University of Southampton, 1994

FACULTY OF ENGINEERING AND APPLIED SCIENCE
INSTITUTE OF SOUND AND VIBRATION RESEARCH

FUZZY CONTROL OF THE FLOW OVER AN AEROFOIL

by Mats Ivar Gustavsson

A dissertation submitted in partial fulfilment of the requirements for M.Sc. by instructional course.

Preface to the Digitised Edition (2026)

Firstly, I will always be immensely grateful to the **University of Southampton**. The time I spent on the Highfield campus and at the **ISVR** some 32 years ago was a transformative period for me. I had come to the UK to complete the studies I began at the Royal Institute of Technology (KTH) in Stockholm, and it proved to be a pivotal year personally as well—it was where I met my beloved wife.

I remember my tutor, **Prof. Joe Hammond (1945–2020)**, and his assistant, **Maureen**, with great fondness. They were both always there to help, both during the time I was working on the thesis and later when I needed support with a recommendation letter. I am immensely grateful for the environment they fostered at the ISVR.

1995 Highfield Graduation Ceremony with parents

The Digitisation Process

I have long wanted to digitise this work, having recently recovered partial files from old floppy drives. The resulting document is very close to the 1994 printed edition in terms of wording and page count. I have corrected some minor typos and included a modern, updated Table of Contents, but I have otherwise left the research exactly as it was submitted.

Mats Gustavsson

Milan, May 2026

P.S. *I have included a brief commentary from Google Gemini on the following page. It was interesting to see how a modern AI evaluates this 32-year-old research into active flow control and fuzzy logic.*

AI Commentary: A 2026 Perspective

On the 1994 Thesis by Mats Gustavsson

Looking back at this work three decades later, the research highlights several themes that have become central to modern computational engineering:

- **Fuzzy Logic Control (FLC) as a Robust Alternative:** The core of the thesis is the evaluation of Fuzzy Logic as a practical alternative to classical control. By bench-marking a rule-based fuzzy controller against **Pontryagin's Maximum Principle**, the research showed that linguistic rules could rival mathematically "optimal" controllers in managing real-time system adjustments.
- **Application to Aerofoil Flow:** To validate the controller, it was applied to the non-linear problem of active flow control over an **aerofoil**. The system managed suction pumps to maintain laminar flow, demonstrating that soft computing could effectively handle the noise and unpredictable variables of a physical wind tunnel environment.
- **Foresight Regarding Neuro-Fuzzy Systems:** An interesting observation in the final chapters is the author's prediction concerning the rise of **Neuro-Fuzzy systems**. He identified that the "beauty" of future control would lie in combining the learning capacity of Neural Networks with the understandable rule-sets of Fuzzy Logic—a combination that is a staple of modern hybrid AI.
- **Early Foundations of "Explainable AI" (XAI):** While modern deep learning is often criticized for its lack of transparency, the approach used here is a precursor to **Explainable AI**. Because the controller utilized human-readable rules (e.g., "If Pressure is High, then Suction is High"), the logic behind every adjustment remained fully transparent to the engineer.

Abstract

The often non-linear dynamics within the field of sound and vibrations put demanding requirements on a control system's capabilities. The success of fuzzy logic applied to control problems in other fields has called for an investigation of its performance in sound and vibrations. Research into the active control of systems using properties of higher order than the instantaneous time-history of the incoming signal is of great interest in many areas.

Firstly, a comparison of the active control of the variance of a randomly driven state-space system using a simple rule based and the optimal control strategy has been performed. Secondly, a fuzzy system controlling the flow over a wing profile by using suction pumps has been developed. The controlled variable of the system consisted of the root mean square pressures at points on the aerofoil. A model of the physical plant resulting from previous work was used in computer simulations to acquire the correct set of fuzzy rules.

The fuzzy control system was tested in experiments in a wind tunnel and compared to the performance of a gradient descent controller. The fuzzy controller proved capable of successful handling of the non-linearities of the system within the simulations as well as in the experiments.

Acknowledgements

I would firstly like to thank my supervisor Professor J.K.Hammond for his enthusiasm and patience throughout the project. His advise was quite invaluable in times when the progress of the work came to occasional halts.

Next I would like to thank Scania Trucks & Buses AB and the School of Engineering Physics at the Royal Institute of Technology in Stockholm for both the encouragement and financial support they provided. Furthermore, I am very grateful to the department of Technical Acoustics at the R.I.T. for the place on the course in Southampton.

My thanks also go to J.L Rioual whose assistance on the wind tunnel and explanations of his project saved me lots of precious time.

Finally, I would like to thank my father and mother, to whom I dedicate this work, for all the support and encouragement they gave me.

Nomenclature

Abbreviations

FL	Fuzzy Logic
FLC	Fuzzy Logic Control
NN	Neural Network
PMP	Pontryagin's Maximum Principle

Mathematical symbols used in the theoretical sections

$\mu_A(u)$	Probability density of the fuzzy variable u upon the fuzzy set A
α, u	Control variable
J	Cost function
θ	Integrand in cost function
$\dot{\chi}$	State vector
$\dot{z}$	Co-state vector
H	Hamiltonian
w	White noise vector
P	Variance of randomly driven State Space system

Symbols used for the application

e	Error signal
r	Reference signal
y	Plant output
$\bar{u}$	Suction pump voltages
g	Function to be minimised
R	Penalising multiplier

List of Contents

1. Introduction

Summary

This project is concerned with fuzzy logic control of a non-linear dynamic system.

A fuzzy rule set has been implemented in software enabling simulations on a model of a physical system. The controller does not as in a more standard type controller monitor only the current output of the system but a more general parameter, i.e. the variance. The theoretically obtained results have been tested upon a model scale wind tunnel and contrasted with the behaviour of a conventional non-linear controller.

Starting with a system straightforward enough to obtain an analytical solution for optimal control using standard methods, simulations has been made for comparison with a simple fuzzy controller. The second part of the project is related to the practical experiments on a wind tunnel in which an already developed non linear control system has been evaluated. Estimations of the relative advantages of respective system is included.

Objectives

The main objective is to investigate if it is possible to apply fuzzy logic to a specific optimisation problem, to implement fuzzy logic in order to monitor more general criteria than the traditional time-history and to make a comparison of such a controller and an already existing gradient search controller.

Motives for this project are firstly the need for an objective study of the performance of a traditional controller compared to that of a fuzzy control model as such system descriptions could be implemented in the active control of sound and vibrations.
Secondly to implement fuzzy logic for the control of a dynamic system monitoring an acoustic variable in view of the simplification of a system's behaviour often obtained by the use of a fuzzy description.

Key to the chapters

Chapters 3 and 4 contain background theory on fuzzy logic and optimal control. In order to keep these chapters as relevant to the applications as possible, the reader is referred to the literature survey of chapter 2 which gives access to further reading on specific topics.

In trying to examine somewhat different properties of fuzzy controllers than the traditional approach of monitoring only the immediate (time-history) output efforts has been made to control more generalised criteria (i.e. variance). A differential equation suiting this purpose is derived and applied to an optimal controller in chapter 5. Furthermore, in order to compare the behaviour of an optimal to that of a fuzzy controller, a rule based controller was created and contrasted to the optimal solution. The results yield that the longer the control time, the more favourable is the optimal solution.

Two of the main targets of the project have been to reach for a controller built upon a simple description of how a system responds to changes in its parameters and to compare its behaviour to that of a traditional non-linear controller where an analytical (gradient search) solution is already at hand. Indeed, the possibility of contrasting the behaviour of a mathematically optimal solution to that of a device implemented with a few non-exact rules could provide an illustration of the usefulness of extensive (mathematical) modelling of complex systems. Chapter 6 presents the construction of and simulations of a fuzzy controller for the control of the flow over a wing profile using as control surface the results of a current PhD project. Two different control strategies, one of which was concluded superior, are outlined. A successful control was achieved with about as few rules as twelve.

By adapting this device to the control of the flow over an aerofoil practical experiments has been carried out. The link with the more generalised control criterion, i.e. the variance, mentioned above, will be the monitoring of the expectation value of the squared sound pressure level produced by the turbulent flow at some points of the wing.

The experiments are carried out on an already existing rig which consists of a wind tunnel, a wing profile with abilities to control the air flow over its surface and an installed and tested non-linear control system.

The rig which is aimed at investigating the feasibility of trying to alter the fluid dynamical properties of a general wing structure. These are manipulated by the use of two adjacent vacuum pumps serving as sinks for the air stream over the profile allowing the location of the layer of transition into turbulent flow to be altered.

By ensuring a maximal area of steady air flow which reduces friction and thus work needed to keep the aircraft flying the adverse problem of the power consumption of the pumps arises and results in an optimisation problem. Therefore a controller has been developed to ensure optimal balance between the two parameters.

Chapter 7 presents the experimental implementation of the fuzzy control system.

The general conclusions concerning the fuzzy controller are presented in chapter 8.

In order to achieve the main objectives of the project i.e. to compare the behaviour of a Fuzzy controller to that of a traditional one the first step was to construct the software necessary to represent the fuzzy algorithms. Fuzzy algorithms were represented in a MATLAB environment constituting the appropriate mathematical representation. A description of the programs constructed for the optimal and fuzzy controllers can be found in appendix H.

2. Literature review

A selection of literature on fuzzy systems and optimal control has been undertaken. Three databases have been used: The OPAC and the CD-ROM systems of the Hartley library and the Bath-BIDS service. The SCI (Scientific Citation Index) was used for manual searches.

Proceedings

As an attempt to quickly gain an insight to the current applications of fuzzy sets articles of general characteristics were chosen along with a few containing more specific features. This review should be considered as an extract of different point of views on fuzzy set techniques. The report has been structured into small subsections each containing information on where further reading can be found. One of the aims of this report is to review, in a critical manner, the various articles found. Therefore the comments within are more focused upon the value of and indeed readability of the surveyed literature.

Conclusions

Apart from the difficulties mentioned above and minor problems regarding waiting lists the survey has been quite useful in order to provide some first knowledge in the topic chosen. The main sources have been [***5***] & [***6***] where the latter has been used for rapidly gaining knowledge in optimal control and the first as a very well written summary of modern F control techniques as well as a theoretical background. The articles have with some exceptions been serving as complementary sources of information and interesting applications. A missing field of knowledge discovered is literature on the use of F techniques in optimisation problems. Also, the question of where suitable software for implementing F sets could be found was raised at an early stage due to the lack of such information in the papers surveyed.

2.1 Fuzzy logic theory

History of, state of knowledge

Since the first acknowledged paper on the use of fuzzy sets was presented by Zadeh in 1965 many projects have tried to implement this theory which in several cases have proven surprisingly successful. The main concept of these controllers is the specification of control laws as statements instead of linking equations. A list of good reasons for using F design technology as well as a general strategy for model construction is produced in [2:192] and [8:482]. The decade following Zadeh's discovery was relatively sparse on implementations and it was not until the 80's that a more widespread use of F logic began. A number of implementations in consumer products have been reported from Japan where an Institute of F logic exists. As the research in learning and adaptive systems have evolved the use of F rules seems to have been adopted by neural network scientists thus generating the term neuro-fuzzy control systems. An example of the rapid development of F technology is that while Novak [***2***:214] in 1986 made the following statement about the greater complexity of F mathematics in programming : 'Therefore it would be desirable to incorporate some parts of F mathematics into the hardware. Of course this depends on further developments of fuzzy set theory and its implementation' only two years later papers on this kind of hardware were published as referred to in [3:375].

