volume 2, 273-281 (1991)

[24] Liou R., Azimi-Sadjadi M. R., Dent R., *Detection of dim targets in high cluttered backround using high order correlation neural network*, Proc. of Inter. Joint Conf. on Neural Networks, IJCNN91, Volume 1, 701-706 (1991)

[25] Hale J. K., *Ordinary differential equations*, New York, NY, Wiley-InterScience (1969)

[26] Cotter N. E., *The Stone-Weierstrass theorem and its application to neural networks*, IEEE Transactions on Neural Networks, Volume 1, no. 4, 290-295 (1990)

[27] Cybenko G., *Approximations by superpositions of a sigmoidal function , Mathematics of Control, Signals and Systems*, Volume 2, 303-314 (1989)

[28] Funahashi K*., On the approximate realization of continuous mappings by neural networks*, Neural Networks, Volume 2, 183-192 (1989)

[29] Hartman E. J., Keeler J. D., Kowalski J. M., *Layered Neural Networks with Gaussian Hidden Units as Universal Approximators*, Neural Computation, Volume 2, 201-215 (1990)

[30] Hornik K. M., Stinchombe M., White H., *Multilayer feedforward networks are universal approximators, Neural Networks*, Volume 2, 359-366 (1989)

[31] Marino R., Tomei P., *Global Adaptive Output Feedback Control of Nonlinear Systems, Part II: Nonlinear Parameterization*, IEEE Transactions Aut. Contr., Volume 38, no.1, 33-48 (1993)

[32] Corless M. J., Leitmann G., *Continuous State Feedback Guaranteeing Uniform Ultimate Boundness for Uncertain Dynamic Systems*, IEEE Transactions Aut. Contr., Volume AC-26, no. 5, 1139-1144 (1981)

[33] Barmish B. R., Leitmann G., *On ultimate Boundness Control of Uncertain Systems in the Absence of Matching Condition*, IEEE Transactions Aut. Contr., Volume AC-27, no. 1, 153-158 (1982)

[34] Chen Y. H., *Adaptive Robust Control of Uncertain Systems with Measurement Noise*, Automatica, Volume 29, no. 4, 715-728 (1992)

[35] Freeman R. A., Kokotovic P. V., *Design of Softer Robust Nonlinear Control Laws*, Automatica, Volume 29, no. 6, 1425-1437 (1993)

[36] Cohen M. A., Grossberg S., *Absolute stability of global pattern formation and parallel memory storage by competitive neural networks*, IEEE, Transactions on Systems, Man, and Cybernetics, Volume SMC-13, 815-826 (1983)

[37] Dempo A., Farotimi O., Kailath T., *High-order absolutely stable neural networks,* IEEE Transactions on Circuits and Systems, Volume 38, no. 1 (1991)