AN INTELLIGENT FRAMEWORK FOR MODELLING AND ACTIVE VIBRATION CONTROL OF FLEXIBLE STRUCTURES

by

SITI ZAITON MOHD HASHIM

A thesis submitted to
The University of Sheffield
for the fulfillment of the degree of

DOCTOR OF PHILOSOPHY

The Department of Automatic Control and Systems
Engineering,
The University of Sheffield
November 2004

Acknowledgement

First and foremost, I thank God Al-mighty for giving me the strength to be able to finish this thesis. I am truly indebted to my supervisor, Dr. Osman Tokhi for his excellent guidance, consistent encouragement and patience throughout the course of this research. I have learnt a lot from his comments and suggestions, which are always very inspiring and fruitful. Here, I would like to extend my prayer and best wishes to him in his future career.

I am grateful to my colleagues, past and present members of Dr Osman Tokhi's group, especially Intan Zaurah Mat Darus, Hazinah Kutty Mammy, Zarhamdy Mohd Zain, Rasha Massoud, Zaharuddin Mohamed, Takatoshi Okuno, Omar Khaidzir, Miguel Martinez, Rapelang Marumo, Kamil Ahmad, Maziah Jafferi, Fareg Aldebrez and Shafiul Alam for their helpful and stimulating discussion. Special thanks also due to Dr Alamgir Hossain for his kind assistance in the early part of my research.

I wish to thank Universiti Teknologi Malaysia and Malaysian Government for their financial support during the full term of this research.

Last but not least, I want to say a heart-felt 'thank you' to my dear husband, parents and children, for their continuous prayer, support and love throughout the years.

Abstract

This thesis presents investigations into the development of an intelligent framework for modelling and active vibration control (AVC) of flexible structures. Dynamic characterisations of one-dimensional flexible beam and two-dimensional flexible plate structures are presented and simulation algorithms characterising the behaviour of each structure is developed using finite difference methods. The deflection of the structures in several modes, obtained from the simulation, has been validated by comparing these with previously reported theoretical work.

Parametric and non-parametric modelling of such systems is investigated. Parametric approaches include linear parametric modelling of the system using recursive least squares (RLS) and genetic algorithms (GAs); and non-parametric approaches include multi-layered perceptron neural networks (MLP-NNs) and adaptive neuro-fuzzy inference systems (ANFIS) are employed. A comparative assessment of the techniques used is presented and discussed in terms of accuracy, efficiency and performance in estimating the modes of vibration of the system.

Single-input single-output AVC systems using RLS, GAs, MLP-NN and ANFIS are developed and implemented within a flexible beam and flexible plate simulation environments to yield optimum cancellation of broadband vibration at an observation point on the structure. Two controller design formulations are proposed. The first controller design is formulated so as to allow on-line modeling, controller design and implementation and thus, yield a self-tuning control algorithm. Performance of the AVC algorithm is assessed based on parametric design techniques, using RLS and GAs, and non-parametric design techniques, using MLP-NN and ANFIS in the suppression of vibration of the flexible structures. The second controller design strategy is based on a cost function optimization using GAs. This approach bypasses modelling of the plant and results in direct estimation of the controller characteristics. Performance of this controller formulation is assessed in flexible beam and plate structures.

The work is further extended to developing and integrating the idea of active control of flexible structures into an interactive learning environment. The environment is implemented in such a way that it allows the user to simulate and visualise behaviour of flexible structures with given physical characteristics, to test and validate controller

designs, and furthermore, to execute such processes repeatedly in a friendly and easy manner. The simulation algorithm and interactive environment thus developed and validated form suitable test and verification platforms for the development of AVC strategies for flexible structures as well as for learning and research purposes.

TABLE OF CONTENTS

List of figures and tables

Chapter 1

Introduction

1.1 Motivation of research

The control of structural vibration has numerous applications in manufacturing, infrastructure engineering, and consumer products. In the machine tool industry, mechanical vibration degrades both the fabrication rate and quality of end products. In civil engineering constructs, structural vibration degrades human comfort. In automative and aerospace systems, vibration reduces components life, and the associated acoustic noise annoys passengers. Since advanced technical systems, such as aircraft, space structures and automobiles have to combine high performance with low weight, passive vibration absorbers reach their limits (Stobener and Gaul, 2000). This is why active control has become more important in structural dynamics with successful applications.