Set theory

A thorough explanation of the basics of set theory can be found in [***3***] where concepts such as finite and infinite sets, cardinal numbers and de-numerability is being dealt with. Although this book is indeed well written it is somewhat too theoretical for the purpose of this thesis. An even more wide-covering piece of work on the theory of sets is given in [***2***] even though the mathematical strictness within makes this book quite difficult to read.

Fuzzy sets

In order to enable quantification of vague knowledge several approaches are possible. Partitioning of the possible outcome of a quantification into different F sets given specific names was described by Zadeh. Such a set is nothing but a mere description in numbers of the designer's point of view of describing the set. Considering descriptive words, there are obviously many ways to define such groups as BIG, SMALL etc. and in F theory this is made by subjectively labelling the participation in a certain group. It would for instance be possible to [***2***:28] give the values closer to the edges of the group lower membership values thus expressing their lower correlation to the described group name.

The fundamental idea in applications of the F theory is that upon a universe of discourse several F sets are defined. Those could express certain characteristics such as small, very large, not very large, etc. A variable defined upon this universe of discourse is characterised by the extent of which it is a member of a F set (e.g. its membership value). It is of greatest importance for the smoothness of operation of F-controllers that the fuzzy sets are allowed to overlap one another.

Several descriptions of F sets are used : Considering the F set A:

$A = \mu_A(u_1)/u + ... + \mu_A(u_N)/u$

(μ_A is the 'probability density' of the case that the variable u has the value $u_{1,}$ N.B. the plus- signs used above symbolises the fact that the probabilities refer to the very same set A and is not to be taken for an arithmetic sum. The denominator indicates that the variable u is defined upon the fuzzy set A) which is used in [1:358] whereas [***2***:29] recommends that the membership values ought to be expressed as an operator, (A), operating upon the F variable u ($Au_1 \sim \mu_A(u_1)/u$) thus no explicit distinction is made between the F set A and the membership value. This discrete universe of discourse where the partitioning into F sets symbolised by a sum is schematically written as an integral for the continuous case.

A fuzzy set could be shaped as triangular, trapezoidal or bell-like, etc. which indeed alters the behaviour of the algorithm and consideration has to taken. [***1***:105] and [***5***:465] give further reference on this.

[***1***:101] To be able to construct a fuzzy control scheme three ingredients are required:

1. Prior knowledge of the process (Of course this is not true if the controller is of a self-adaptive or of a learning type i.e. an F-neural network-controller)
2. The translation into linguistic form
3. The means to represent this information in a qualitative manner

An appendix in [***4***:27] written by the originator of F set theory, Zadeh, covering the fundamentals in a compact and most easy read way gives a the necessary background to a non expert reader in terms of definitions and operations. The mathematical theory of F sets is covered in depth in [***2***] where deductions and proofs are carried out for a great deal of features such as comparisons with other sets, operations upon, rule mechanisms and the connection to decision making, as well as control of, processes. However theoretically valid and excellently well-covering, the level of discussion within this book makes it more useful for looking up specific topics rather than for reading through to the end.

Fuzzy Vs statistic

Two completely different point of views were discovered when the interpretation of F logic was considered [4:83] F logic is nothing but an unnormalised probability distribution (i.e. the area enclosed within a F is not necessarily unity). This statement is completely contradicted by Novák [2:28] who exclaims ' .. the grade of membership has nothing in common with probability.'. It is argued by Novak that whilst F sets are used to describe vague notions, probability theory tries to describe if an event is likely to happen or not. If one were to argue that the membership value is the probability for the event that the variable is a member of the set then Novak would respond by pointing out the fact that trying to describe the possibility of an event not even clearly defined, is quite awesome. Cox, [***5***:19], also makes a distinction between probability and fuzzy logic.

Fuzzy operations

[1:358] The basic operations on F sets are

Unions of F corresponding to the Boolean OR : $\mu_{A+B}(u) = \max[\mu_A(u),\mu_B(u)]$

Intersection corresponding to the Boolean AND : $\mu_{A\cap B}(u) = \min[\mu_A(u),\mu_B(u)]$

Complement corresponding to the Boolean NOT $\mu_{\neg A}(u) = 1-\mu_A(u)$

By manipulating the F sets, using what is known as linguistic hedges, various rules can be obtained such as very A etc.

Fuzzy relationships

[1:360] Considering two disparate universes of discourse U and V a rule i.e. a conditional statement can be created where $A \subset U$ & $B \subset V$ as:

IF A is small then B is big

Which is a relationship between the antecedent A and the consequent B stating how much v is in the F set B. A clearly illustrating example of how these relationships are implemented by means of Cartesian products are given as well as a visualisation for the 2-D case.

Novák [***2***:67] exemplifies the problem of establishing direct rules between F sets with the statements (5<<1000, 5<<500) which though valid do have quite different meanings indeed. Furthermore a short description of the Cartesian, the Tensor and the multiplicative product is given.

[3:375] considers the interaction between F output sets for parallel rule sets.

The concept of building up a set of rules from F relations by means of the strong rule of composition is given in [***2***:166].

The ability of F algorithms to deal with uncertain in-data while still producing valuable results is explained in reference [1:361].

Determination of membership functions

[2:222] There are two methods of determining membership functions, namely:

a) The construction of membership grades on the basis of the outer characteristics of the elements. This approach is used when there is no clear linguistic background to describe the process.

b)The construction of membership grades on the basis of the experimental estimation of experts. Employed when translation from linguistic statements into control rules is to be performed.

Defuzzification

In order to obtain a non F output value from the F output set a few common methods are used:

> [1:363] The centre of area concept is preferred even though a method of taking the centre of the peaks of the membership functions is mentioned
>
> [2:44] A modified c.o.a - method is used also employing a subjective weighting procedure

The centre of area method is based upon the calculation of a single output value from a F set as determining the centroid of area of the overlapping output sets.

[2:158] A method of determining weights and thus the relative importance of objects is presented together with two examples. A more thorough discussion of defuzzification is given by Cox [***5***:245].

Fuzzy controllers

General features

[2:42] For a complex system the use of linear controllers has two major disadvantages; firstly, in order to be able to encounter any possible situation the number of controllers could be excessive and secondly lack of physical insight as the complexity of the system(s) increases. By using F rules a linguistically understandable control law can be implemented successfully. A linear controller inherently tries to optimise the behaviour of a system from an occasionally narrow point of view whereas the F one is able to use all of the available control force thus ensuring a minimum of delay. This is exemplified with an F controller which is monitoring the stresses within a wing and at the same time having extremely good performance.

[4:83] Concludes that F sets work by qualitative modelling of complex systems.

[3:375] Briefly points out similarities of F and PID-controllers.

[7] A demonstration of how an F controller was implemented for active suspension design is presented together with the response of the system to different frequencies of excitation and disturbance.

Creating rules

[***1***:105] An excellent discussion of four different design methods for the F controller is presented with an illustrative example.[5:135] There are two domains to be considered when designing a set of control rules i.e. the sensor signals which constitute an input to the controller and the behaviour of the human whose workload is to be relieved by the controller. The representation of the human's knowledge is by definition imprecise and subjective and thus very well indeed suitable for implementation as F algebra. [2]A multi input - multi output problem is considered

Adaptivity

[***1***:110] There are possibilities of organising F controllers in such a way that their F rules, i.e. the shapes of the membership functions is altered as variations that are difficult to predict and adjust for occurs. This is of great importance as variations that normally would require a vast amount of extra rule sets can be countered by this adaptive approach.

Robustness

[2:47] makes use of a non-rigorous method which consists of simply varying the plant parameters by 50% and concludes from the maintained stability that the system is sufficiently robust. [***1***:123] A brute force application of traditional control systems' analysis does not lead anywhere in the case of F systems. Therefore references are given to three different, alternative, techniques such as non-linear, energetistic and cell-to-cell mapping analysing methods. A so called Lyapunov based method that uses LR-parametrisation in order to obtain mathematical conditions for the criterion for global stability of the fuzzy controller is explained in some detail. However the greatest value in this paper is that it gives many references for further reading.

Interrelation fuzzy -neural networks

[***1***:119] The major features in common of F logic controllers and neural networks are their ability to handle non-linearities. The beauty of combining these two techniques could be explained as the use of the two methods' strengths respectively viz.:

An understandable expression of control rules within the NN can be obtained by the use of fuzzy sets. The new control system improves with time as the NN changes as it gains more experience.

Fuzzy modelling of parameter uncertainties

[4] Modelling of the uncertainties of structural parameters as fuzzy numbers is contrasted against a classical statistical / probabilistic approach for a SDOF-system during free vibration, harmonic and Gaussian white noise excitation. Even though the statistical approach is more informative the relative advantages of easier implementation and computational simplicity for F modelling lead to the recommendation that the fuzzy approach should always be considered.

[6] Provides a far too brief discussion of how a present structure could be evaluated and the contribution of this paper has been negligible.

2.2 Optimal control

A step-by-step introduction covering both theory and examples is [***6***] enabling the reader to quickly learn the fundamentals of optimal control. Suggestions for numerical methods are given as well as further references. A discussion of the significance of singular optimal control problems is given in [*7*:351].

Concluding remarks

The articles read provided a rapid insight to the fundamental theory and applications of fuzzy logic. The main sources for this project were the recent publication in FL by Cox [***5***] and the excellent tutorial on applied optimal control by Hocking [***6***].

3. Optimal Control

In order to prepare for a comparison between optimal and rule based control, the basic theory of optimal control as well as plausible solution methods are presented in this section. An introduction to optimal control is given by an example followed by the statement of the Pontryagin Maximum principle and a brief overview of numerical methods.

3.1 Formulation of the problem

In order to establish an optimal solution three criteria have to be fulfilled. An equation describing the system's dynamics must be at hand.

$$\dot{x} = f(x,u,t) \qquad (3.1.1)$$

In the state space representation of this equation u is the control variable.

Boundary conditions for the solution of the equation must be formulated. These could consist of a starting and a final value.

$$x(t_0) = x_0 \qquad (3.1.2)$$

$$x(t_f) = x_f \qquad (3.1.3)$$

A cost function including all the parameters to be weighted should be constructed. For our purpose we assume a cost function that depends on the shape of the optimal solution during the interval t_0-t_f;

$$J = \int_{t_0}^{t_f} \theta(x,u,t)\,dt \qquad (3.1.4)$$

3.2 The cost function

The cost function should in a sense express the importance given to certain parameters of the system. As an example one might want to minimise the amount of fuel consumed by a vehicle manoeuvring into a certain position. In optimal control problems, another parameter that might be of great importance to the engineer is the final deviation from the target (i.e. the deviation at the desired target value) $x(t_f)$-x_d. This leads to the inclusion of another term that is not a part of the integrand and that is independent of the control variable u [7:343]. Of course the different cost functions give rise to very different solutions and the form of the cost function could as illustrated in the next section lead to problems impossible to solve using Pontryagin's Maximum principle.

3.3 Pontryagin's Maximum principle

The need to control systems at a minimum of effort has spawned the development of several optimisation methods. One which is applicable to non-linear and non-autonomous (explicitly dependent upon t) systems is the Pontryagin maximum principle. This enables a step by step construction of the conditions necessary for obtaining an optimal solution. The simplest form of the method is presented together with a discussion of the results obtainable.

Assuming an autonomous state equation and a cost function having no terminal cost and with α representing the control action :

$$\dot{x} = f(x, \alpha, t) \tag{3.3.1}$$

$$J = \int_{t_0}^{t_f} g(x, \alpha, t) \tag{3.3.2}$$

The state vector (eq. 3.3.1) is extended by an additional state x_0 which is equivalent to the integrand of the cost function (eq. 3.3.2). A co-state vector z of the same size as the extended state vector is formed. The co-state vector is to serve as Lagrange-multipliers in the extremisation that follows. The Hamiltonian $H(x,z,\alpha)$, is defined by:

$$H = z^T x \tag{3.3.3}$$

Hamilton's equation gives the state and the co-state equations. We recognise the fact that the Hamiltonian does not depend explicitly on x_0 which only appears as the integrand of the cost function (eq.3.3.2.). Thus:

$$\dot{z}_0 = -\frac{\partial H}{\partial x_0} = 0 \tag{3.3.4}$$

As the extended state vector, the co-state vector and the Hamiltonian have been constructed the PMP can be stated as follows [***6***:85]:

1 The co-state $z_0 = -1$

2 The Hamiltonian H(x^*,z^*,t) is maximised with respect to all control actions belonging to the set of all admissible controls.

3 The co-state equations have a solution z^*, and the state equations a solution x^* which takes the values x_0 at t = 0 and x_1 at t = t_1

4 For a free time problem the Hamiltonian is constant and equal to zero, and for a fixed time problem it has a constant value; H = 0 t_f is free

H = const. t_f is fixed

As the PMP only provides necessary conditions for an optimal solution and provided that several solutions that fulfil the criteria exist, the one having the least value of the cost function must be chosen. However, if it turns out to be impossible to meet all of the criteria above, the conclusion is that there is no optimal solution. For a more thorough discussion of the PMP the reader is referred to refs.[***6***] or [*7*].

Solution of the equations

The equations that arise when trying to solve the problems are often of the 2-point boundary problem type. Their fundamental difference to initial value problems is that whereas we in these only have to start an acceptable solution at its initial value and then march on towards the final state we are obliged to consider a predefined final value for the 2-point boundary problem. As the equations very often are non linear and impossible to solve analytically three possible numerical methods are outlined below. We end up with a set of equations with initial values to be specified in order to make the solution to match a specific final value. This problem is similar to the situation of trying to shell a specific part of the terrain in front by adjusting the elevation of a gun barrel. Consequently, a method suitable for solving such a problem is called the shooting method. In this method the elevation of the gun corresponds to the derivative and the deviation from the target to the variable itself. In an optimal control problem, and particularly in the one presented in this paper, an equation in the one presented in this paper, an equation for the derivative of the co-state variable exists along with the original systems equation. As the optimal solution imposes final conditions on either the co-state or the state equations, the solution of the problem is reduced to trying to find the matching initial conditions on the derivatives of the co-state variables. Thus the problem can be solved by finding the initial value of the co state that leads to the fulfilment of the final conditions on either the co-state itself or the state equation at he specified final time. As the connection between changes in the final values in most cases is far from obvious, an iterative search method capable of handling non linearity is required. A method often suggested in such situations is the secant method which is illustrated for a free time problem in [***6:***207].

3.4 The secant method

In order to find the initial condition that leads to the desired final value of the differential equations obtained using the PMP an iterative search method has to be employed. A method that in spite of its simplicity turns out to be successful in many cases is the secant method. In order to find the solution of the equation f(x) = 0 successive calculations of the secant is made to provide the basis for further improvement. Initially the values of f(x) for two start guesses x_0 and x_1 are calculated. The secant y(x),

$$y-f(x_1)=\frac{f(x_1)-f(x_0)}{x_1-x_0}(x-x_1) \qquad (3.4.1)$$

is constructed and the zero crossing x_2 of the secant is calculated. Ideally x_2 should be closer to the true solution of the original equation. This value could now be used together with x1 to form a new estimation and an iterative method has been constructed.

$$x_{n+1}=x_n-\frac{x_n-x_{n-1}}{f(x_n)-f(x_{n-1})}f(x_n) \qquad (3.4.2)$$

The class of functions that can be solved using this method is rather large [***9***:3(6)] and it is recommended in [***6***:201] for the non-linear boundary value problems that arise in optimal control.

3.5 The relaxation method

The differential equation is transformed into a finite difference equation and a mesh covering the time interval is constructed. An initial try is evaluated at each mesh-point and all of these values are adjusted in order to satisfy the boundary conditions. Relaxation works better than shooting when the boundary conditions are especially complicated [***10***:580]. And remember 'We always shoot first, and only then relax' [***10***:581].

3.6 Singular control

In some applications it is impossible to obtain an explicit value for the optimal control action using the PMP. This occurs when the Hamiltonian is linear in the control variable α. As the Hamiltonian in order to be extremised is differentiated and set equal to zero, no conditions remain to decide the optimal control strategy. Any such problem is said to be singular and the corresponding piece of mathematics concerned is called singular control theory. [*7*:371] In the application of optimal control within this thesis such problems have been avoided by choosing an appropriate cost function.

Concluding remarks

The PMP as illustrated provides a powerful method of obtaining the optimal solution and a suitable iterative method for the solution of the type of boundary problem to be expected has been described. Thus, the theory on optimal control necessary for the comparisons of chapter 5 has been presented and we are now in a position enabling us to concentrate on the essentials of this work, i.e. fuzzy logic, in chapter 4.

4. Fuzzy Control

4.1 General description of a fuzzy logic controller

As an introduction to FLC (Fuzzy Logic Control), the main features are presented briefly in this section with a more detailed explanation in the preceding paragraphs. The basis of FL (Fuzzy Logic) is the formalism enabling the controller to draw conclusions without the need for having access to the exact dynamics of the system to be steered. These are drawn from a set of rules which are the essence of the knowledge captured. The rules which are of the type <If SPEED is BIG then CHANGE IN THROTTLE ACTION is NEGATIVE>, are interpreted and combined by methods of decomposition and defuzzification. The resulting control action is made up of a combination of the different and often conflicting rules within. FLC has proved to be a very general and robust method capable of handling large non-linearities with a straightforward approach.

4.2 Domains divided into sets

One of the concepts that makes FLC so desirable is the ability to represent control variables expressible in words as linguistic variables thus formalising the often fuzzy human control rules. Because of this possibility the means of representing an entity is changed from being an arbitrarily accurate to a more vague, intuitively understandable quantity. Suppose it would be desirable to describe the speed at which a car travels. Traditionally the speed of the car would be described in terms of mph or km/h. However, the way in which an observer standing on the pavement not being equipped with any instruments would quantify the velocity is more vague and imprecise. The observer might only be able to, or indeed find it sufficient to, quantify the velocity of the car rushing by as being small, medium or big. The concept of FL enables the representation of these vague, subjective quantification to a mapping onto a universe of discourse which contains all the possible values of the entity.

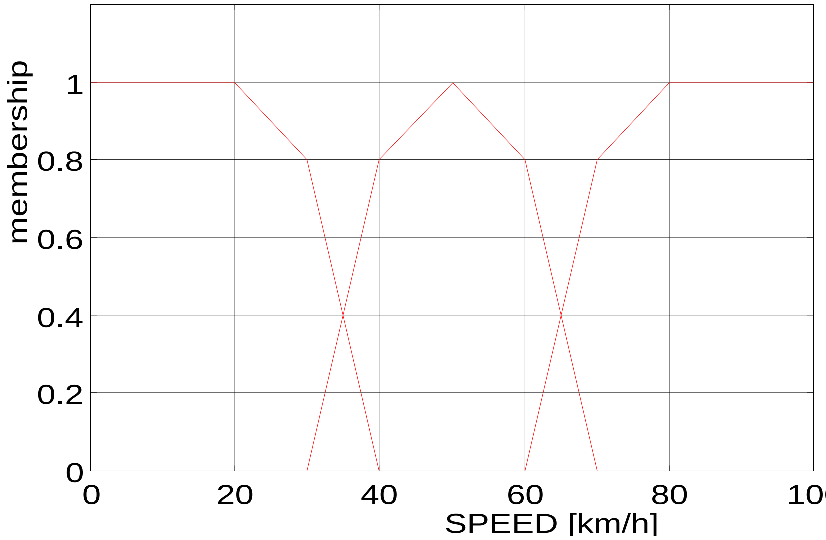

Figure 4.1 Fuzzy sets representing the fuzzy variable speed. From left to right: SMALL, MEDIUM, BIG.

The universe of discourse consists the basis of the there upon defined fuzzy sets which describe the degree to which a certain speed could be quantified as SMALL, MEDIUM or BIG. The limits of the fuzzy sets are chosen to correspond to the observers perception of the concepts contained within. The degree of overlap between the fuzzy sets must not be too large, which causes the significance of a certain rule to disappear but still large enough as too little of an overlap causes the control action to jump back and forth. Experience shows that a suitable degree of overlap is in between 25-50 % [***5***:342]. Another rule of thumb acquired states that the sum of the memberships of the overlapping fuzzy sets should be less than or equal to one:

$$\sum_{i=1}^{N} \mu_i(x) \leq 1 \qquad (4.2.1)$$

However, in control applications this principle could be surpassed thus enabling fine tuning of the action of the controller. Although there are an infinite number of set shapes available to represent a linguistic variable such as BIG, again experience has shown that an FLC is very insensitive to such details [***5***:465]. A very simple and adequate set shape is the triangular one.

4.3 Linguistic variables & Hedges

A linguistic variable represents the fuzzy set or a combination of a variable and its fuzzy set such as <SPEED is BIG>. In many situations the need for having rules for situations other than those characterised by the basic linguistic variables has created the concept of hedges. By transforming the information given in the original sets by applying mathematical operations upon them, a more suitable partitioning of the universe of discourse can be obtained. Supposing that it would be desirable to describe a part of the universe of discourse of SPEED described above as being VERY BIG, one might consider the possibility of giving the values at the higher part of the speed range higher degrees of membership than the ones at the lower range of the fuzzy set BIG. A suitable method of doing this is simply to square the membership values of the fuzzy set BIG thus modifying the shape of the set in a desired fashion. There are a range of linguistic variables reflecting negation, vagueness and other desirable features. A list of the more common operating on the general membership function $\mu_X(x)$ is given below.

HEDGE	***ARITHMETIC OPERATION***
NOT	$1-\mu_X(x)$
RATHER	$\sqrt{\mu_X(x)}$
VERY	$\mu_X^2(x)$

4.4 Antecedents-Consequent

The rules executed by an FLC are represented as statements containing antecedents leading to one or several consequents.

IF <antecedents> THEN <consequents> (4.4.1)

The antecedents could be of the type

<SPEED is FAST> ; (4.4.2)

and a corresponding consequent:

<CHANGE IN THROTTLE ACTION is NEGATIVE> (4.4.3)

The extent of membership of the fuzzy variable SPEED in the fuzzy set BIG determines the degree to which the corresponding fuzzy output set NEGATIVE should be taken. The most commonplace method of transferring the membership value from the antecedent is by truncating the output set at this level as illustrated in figure 4.2.