Active vibration control (AVC) is a vast research area that incorporates interdisciplinary technologies. For example, a typical AVC system is an integration of mechanical and electronic components in synergistic combination with computer/microprocessor control. The major components of any AVC system are the mechanical structure influenced by disturbance (creating unwanted vibration), sensors (to perceive the vibration), controllers (to intelligently make use of the signals from the sensors and to generate the appropriate control signals), and actuator (which counteracts the influence of the disturbance on the structure). Destructive interference from the forces generated by the actuators reduces and/or cancels the effects of the disturbance on the structure (Alkhatib and Golnaraghi, 2003; Tokhi and Veres, 2002).

There are many types of active methods that can be considered for obtaining a suitable control law; these include PID, deadbeat, optimal control and state space control techniques (Al-Dmour and Mohammad, 2002). However, to be able to control highly complex systems with noisy environment, deeper understanding of the processes involved and systematic design methods are needed, and quantitative models and design techniques are required. Such a need is quite apparent in intelligent autonomous control systems and in particular in hybrid control systems (Xiao-Zhi, 1999).

There is no formal or single definition of an intelligent control system. Generally, an intelligent system should satisfy the famous Turing test, which can be concisely expressed as follows: if a man and a machine (or a program) perform the same task, then if one cannot distinguish between the machine and the human by examining only the nature of their performances, the machine is said to be intelligent, otherwise not (Turing, 1950).

Figure 1.1 shows an illustrative diagram of the relationship between intelligent control and four significant relevant fields. However, intelligent control systems can be broadly described as the use of artificial intelligence (AI) related methods to design and implement automatic control systems (Omatu *et al.*, 1995). Soft computing is the most prominent AI technique in the control field. Among these, for example, fuzzy logic and neural networks (NNs) are already utilized in various control systems, such as washing machines, ship navigation, subway train control, and automobile transmission (Passino and Yurkovich, 1998).

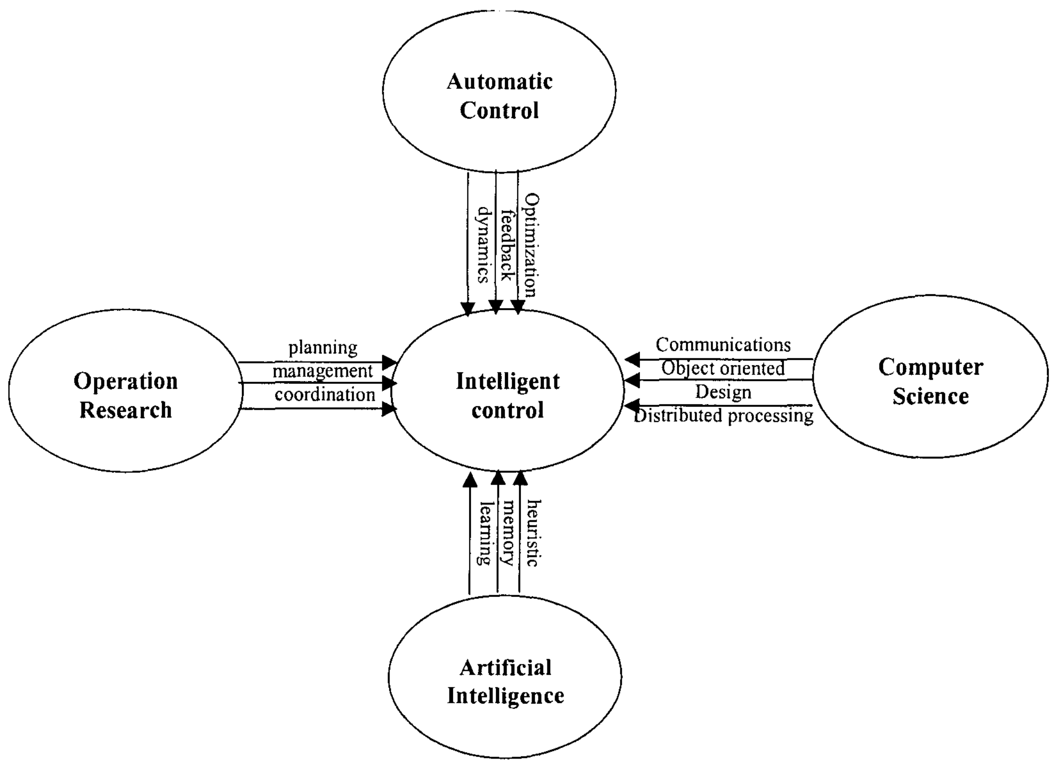

Figure 1.1: Techniques employed in intelligent control (Brown and Harris, 1994)