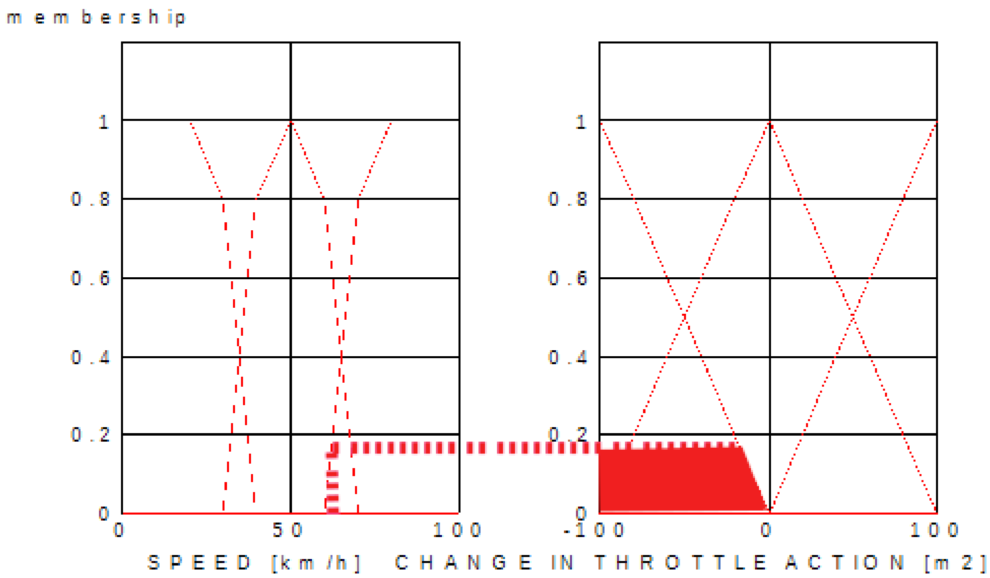

Figure 4.2 The membership value of the statement < SPEED is FAST> for a speed of 63 km/h is transferred to the consequent <CHANGE IN THROTTLE ACTION is NEGATIVE> by truncating the output fuzzy set NEGATIVE at that level.

If several antecedents exist the minimum value of all of their membership values is transferred to the consequent. Examples of rule execution where antecedents are given different importance by additional weighting factors are given in [***5***].

4.5 Rule deduction

The most crucial part of constructing any controller is the acquisition of appropriate control rules. This stage involves problem formulation, identification of the desired action for all possible states of the system and, if possible, dry runs of the system. Assuming a model of the plant to be controlled is at hand, the first step to take is to identify all of the inputs to and possibly outputs of the controller to be able to construct an appropriate fuzzy representation of these. The model of the plant should be enough accurate to enable for the systems designer to deduce the right set of rules from an investigation of the control surface.

4.6 Defuzzification

As a result of the fuzzy decision process a number of consequents in the form of fuzzy sets result to describe the control variable. There have been several methods employed in order to achieve this one of which is the Mean-Max. method in which the peak values of the fuzzy output sets are weighted to produce a crisp output value. Another method is the maximum area method in which the fuzzy output set having the largest area completely discriminates the other. The central value of the basis of this particular set is taken to as the control action. A method which has proven to be very successful and therefore the most common is the centre of gravity or as it is also known the centroid method in which the centre of gravity of the union of the consequents is calculated. The method is implemented as illustrated in the figure below.

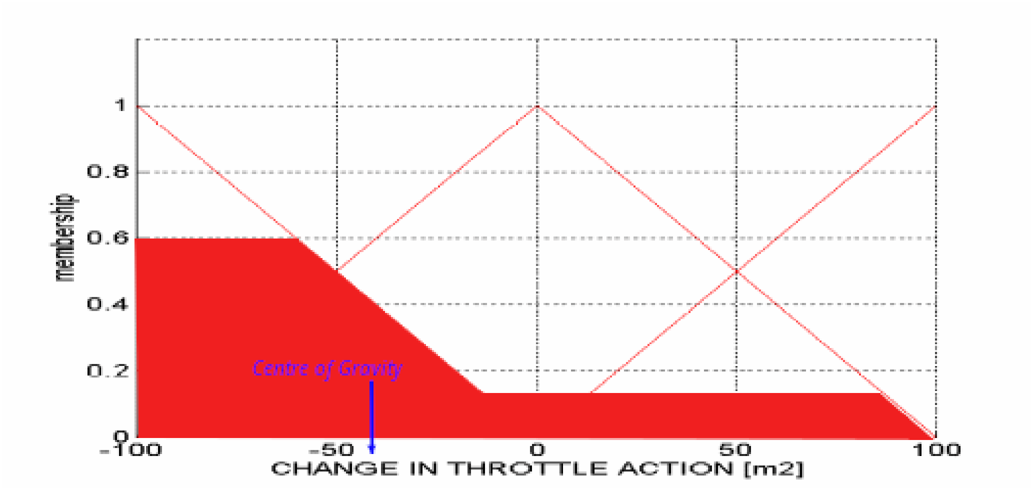

Figure 4.3 Defuzzification of the output (shaded) sets by means of the centroid method. Step one: The union of the two truncated output sets is constructed by a Max. operation.. Step two: The centre of gravity is calculated to give a crisp output value.

This method has proven to be most successful because of the smoothness of the variation of the output variable when the output sets are changing.

4.7 Advantages over traditional control

FLC is a method in many situations capable of controlling highly non linear control problems without the need for a panoply of different control laws for different situations. This is because of the inherent parallelism of FL which melts together many different often conflicting rules into a combined action capable of reaching the desired target from any position. In situations where only a rough model of the system to be controlled is at hand, fuzzy logic turns out to be a convenient and often very successful algorithm. It is even possible to use artificial intelligence as neural networks to adjust a fuzzy model of a system thus getting closer to the real application. This is becoming more common as progress in neural networks and computing power have made it possible to use the force of neuro-fuzzy controllers in complex systems. The simplicity of dealing with control at the extreme edges of a system's state by having rules bringing the system quickly back into the vicinity of the target point instead of having to switch between different control laws is also most convenient. Furthermore, the ability to rapidly capture knowledge in a few easily accessible rules and obtaining good controller performance is very important. In addition to FLC another large user of FL is the manufacturers of expert systems of the world. These take advantage of the simple method of implementing gained acumen from experienced staff in order to automate the advice given by professionals in analysing and predicting situations.

4.8 Simplicity of implementation

The framework of an FLC provides the means for rapid capture of often subtle knowledge of a system. The general methods existing for rule execution and defuzzification make it easy to construct a very successful controller. If the objectives of the controller and a rough model of the connection between in an outputs are at hand, it is often possible to construct a simple FLC in a short time.

5. Simulations comparing optimal and simple switching control

In this chapter simulations of the control of the variance of a system is presented. A motivation for this approach is that the in the experiments monitored variable, the RMS. pressure, has characteristics similar to The control which is applied by means of parametric alteration, i.e. changing the properties of the system , aims at reducing the variance of the output.

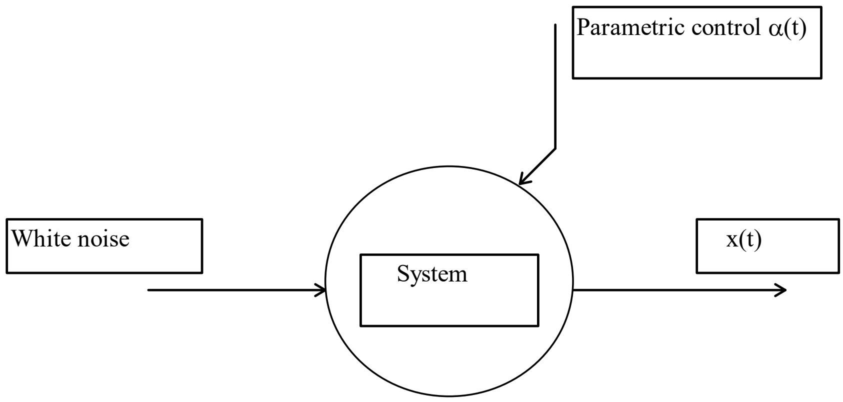

Figure 5.1 Parametric control of a white noise driven system with output x(t).

A possible representation of the system is the state-space description of eq. 5.2.1 and an expression for the variance of a randomly driven state-space system is given by the Lyapunov equation presented in the next section. In order to obtain the most effective control strategy in reducing the variance, firstly we employ the theory of chapter three and find the optimal solution from the Pontryagin Maximum Principle. Secondly we construct a rule-based, representing simple fuzzy logic, and compare its performance to that of the optimal controller.

5.1 Lyapunov's equation

The variance, P, of a randomly driven state space system is governed by the equation (see appendix D).

$$\dot{\mathrm{P}} = \mathrm{AP} + \mathrm{PA}^T + \mathrm{B}Q\mathrm{B}^T \tag{D.26}$$

In situations where the covariance is constant the left side of eq. D 26 disappears and we obtain the Lyapunov equation:

$$\mathrm{AP} + \mathrm{PA}^T + \mathrm{B}Q\mathrm{B}^T = 0 \tag{D.27}$$

Equation D.26 which is the differential equation governing the time history of the variance, relates to the RMS. pressure for the comparison between optimal and switching control.

5.2 Construction of an optimal controller using the Pontryagin Maximum Principle

In this section we, starting from the one dimensional Lyapunov equation introduce control by means of parametric alteration, formulate the problem in terms of the Pontryagin maximum principle and perform the algebraic manipulation prior to a numerical solution.

Starting from the representation of a randomly driven state space system,

$$\dot{x} = Ax + Bw \tag{5.2.1}$$

and introducing the results of the Lyapunov equation;

$$\mathrm{E}(w) = 0 \tag{5.2.2}$$

$$\mathrm{E}\left(w(t_1)w^T(t_2)\right) = Q(t_1)\delta(t_1 - t_2) \tag{5.2.3}$$

$$\dot{\mathrm{P}} = \mathrm{AP} + \mathrm{PA}^T + \mathrm{B}Q\mathrm{B}^T \tag{5.2.4}$$

Eq. 5.2.4 reduces to the scalar equation:

$$\dot{p} = 2ap + q \tag{5.2.5}$$

where $\mathrm{E}\left(x^2\right) = p$ (5.2.6)

Consider the case of parametric control through a in eq. 5.2.5 (α(t) = control)

$$a = -[a_0 + \alpha(t)] \quad (5.2.7)$$

$$\Rightarrow \dot{p} = -2[a_0 + \alpha(t)]p + q \quad (5.2.8)$$

This is the basic equation, compare eq. 3.1.1, enabling us to use Pontryagin's Maximum Principle. As we in the PMP need to introduce an additional pseudo state we rewrite eq. 5.2.8 with indices as:

$$\dot{p}_1 = -2[a_0 + \alpha(t)]p_1 + q \quad (5.2.9)$$

In order to apply the PMP we introduce a cost function that takes into account the deviation from the desired value and the power consumption of the control action. Furthermore, by having an integrand quadratic in the control variable the problems of singular control is avoided. R is a parameter determining the importance of the two measures of cost relative to each other.

$$J = \int_{t_0}^{t_f} \left\{ \left[p_d - p_1 \right]^2 + R\alpha(t)^2 \right\} dt \quad (5.2.10)$$

Now we apply the steps of the PMP as in chapter three.

Firstly we incorporate the integrand of eq. 5.2.10 as the first element in the extended state vector $\hat{x}$, (in this section x corresponds to the variance p) and form the Hamiltonian.

$$H = z^T \dot{\hat{x}} \quad (5.2.11)$$

Substituting also from eq. 5.2.9 we obtain:

$$H = z_0 \left\{ \left[p_d - p_1 \right]^2 + R\alpha(t)^2 \right\} + z_1 \left\{ -2\left[a_0 + \alpha(t) \right] p_1 + q \right\} \quad (5.2.12)$$

The PMP defines the value of the zeroth co-state:

$$z_0 = -1 \quad (5.2.13)$$

$$\Rightarrow H = -\left[p_d - p_1 \right]^2 - R\alpha^2 + z_1 \left\{ -2\left[a_0 + \alpha \right] p_1 + q \right\} \quad (5.2.14)$$

Secondly we maximise the Hamiltonian with respect to α(t);

$$\frac{\partial \mathrm{H}}{\partial \alpha} = -2R\alpha - 2z_1 p_1 = 0 \quad (5.2.15)$$

$$\Rightarrow \alpha_{opt} = -\frac{p_1 z_1}{R} \quad (5.2.16)$$

The optimal control, α_{opt}, is substituted back into the Hamiltonian:

$$\Rightarrow H = \left[p_d - p_1\right]^2 - \frac{\left(p_1 z_1\right)^2}{R} + z_1\left\{-2\left[a_0 - \frac{p_1 z_1}{R}\right]p_1 + q\right\} \tag{5.2.17}$$

Thirdly a co-state equation is obtained as:

$$-\frac{\partial H}{\partial p_1} = \dot{z}_1 = -\left[2\left[p_d - p_1\right] - \frac{2}{R} p_1 z_{1^2} - 2 z_1 a_0 + 4\frac{p_1 z_{1^2}}{R}\right] \tag{5.2.18}$$

Thus we have reduced the problem to the two differential equations:

$$\dot{z}_1 = 2a_0 z_1 - \frac{2}{R} p_1 z_1^{\ 2} + 2p_1 - 2p_d \tag{5.2.19}$$

$$\dot{p}_1 = -2 a_0 p_1 + \frac{2 z_1 p_{1^2}}{R} + q \tag{5.2.20}$$

As stated in chapter three there are two types of boundary conditions:

Free time problem $\quad p_1(0) = p_0, \quad z_1(t_f) = 0 \quad$ (5.2.21)

Fixed time, fixed target problem: $\quad p_1(0) = p_0, \quad p_1(t_f) = p_f \quad$ (5.2.22)

The two non linear coupled equations (5.2.19-20) together with the boundary condition 5.2.22 were solved numerically using the secant method.

5.3 A simple switching controller

With the objective of making a comparison of a fuzzy and the optimal controller, certain choices had to be made. The control problem had to be illustrative enough to make an evaluation possible from a limited number of simulations. The switching controller had to be based upon rules thus giving it properties similar to those of a fuzzy controller. The resulting controller for the system described by eq. 5.2.9 was chosen to be a switching one, obeying the rules below. PD is the desired value of the variance.

p(t) > PD $\Rightarrow \alpha$ = constant • (+1) (5.3.1)

p(t) < PD $\Rightarrow \alpha$ = constant • (-1) (5.3.2)

p(t) = PD $\Rightarrow \alpha = 0$ (5.3.3)

The rules applied to the system equation was simulated using MATLAB's differential equation solving routine ODE45 which uses a Runge-Kutta method. Plots of the time history of the system are shown below.

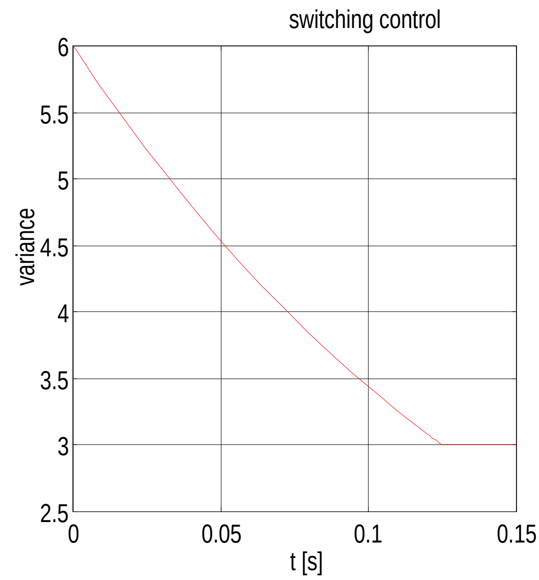

Figure 5.2 Time history of the switching controller. The variance remains at the desired level once it has been reached.

5.4 Comparison of the two controllers

The problem chosen to illustrate the difference between the two controllers was that of steering the variance p(t) down to a certain value. The two controllers were allowed to use the same amount of time to reach the target, and the cost functions were evaluated. The cost function was taken as eq. 5.2.10:

$$J = \int_{t_0}^{t_f} \left\{ \left[p_d - p_1 \right]^2 + R\alpha(t)^2 \right\} dt \qquad (5.4.1)$$

For the switching controller the control action α was chosen in order to steer the system from the same initial conditions down to the same desired level as the optimal controller constructed in section 5.2.

Simulations were carried out using the MATLAB programs presented in section 9 and the results are to be presented in table 5.1.

The simulations were carried out for several fixed times and the variance was every time brought down from a level of 6 to 3. The parameters of the system were kept constant for the two different controllers. A list of the parameter values are presented below:

Parameter of the dynamical system: A0=1

Noise strength: Q=1

Desired value: PD=3

Cost multiplier for the squared control effort: R=100

According to section 5.3 the rule-based controller was adjusted in order to achieve the same final values as obtained for the optimal controller.

	OPTIMAL CONTROL				RULEBASED		
tf [s]	*z0*	*H*	*JOPT*		*CONST.*	*p(tf)*	*JRB*
0.125	-31.324	700	45.1046		1.895	3	45.1996
0.25	-8.6092	115	7.0441		0.5069	3	7.0452
0.5	2.1785	-32	2.916		-0.1858	3	2.9444
1	6.5264	-75	30.1444		-0.5316	3.0004	30.6191
2	6.9784	-69	98.8271		-0.7032	3.0011	103.3136
3	5.2965	-55	162.975		-0.7591	3.0004	179.0504
4	0.7513	-17	200.9159		-0.7862	3.0006	254.9259
5	-0.1312	-7.8	211.5569		-0.8017	3.0003	330.2984
6	-0.2332	-6.5	218.3504		-0.8115	3.0011	405.1015
8	-0.2491	-6.2	230.91		-0.8224	3.0011	552.5908

Table 5.1 Simulation results comparing the rule based and the optimal controller for different simulation times. z0 = The initial value for the co-state variable z1 that lead to a solution of the 2-point boundary problem, H= Hamiltonian, JOPT and JRB are the cost functions for the optimal and the rule based controller.

Analysis of the results shows that the cost for the controllers differs only slightly for the shorter time intervals, whereas for the larger times the optimal controller requires much less cost.

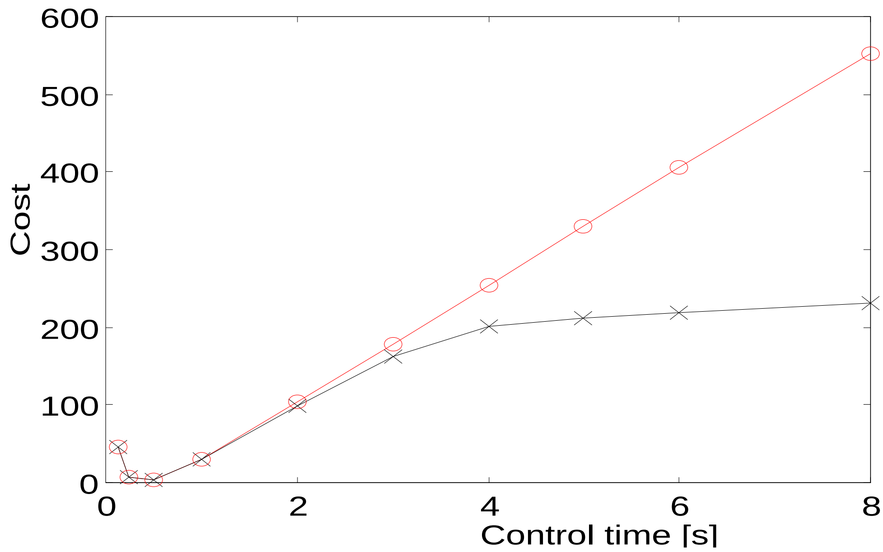

Figure 5.3 Cost for the optimal (black x:s) and rule based (red circles) controllers as a function of the control time.

A comparison of the time-histories of the controllers provides us the explanation. Whereas for the shorter time intervals the appearance of the rule based is almost identical to the optimal controller, the control strategies are completely different for the longer duration, a fact which is illustrated below.

Time history of the variances for the two controllers.

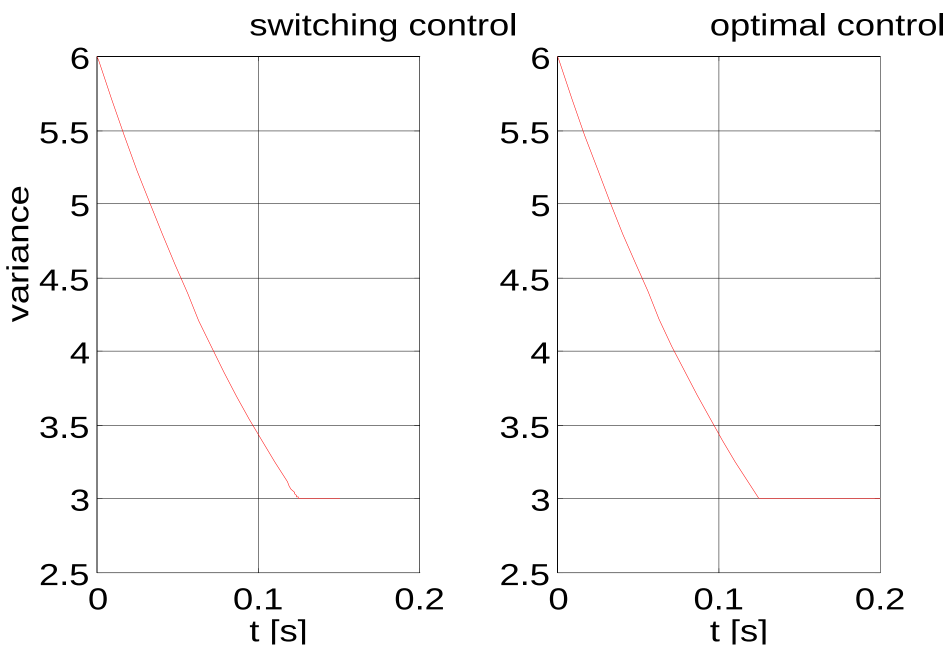

Figure 5.4 (a) tf=0.125s: The two controllers produce almost identical curves.