It has been demonstrated that NNs are well suited for the control of complex dynamic systems, eg. control of autonomously driven vehicle (Thorpe *et al.*, 1991) where good success has been achieved with a vehicle traveled at speeds of 55 mph for distances of 90 miles or more on various road types and under various weather conditions. In steel-making production, the intelligent arc furnace (IAF) regulator, designed by William Staib, an NN-based electrode positioning system for electric arc furnaces released in September 1992, uses the pattern recognition abilities of NNs to predict and correct for changes in furnace operation (Patterson, 1995). Continuously adopting its control strategy, the IAF is said to save several million dollars a year on typical installation by increasing productivity by over twelve percent and by reducing power consumption and electrode wear (Patterson, 1995).

Genetic algorithms (GAs) are on the other hand, widely used in optimising the parameters of diverse kinds of controllers (Chipperfield *et al.*, 1994; Vieira *et al.*, 2003). Hence, current emphasis is focused on developing artificial neural networks (ANNs), GAs, and fuzzy logic based identification and control mechanisms as they have been proven to solve many scientific and engineering problems more efficiently as compared to conventional approaches (Glackin *et al.*, 2004; Khoo *et al.*, 2000; Shen *et al.*, 2000; Vieira *et al.*, 2004).

The concept of e-learning through interactive and user-friendly environment is extensively motivated in a number of fields of study. Hough and Marlin (2000) stated that the use of simulation is particularly effective in process control education because of the complex behaviours of control systems. In light of this and the widely available computing technology, this work is further looking at developing and integrating the idea of active control of flexible structures into an interactive learning environment, generally as a learning and research facility. Hence, an interactive learning environment for dynamic simulation and AVC of flexible structures is implemented.

1.2 Modelling and control of flexible structures

Vibration suppression of flexible structures is a topic of research interest that has received substantial attention in recent years (Chen *et al.*, 2002; Hossain, 1995; Sadri *et al.*, 1996; Sawada and Ohsumi, 1992; Xu and Cao, 1999; Zhuomin, 2002). Examples of flexible structures are bridges, high-rise buildings, still tables, beams, robot arms, enclosures etc. (Tokhi and Veres, 2002). The purpose of vibration control in flexible structures is to dampen the response of the structure to external excitation, which could lead to structural damage. In all cases there are the alternatives of passive or active control solutions. Active control solutions are defined as those which will require external energy to produce control signals within the controlled system. On the other hand, passive control systems can be considered as adding structural elements to a flexible structure in order to improve vibrational characteristics of the system (Al-Dmour and Mohammad, 2002).

It is important initially to recognize the flexible nature of a structure and construct a mathematical model for the system. In order to control the vibration of the structure efficiently, it is required to obtain an accurate model of the structure because an accurate model will result in satisfactory and good control. A model can be constructed using a partial differential equations (PDE) formulation of the dynamics of the flexible structure. Finite difference (FD) method is used here for obtaining a numerical solution of the PDE of one-dimensional flexible beam and two-dimensional flexible plate structures for constructing suitable simulation algorithm characterising the behaviour of such structures. It has been reported that in applications involving uniform structures, such as beam and plate systems, the FD method is found to be more appropriate, and the relatively reduced amount of computation involved in the FD method makes the technique more suitable in real-time applications (Azad 1994; Hossain 1996; Karmoulis 1990; Mat Darus 2003). The method involves dividing the beam or plate into a set of equal-length sections, deriving a difference equation for each section describing its dynamic behaviour, and assembling the set of equations and the corresponding boundary conditions. A linear difference equation is developed for the deflection of each section using FD approximation. The implementations of FD algorithms allow application, detection and observation of a disturbance signal at any grid point on the structure. Such a provision is desirable for the design and development of AVC techniques for the system. The simulation algorithms thus developed and validated form suitable test and verification platforms in subsequent investigations for development of AVC strategies for flexible beam and plate structures.

1.3 Soft computing methodology and its application

Zadeh (2001) emphasized that, by design, soft computing is plurastic in nature in the sense that it is a coalition of methodologies, which are drawn together by a quest for accommodation with the pervasive imprecision of the real world. The principle partnership at this juncture is fuzzy logic (FL), neurocomputing (NC), and probabilistic reasoning (PR), with the latter subsuming GAs, chaotic systems, belief networks and parts of learning theory. The pivotal contribution of FL is a methodology for computing with words; that of NC is system identification, learning and adaptation; that of PR is propagation of belief; and that of GA is systematized random search and optimization. In the main, FL, NC, PR and GA are complementary rather than competitive. For this reason, it is frequently advantageous to use FL, NC, PR and GA in combination rather than exclusively, leading to so-called hybrid intelligent systems. Also emerging is fuzzy-genetic, neuro-genetic and neuro-fuzzy-genetic systems (Zadeh, 2001; Zadeh, 1996).

Basically, soft computing is considered as an emerging approach to computing which parallels the remarkable ability of the human mind to reason and learn in a circumstance of uncertainty and imprecision. Zadeh (1992), as the pioneer of fuzzy logic, has pointed out that 'the guiding principle of soft computing is to exploit the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, low solution cost, better rapport with reality'.

This research concentrates upon utilizing several of the methods mentioned above, namely NNs, neural-fuzzy and GAs to achieve good vibration reduction in applications involving flexible structures such as beam and plate. The work is also compiled together and integrated in the form of a learning environment, whereby it can facilitate works related to this either in research or education contexts.

1.3.1 Neural networks

Neural Networks originated in an attempt to build mathematical models of elementary processing units of the brain and the flow of signals between these processing units. After a period of stagnation, these formal models have become increasingly popular, with the discovery of efficient algorithms capable of fitting them to data sets. Since then, neural nets have been applied to build computerized architectures that can approximate non-linear functions of several variables, and classify objects. A neural network is nothing more than a sophisticated black box non-linear model that can be trained on data. They are networks of highly interconnected neural computing elements that have the ability to respond to input stimuli and to learn to adapt to the environment. It is believed by many researchers in the field that NN models offer a promising approach to building intelligently computer systems (Patterson, 1996).

Artificial NNs have been shown to be effective as computational processors for various tasks including pattern recognition (e.g. speech and visual image recognition), associative recall, classification, data compression, modeling and forecasting, combinatorial problem solving, adaptive control, multisensor data fusion and noise filtering. They exhibit a number of desirable properties not found in conventional symbolic computation systems including robust performance when dealing with noise or incomplete input patterns, a high degree of fault tolerance, high parallel computation rates, the ability to generalize, and adaptive learning (Omatu *et al.*, 1995; Patterson, 1996).

Bernard Widrow was one of the early researchers to develop practical applications of NNs. He developed a simple neural element similar to the perceptron (network that is made up of several layers) called ADALINE (Adaptive Linear Neuron), and networks of ADALINEs called MADALINES (Multiple ADALINEs). These types of units are in use today as adaptive echo suppressors for long distance telephone circuits and as noise suppressors for high speed MODEMs (Patterson, 1996).

Neural networks have been used effectively in learning to control outdoor mobile robots, including autonomous and land vehicles tasks. They have been trained to learn the difficult task of backing up a trailer truck to a loading dock with minimal effort, even trucks with double trailers attached. They have also been used to efficiently control the positioning of huge electrodes in electric and arc furnaces used by steel-making companies, saving the companies millions of dollars through reduced electricity consumption and extended life of costly equipment. NNs have been used to control and optimize chemical plant processes saving companies huge sums through better process control and material usage. Dozens of consumer products, especially those manufactured by Japanese companies, now use NN technology for more effective and efficient control (Miller *et al.*, 1990; Patterson, 1996).

1.3.2 Genetic algorithms

Genetic algorithms are stochastic combinatorial approximation procedures that have been inspired by a biological analogy, the mutation and crossover of chromosomes in genetics. They belong to a large class of meta-heuristics that are used to avoid local minima in heuristic search methods, such as simulated annealing (Man *et al.*, 1999).

GAs were first introduced and studied by John Holland and his coworkers in 1975 (Patterson, 1996). A GA performs a global, random, parallel search for an optimal solution using simple computations. Starting with an initial population of genetic structures, genetic inheritance operations based on selection, mating, and mutation are performed to generate "offspring" that compete for survival ("survival of the fittest") to make up the next generation of population structures.