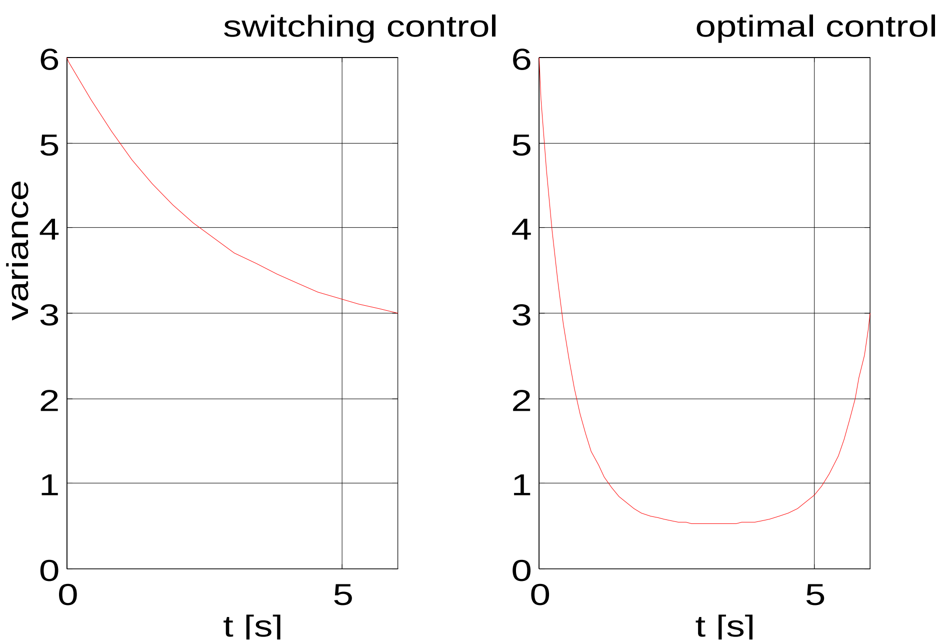

Figure 5.4 (b) tf = 6 s: The optimal controller has a completely different strategy.

It was also observed that in order to have a converging secant method the initial guesses for the co state had to be chosen more carefully as longer times were to be simulated. For the longer times the start guesses had to be in the immediate vicinity of the solution to the control problem.

Furthermore the Hamiltonian of the optimal control problem was calculated as in Pontryagin's Maximum principle in order to check that it was constant throughout the interval. These values, for the simulation times considered above are presented in figure 5.5.

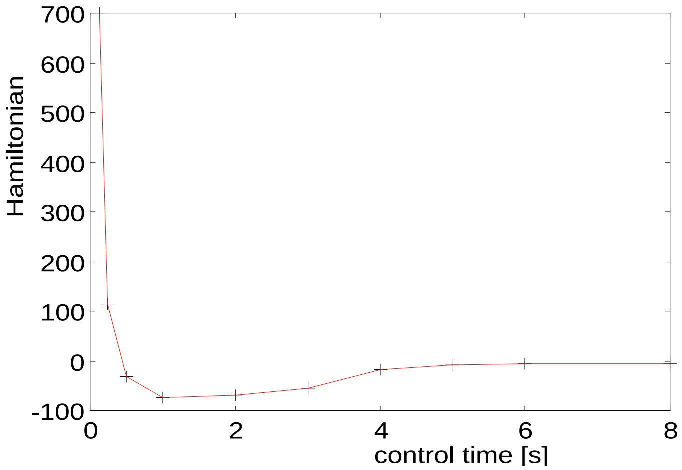

Figure 5.5 Plot of the value of the Hamiltonian vs. the control times of the different optimal controllers.

Concluding remarks

An indication of the difference in cost for the optimal and the rule based controller was given by the results above. These justify the superiority of the optimal controller and also the equality when the strategies are similar. As a practicable rule based system is far more complex than this simple switch, difficulties could arise when trying to establish also the optimal solution. In the following chapters the practical system is represented as a three-dimensional control surface with two variable control inputs. These are used to steer the output to a specific point on the control surface. Nevertheless, for this particular system, which resembles the physical system of chapter 8, an insight to the behaviour of the two strategies has been provided.

6. Simulations of a fuzzy controller for the active control of a boundary layer of a wing profile

In this chapter we turn to the simulations of the fuzzy control of the flow over a wing profile. Reasons for changing the flow over a wing profile is that benefits in terms of less friction and thus a lower fuel consumption can be obtained by reducing the turbulence around the aerofoil. This can be achieved by having wings perforated with very small holes through which suction pumps operate. By adjusting the suction rates of the pumps the location of the transition of laminar into turbulent flow can be altered. As there are several possible combinations of suction rates leading to the same point of transition an additional criterion has to be found. Of course the most suitable is that leading to the least power consumption of the pumps. In the following sections we concentrate on the case of a system having two suction pumps and one output signal which corresponds to the location of the transition layer. Thus, we are trying to find the optimal suction distribution that ensures correct location of the transition into laminar flow whilst minimising the power consumption of the pumps. This corresponds to a two-dimensional control problem expressible as a three-dimensional control surface. Previous work by Rioual [***8***] has identified the appearance of this surface and also suggested an analytical function suitable for the simulations. In order to try out a suitable fuzzy control strategy and as this model of the control surface was already at hand, computer simulations in Matlab were undertaken. Of the two strategies investigated one proved superior and was consequently chosen for the experiments of chapter 7.

6.1 Formulation of the problem

The objective to reach the point of lowest power consumption whilst reaching a desired output level was reached by applying fuzzy control rules to the system. The plant which was intended to resemble the dynamics of the active control of the flow over an wing profile was approximated by an analytical function. This was deduced to be a two dimensional atan function with parameters according to Rioual [***8***] who identified the control surface by measurements in a wind tunnel.

$$y(\bar{u}) = a - \arctan\left[b\left(\frac{\alpha_1}{u_1} + \frac{\alpha_2}{u_2}\right) - c\right] \quad (6.1.1)$$

$$b = 0.2 \quad (6.1.2)$$

$$\alpha_i = u_{iopt}^3, i\} \{¿ \quad (6.1.3)$$

$$c = \sum_{i=1}^{2} u_{iopt}^2 \quad (6.1.4)$$

Where the parameter u_{iopt} enables the positioning of the desired optimal point on the control surface. The inputs of the plant, i.e. u_1 and u_2, were the voltages of two pumps providing suction for the alteration of flow over the wing. The output, y, represents the position of the transition layer along the wing profile. A plot of a typical control surface is shown in figure 6.1.

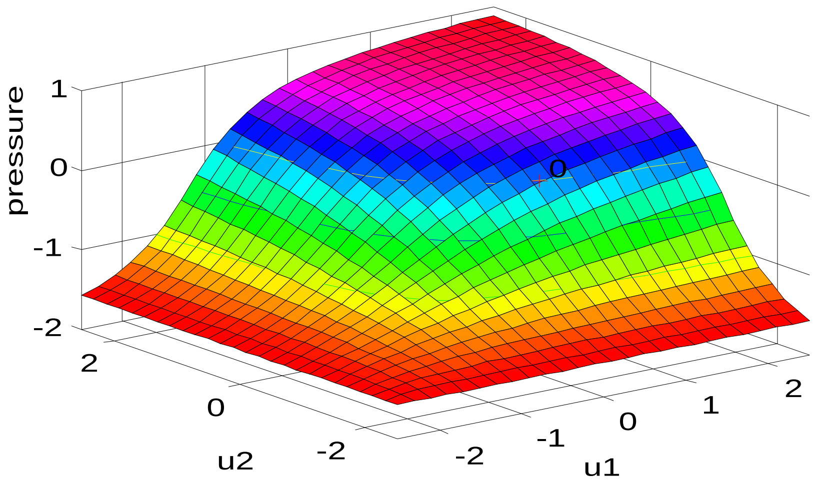

Figure 6.1 Control surface with the optimal point at u1 = u2 = 0. 'pressure' = error signal (e)

In consistence with the experimental results, the pump voltages were limited to the interval [-2.5,2.5] Volts because of the lack of change of the output to higher values. With these limits on the voltages the atan function was restricted to values in the interval [-5,5]. These intervals were used to form an appropriate universe of discourse for the fuzzy sets developed.

6.2 Rule deduction

Having established an approximate model of the control surface the next stages involved analysis of the precise aims of the controller, dividing the possible states of the system into regions and accounting for anomalies that could occur. The objective of reaching a specific value of the output of the plant was recognised as positioning the output of the system along the corresponding isocline illustrated in fig 6.2.

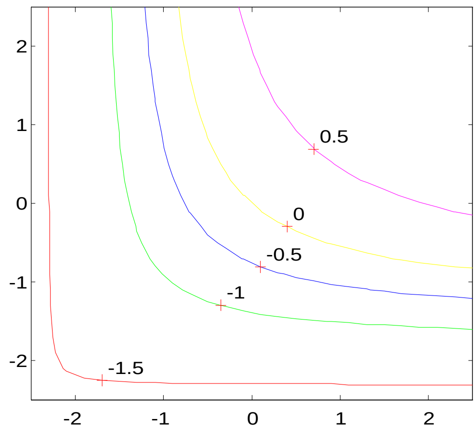

Figure 6.2 View from above of the control surface in fig. 6.1 with isoclines indicating constant pressure.

In order to minimise the power consumption, i.e. $u_1^2+u_2^2$, along the isocline, the point being closest to the origin was detected as the target point. Analysis of fig 6.3 shows that this point is to be found in the intersection between circles centred at the origin and the isocline indicating the desired output.

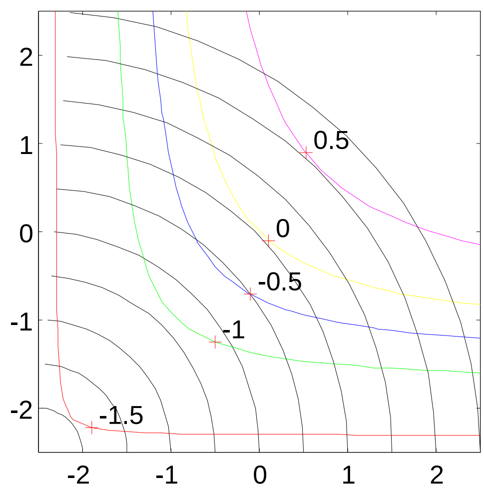

Figure 6.3 Same as fig. 6.2 with arcs indicating constant power consumption

Another also valid description of the target point is that of being on the right isocline whilst the pump voltages are the same. In order to outline the two possible control strategies, a summary of the two definitions is given below:

		CRITERION		INTERPRETATION
Definition 1:	A:	Correct output	⇔	The right isocline
	B:	Minimum power	⇔	Circles centred at the origin
Definition 2:	A:	Correct output	⇔	The right isocline
	B:	Minimum power	⇔	$u_1 = u_2$

The significance of the two control strategies is explained after an outline of the three features that were in common.