GAs have been theoretically and empirically proven to provide robust search in complex spaces, because they are not limited by restrictive assumptions of continuity, existence of derivatives, unimodality and other restrictions (Goldberg, 1989; Man *et al.*, 1999; Patterson, 1996). In the GA paradigm, knowledge is represented as a population pool of several genetic structures. The genetic structures can represent for example:

- Fuzzy membership function parameters

- "If … Then …" rules
- Connections or weight values in a neural network
- The response surface of a nonlinear control system
- Moves in a board game
- A function value

The GA operators are then used to optimize the structures invoked through random search.

1.3.3 Fuzzy logic

Fuzzy logic was invented when it was realized that control theory had become beautiful enough to carry on its development on its own, without worrying about many real problems it could not solve. Most real complex control problems involve human. Hence applying control theory to complex control problems may require a formal understanding of how a human operator understands his system, what his goals are, and how he proceeds when controlling it. It requires a dedicated tool for representing human-originated information in a flexible way, and this is where fuzzy logic enters the picture (Zadeh, 1965; Zadeh, 1973).

The term "fuzzy logic" is itself rather ambiguous because it refers to problems and methods that belong to different fields of investigation. In its most popular accepted meaning, it refers to numerical computations based on fuzzy rules, for the purpose of modelling a numerical function in systems engineering. However, in the mathematically oriented literature, fuzzy logic means multiple-valued logic, with the purpose of modelling partial truth-values and vagueness (Lin and Lee, 1996; Novak, 1996).

1.3.4 Neuro-fuzzy systems

Fuzzy systems are structured numerical estimators. They start from highly formalized insights about the structure of categories found in the real world and then articulate fuzzy IF-THEN rules as a kind of expert knowledge. Fuzzy systems combine fuzzy sets with fuzzy rules to produce overall complex non-linear behaviour. Neural networks on the other hand, are trainable dynamical systems whose learning, noise-tolerance, and generalization abilities grow out of their connectionist structures, their dynamics, and their distributed data representation. Neural networks have a large number of highly interconnected processing elements (nodes) which demonstrate the ability to learn and generalize from training patterns or data; these simple processing elements also collectively produce complex non-linear behaviour. Putting the two capabilities together, the power of a neuro-fuzzy computing system is realized (Lin and Lee, 1996; Patterson, 1996). Several proposals have been made for neuro-fuzzy systems, (Patterson, 1996; Mitra *et al.*, 2000; Yager *et al.*, 1994). The proposed methods differ in both representations and network architectures.

Networks of this type have several advantages over conventional systems. For example, they are able to encode expert knowledge given in imprecise linguistic terms and still improve the accuracy of the knowledge through training examples. Furthermore, the network is not simply a black box; its operations can be understood from the rules it encodes. Neural networks and fuzzy logic expert systems have also been used in a complimentary rather than combinatorial way. For example, neural networks have been used to speed up the design of fuzzy systems and improve their performance through higher accuracy in (Jang *et al.*, 1997; Lin and Lee, 1996; Patterson, 1996; Pedrycz, 1993):

a) determining the optimal number of fuzzy rules for a system

b) determining the best membership functions and

c) the adjustment of membership functions adaptively when environmental changes occur.

Neural networks can also be used to modify the fuzzy reasoning results of a fuzzy expert system. Neural networks and fuzzy logic compliment each other in the following ways:

- Fuzzy logic can express qualititative "values" of human logic well and provide smooth actions through continuous membership functions (good for control and other application).
- Fuzzy logic rules can express a wide range of condition/action relationships thereby requiring fewer rules than conventional logic-based expert systems.
- Neural networks are good for unstructured tasks such as pattern recognition and they can determine membership relations.
- Neural networks can learn to formulate complex nonlinear functions from training examples such as multidimensional membership function surfaces, which are difficult to design (e.g. temperature, humidity, wind velocity).
- Neural networks can learn various tasks from training examples (adaptive reasoning) including temporal pattern sequences.

1.3.5 Analysis of soft computing methods

Figure 1.2 shows at least as far as FL, NNs and GAs are concerned, how the various components of soft computing can be approximately ordered on a time scale and on a scale relating to their learning capacity (Sharma, 2000). The time scale shown is ordered according to the learning time. FL is not capable of learning anything while NNs and GAs have this capacity although it can be said that pure GAs in general require longer learning times. Looking from another aspect, FL needs more detailed knowledge of the problem to be solved compared to GA and NN that need no knowledge and very little knowledge respectively.