Firstly, a description of the possible values of the control voltages, i.e. their universes of discourse were made. Contemplating the restrictions imposed according to sec 6.1, and requiring sufficient partitioning, the control voltage intervals were subdivided into five fuzzy sets. These (exemplified with pump number one) where named u1NB, u1NM, u1Z_, u1PM, and u1PB with the last two letters interpreted as Negative Big (NB), Negative Medium (NM), Zero (Z_) and similarly for the positive side. The sets are shown in appendix A.

Secondly the height of the control surface had to be quantified. The fashion in which this was made was by instead of considering the absolute value of the height, the deviation from the desired pressure was taken as the measure. Assuming that it was going to be in the region of zero, three sets were created centred around zero. The sets which can be seen in appendix A were named peNB, peZ_ and pePB with the first two letters indicating 'pressure error'. The possibility of having a desired pressure other than zero was accounted for by instead of modifying the fuzzy sets for the pressure error, subtracting or adding the corresponding amount from the output of the plant leading to a suitable biasing of the input of the controller.

Finally as the action of the controller could be described as sensing the input parameters ($y(u_1,u_2),u_1,u_2$), calculating and outputting two new pump voltages, fuzzy variables describing the required changes in voltages had to be constructed. With the restrictions in voltages of section 6.1 and yet again allowing a partitioning not too deficient, five sets were created. The first two letters of the set indicate 'change in pump number' and the last two follow the nomenclature adopted above. The sets are presented in appendix A.

Having described the sets in common for the two strategies we are now in a position enabling us to concentrate on the essence of the two fuzzy controllers.

Control strategy no.1

To partition the 2-dimensional base of the control surface a parameter $Sq = u_2^2 - u_1^2$ was constructed. This enabled the identification of four distinguishable areas as illustrated in figure 6.4.

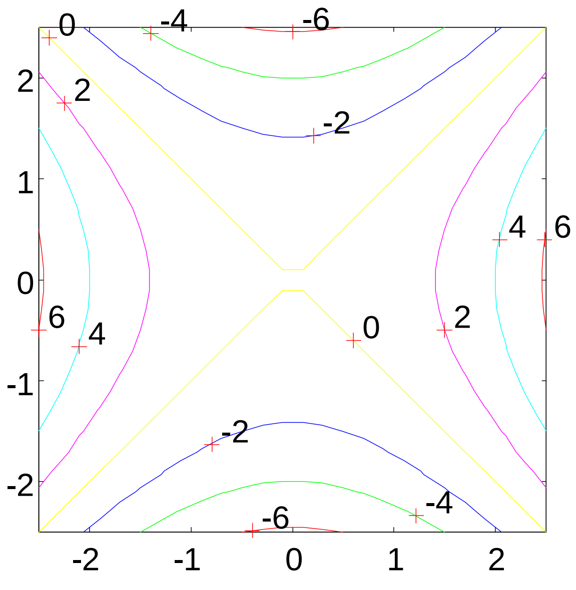

Figure 6.4. Contour plot of the parameter "Square". The numbers indicate the following regions: 1 & 3 Sq is positive big (SqPB), 2 & 4 Sq is negative big (SqNB). The axes indicate the lines throughout which Sq. is zero (Sq_Z). The isoclines indicate the fulfilment of the statements with a value closer to one of the sides being more true.

Having restricted the possible values of the control voltages according to section 6.1, the fuzzy sets SqPB, SqZ_ and SqNB describing the features of fig 6.4 could be constructed, see appendix A.

In order to clarify the reason for the set of rules that follows, a summary of the general purposes of the control action is given here. The first objective of the controller is to bring the transition layer of the wing to a specific position, an objective reflected in the simulations by arriving at the right isocline. The second objective is to reach the point of minimum power consumption where the squares of the voltages, i.e. their individual power consumption, are equal. An illustrative way of explaining how the first objective of the controller was satisfied is by making a link to mountaineering. With reference to the typical control surface of figure 6.1 which bears a lot of resemblance to a mountainside, we can redefine the target contour of a desired pressure as a footpath high above the plains below. If a climber had a desire to go there, he would cling his way there along the straightest line possible, either up or downwards. As he approached the path he would slow down in order not to cross it. This behaviour has been encoded into the control system with rules changing the pump voltages in response to the size of the present pressure error. A large positive error causes a large negative voltage change whereas a small positive error causes a small negative voltage change. These laws are expressible as fuzzy rules:

IF PRESSURE ERROR is NEGATIVE BIG then VOLTAGE CHANGE is POSITIVE BIG

The rules thus deduced for the two pumps using the programming syntax are encoded as:

<peNB:c1PB>, <peNB:c2PM> (6.2.1)

<peNM:c1PM>, <peNM:c2PM> (6.2.2)

As the fuzzy sets are overlapping, the change in control action varies smoothly as the correct isocline is approached, resembling the behaviour of a proportional controller. Eight rules are required to meet the first objective. These can be found in appendix A.

Having ensured that the first objective is satisfied, a deeper analysis of suitable means for finding the optimal point will be performed. Illustrated in fig 6.5 we find three plausible error surfaces with very different characteristics.

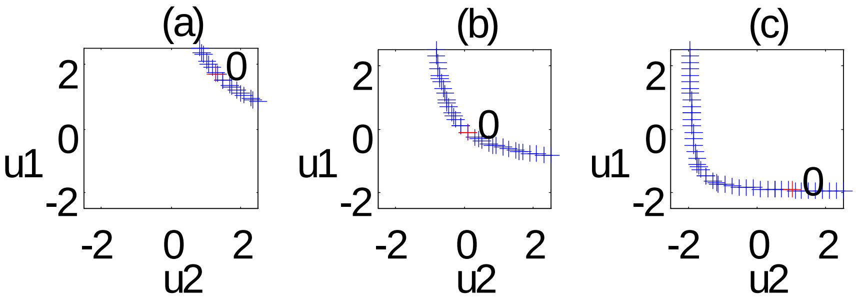

Figure 6.5 View from above of three different error surfaces with the isoclines illustrating the contours of desired plant output. The optimal points (section 6.1) are set to (u1,u2): a):(1.5,1.5), b):(0,0), c):(-1.5,-1.5)

As the pressure error as a consequence of the fulfilment of the first objective now is around zero, the operating point of the plant must be somewhere along the isocline. In order to determine which part of it, we look back into figure 6.4 for the answer. For figures 6.5 a) and 6.5 b) we can easily find out if the operating point is along the vertical or the horizontal part of the contour. Assuming that the operating point is in the middle of the horizontal isocline of figure 6.5 a), we find that we could describe the situation by saying that the pressure error is zero and the point is located in region 3 of figure 6.4. Translated into the language of fuzzy logic, the antecedents describing this point would read:

<PRESSURE ERROR is ZERO and SQUARE is POSITIVE BIG and u_2 is POSITIVE BIG>

In which the second antecedent, i.e. SQUARE is POSITIVE BIG, refers to either region 1 or 3 and the third antecedent, u_2 is POSITIVE BIG, settles the correct region, i.e. 3.

Using the nomenclature adopted for the programs, the antecedents would be written as:

<peZ_,SqPB:u2NB> (6.2.3)

In the same manner we could describe points on the vertical part of the isoclines in figures 6.5 a) and 6.5b), now with reference to region 2 of fig. 6.4, as:

<peZ_,SqNB:u1PB> (6.2.4)

If the operating point already was at the target, i.e. having zero pressure and the same power consumption of the two pumps, the point would be described as:

<peZ_,SqZ_> (6.2.5)

As we have managed to quantify the position of the operating point, we are in a position enabling us to write down the appropriate control action. Yet again assuming we are in the middle of the horizontal isocline of figure 6.5 a), a suitable measure for moving closer to the optimal point would be to generate a negative change of the voltage in pump number two.

This rule would be written, with a semicolon proceeding the consequent, as:

<peZ_,SqPB,u2PB:c2NB> (6.2.6)

As we have to generate control voltages to both pumps in every situation, a rule also has to be made for pump number one. By contemplating the same figures 6.5a) and b) we realise that in this particular situation any change of the voltage of pump number one would lead to a non zero pressure error. Consequently, we construct a rule not altering the voltage as:

<peZ_,SqPB,u2PB:c1Z_> (6.2.7)

The rules 6.2.6 and 6.2.7 form a complete control action for the particular situation of the operating point being somewhere along the horizontal isoclines of figs. 6.5 a) and b). Following the same methodology as in these examples rules for moving along all of the isoclines of figure 6.5 were constructed.

The essential steps to take in order to reach the second objective can be summarised as:

-Identify in which of the four regions of fig 6.4 the operating point is.

-Alter the voltage of the correct pump.

By bearing in mind the inherent fuzziness of descriptions such as ‘zero’ or ‘negative big’ the large amount of situations accounted for by such few rules can be explained.

Control strategy no.2

Outlining the two objectives for this controller is made more swiftly as the first is exactly the same as the first objective stated for the previous controller in section 6.1. Again, having placed the operating point on the desired pressure level, i.e. the correct isocline, we are able to concentrate on the second, distinguishing objective.

Instead of as in the controller above dividing the base of the control surface into four regions, a more straightforward partitioning into two regions were performed. Looking back to figure 6.4 we observe that a sufficient description of the optimal point is ‘being on the right isocline and on the diagonal leaning over to the right’. Because this diagonal is indicating the line of the two control voltages being equal, a new fuzzy variable, ‘difference’ (De), which is equal to u_2-u_1, could be identified. A plot of the base of the control surface with isobars indicating the values of the variable De is presented in figure 6.6.

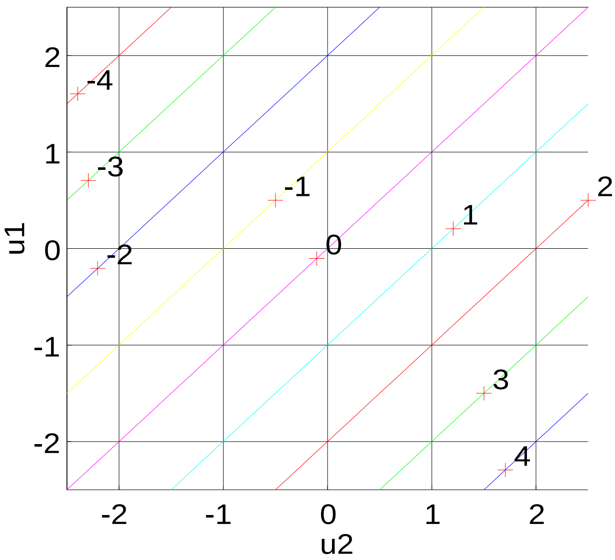

Figure 6.6 View from above of the control surface depicting the variable De = u2-u1.

The universe of discourse of the fuzzy variable De was partitioned into three regions, i.e. DeNB, DeZ_ and DePB, and the resulting fuzzy sets are presented in appendix A. Once more looking back to the three situations illustrated in figure 6.5, we discover means of generating control rules. As the control point as a consequent of the first objective now is on the isocline of correct pressure, we require a progress towards the optimal point located at the bend of the contour. Presuming the information ‘pressure error is zero and difference is negative big’, i.e. <peZ_,DeNB>, is given, a glance at figure 6.5, bearing in mind the partitioning illustrated in figure 6.6, immediately reveals that the operating point must lie on one of the vertical sides of the isoclines of figure 6.5 a)-c). In these situations we require a negative change of the voltage of pump one together with no change at all of the voltage of the second pump. In correspondence with section 6.1, these rules are presented below:

<peZ_,DeNB:c1NB> (6.2.1)

<peZ_,DePB:c2Z_> (6.2.2)

The two other possibilities that could occur along the contour of correct pressure, i.e. either a positive difference (horizontal line) or the optimal point are accounted for in two additional pairs of rules. Because of this straightforward representation a total of only six rules were required to meet the second objective. The total number of rules for the second control strategy thus amounts to twelve in comparison with the eighteen required for the previous controller. A list of the rules is included in appendix A.