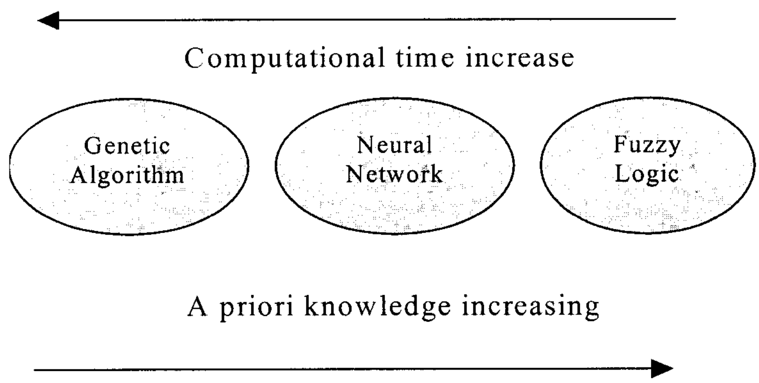

Figure 1.2: Soft computing components

In effect, each of these three areas of soft computing has its own advantages and disadvantages. FL does not share the inherent concept of NNs, i.e. automatic learning. So, FL is impossible to be used when experts are not available. It does however, have a great advantage over the other two techniques. Expressed accordingly to fuzzy canons, the knowledge base is computationally much less complex and the linguistic representation is very close to human reasoning.

NNs are quite different, at least in respect of the typical features of gradient descent learning networks. They are, therefore, fundamental when only some significant examples of the problem to be solved are available. There are two evident advantages in using NNs. In general, NNs can learn correctly from examples. However, what is learned is not easy for humans to understand. That is, the knowledge base extracted from NNs does not have such an intuitive representation as that provided by FL for example. Secondly, the type of functions that can be used in NNs have to possess precise regularity features and the derivative of these functions has to be known prior to the process.

Similar considerations hold for GAs, with certain clarifications. Their learning speed is usually slower. However, they have two great advantages over NNs. The functions that can be used in GAs can be much more general in nature, and knowledge of the gradient of the functions is not usually required. Finally, as these algorithms explore in several directions at the same time, they are affected much less than NNs by finding a local extreme rather than a global one. Even if the extreme found is not a global one, it is likely to correspond to a less significant learning error.

On the basis of these considerations, it is the opinion that a technique, which makes use of soft computing (SC) components, i.e. GAs, NNs, and FL, would be an interesting and useful prospect.

1.4 Overview and recent developments

The use of soft computing techniques in identification and control applications is widely reported in the literature. Omatu *et al.* (1996) highlighted that neuro-control can realize a nonlinear control algorithm which is robust to noise, complexities, and variations in the plant. One of the reasons why a lot of attention has been paid to neuro-control is that conventional and traditional control methodologies are mainly based on linear systems theory while real plants are in effect nonlinear in nature and have unmodeled dynamics.

The idea of applying fuzzy logic to control systems was first developed by Mamdani and his colleagues, based on Zadeh's fuzzy set theory, where they developed what is now referred to as the basic fuzzy logic controller which is used to regulate the output of a process around a given set-point using a digital computer (Jang *et al.*, 1997). There are several advantages in applying fuzzy logic to control, e.g. a controller can be developed along linguistic lines to define the individual control situations. These linguistic conditional statements can be easily developed from common sense or from engineering judgement of the process to be controlled. In addition, fuzzy logic controllers can also deal with ill-defined systems of unknown dynamics as they do not require a priori mathematical model of the plant. Also, fuzzy logic controllers can now be feasibly implemented in digital or analog VLSI circuitry where sampled information can be encoded in a parallel-distributed framework (Omatu *et al.*, 1996). In addition, GAs have been recognised as a powerful tool in many control applications such as parameter identification and control structure design (Hossain, 1996; Lennon and Passino, 1999; Marumo and Tokhi, 2004; O'mahony *et al.*, 2000), where for example, GA can be employed as an alternative to orthogonal least-square regression to find a smaller set of non-linear model terms from a broader set of possible terms.

Application of NNs for identification and control of systems has also gained significant momentum in recent years. The mapping properties of artificial neural