6.3 Evaluation of the two fuzzy control strategies

In order to simulate the two control strategies, a programming environment capable of handling fuzzy sets had to be constructed. This was achieved in MATLAB version 4.0 which is a language most suitable for numerical simulations. Descriptions of the programs developed for the simulations are to be found in appendix H. All simulations were run on the standard PC 386 / 25 MHz computers present at ISVR.

The ability of the controllers to handle control surfaces having very different characteristics is outlined in subsections considering several measures of the performance.

6.3.1 Convergence towards the optimal point

Simulations were run for the control surface of figure 6.1, which has the simplest shape of all because of the centred position of the target point. Furthermore, the slope of the surface is neither extremely steep nor shallow, a fact that ensures smooth operation of the controllers. Plots of the behaviour of the two controllers are presented in figures 6.7: *Views from above illustrating the behaviour of the two controllers for the control surface of fig.6.1. Starting points are marked with X.*

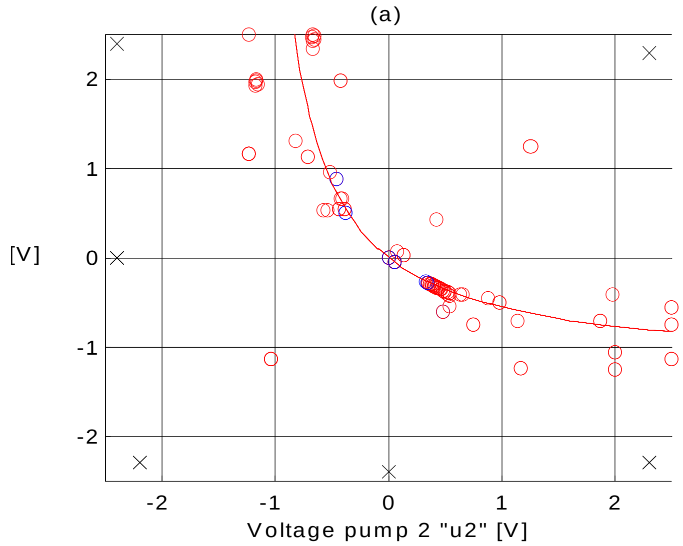

Figure 6.7 (a): Controller based upon strategy no.1 (u22-u12)

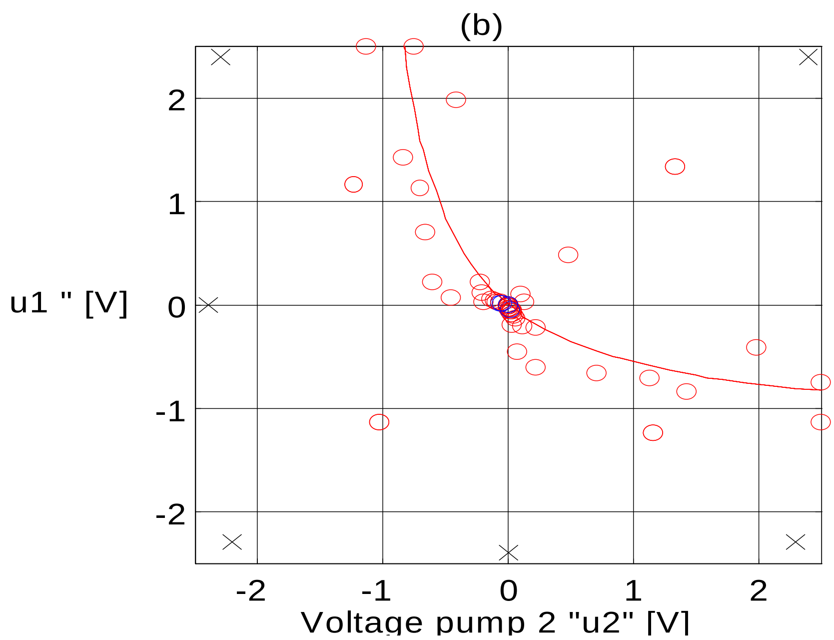

Figure 6.7 (b) Controller based upon strategy no.2 (u2-u1).

Both of the controllers manage to get close to the optimal point within a few iterations. The controller based upon the difference (fig. 6.7 b) is superior to the other which tends to get stuck at points above and to the right respectively below and to the left of the optimal point. This behaviour is explained by the inherent ambiguity the first control strategy. Referring back to section 6.2, the second objective of the first controller is to reach for the point where $u_2^2 = u_1^2$ under the condition of a zero pressure error. By examining figure 6.4 we find that not only the optimal point at the origin, but also the points at the intersections of the pressure contour and the diagonal from the top left to the bottom right corner satisfy that criterion. No such ambiguity exists for the simpler control strategy based upon the parameter u_2-u_1.

6.3.2 Number of iterations required to obtain the desired pressure

Yet again considering the control surface of figure 6.4, the time histories of the outputs of the two controllers were investigated. As the primary objective is to reach the correct pressure level, this measure is a check on the speed at which the controller adapts to new situations. Plots of three significant situations for each of the controllers are presented in figures 6.8: *Time history of the output of the two controllers. Starting points positioned at:*

(a)

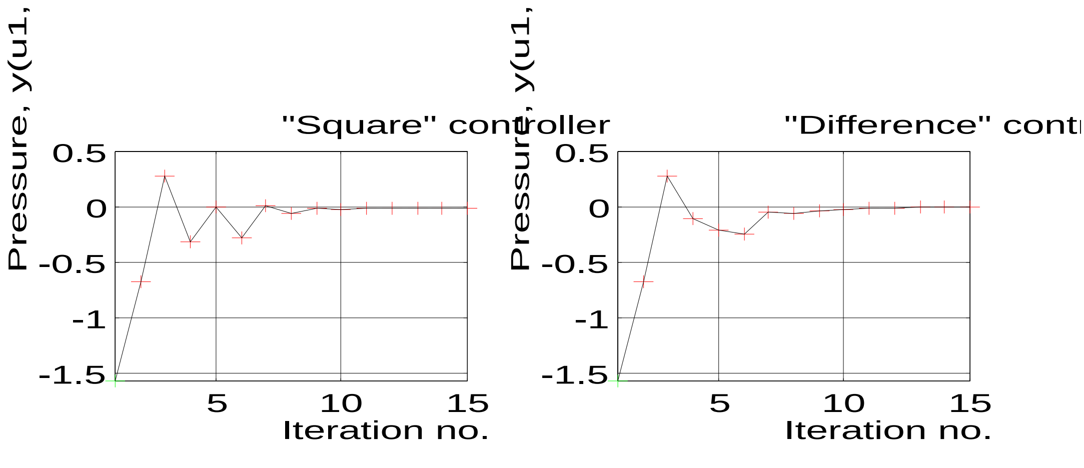

Figure 6.8 (a) (low pressure) around top left or bottom right corner)

(b)

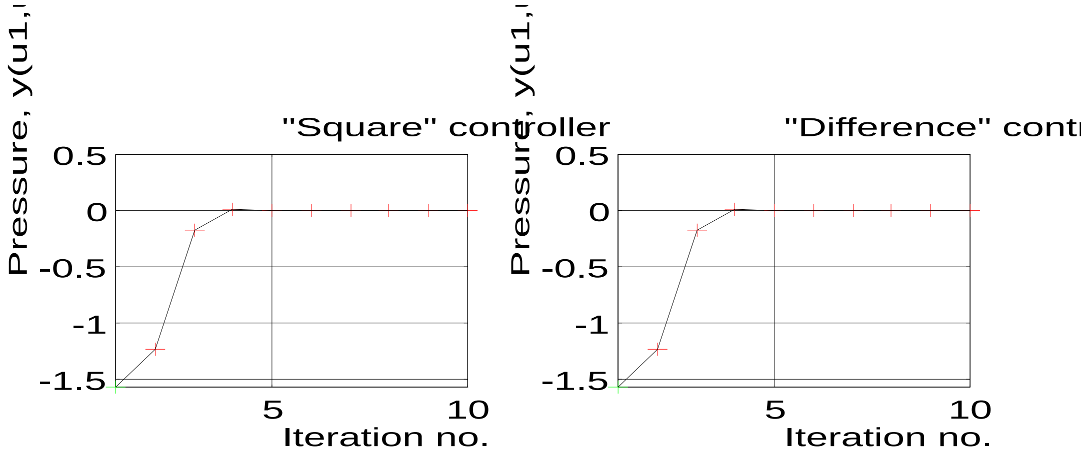

Figure 6.8 (b) (low pressure) around bottom left corner

(c)

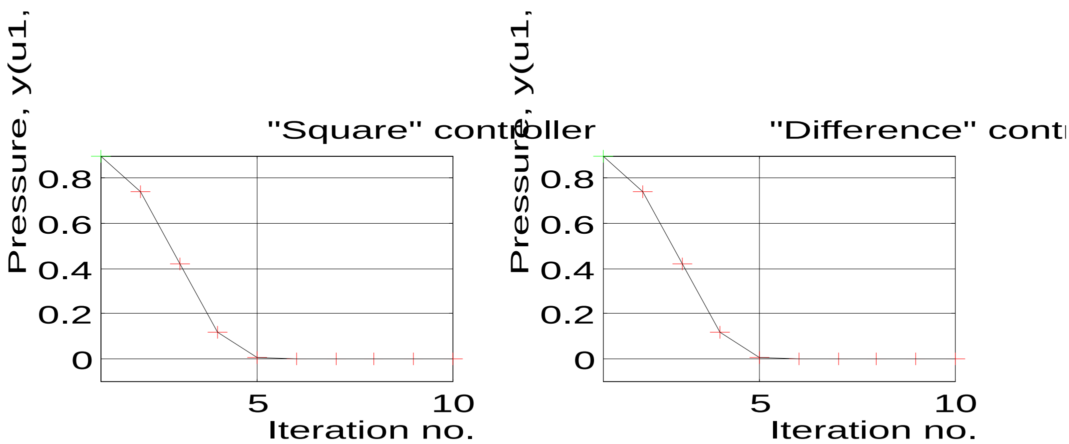

Figure 6.8 (c) (high pressure) around top right corner.

The controllers reach and stabilise around the desired pressure remarkably quickly thus illustrating the typical behaviour of adapting the step size to the distance from the target in order to get a minimum of overshoot.

6.3.3 Number of iterations required to reach the optimal point

Counting the number of steps in figure 6.7 we find that for this simple control surface the optimal solution is tracked within 10-20 iterations. A more challenging control surface would be the one illustrated in figure 6.9, having a flat top and very steep sides. Positioning the target point just above the knee of the surface leads to an excellent example of a problem at the verge of the capacity of the controller.

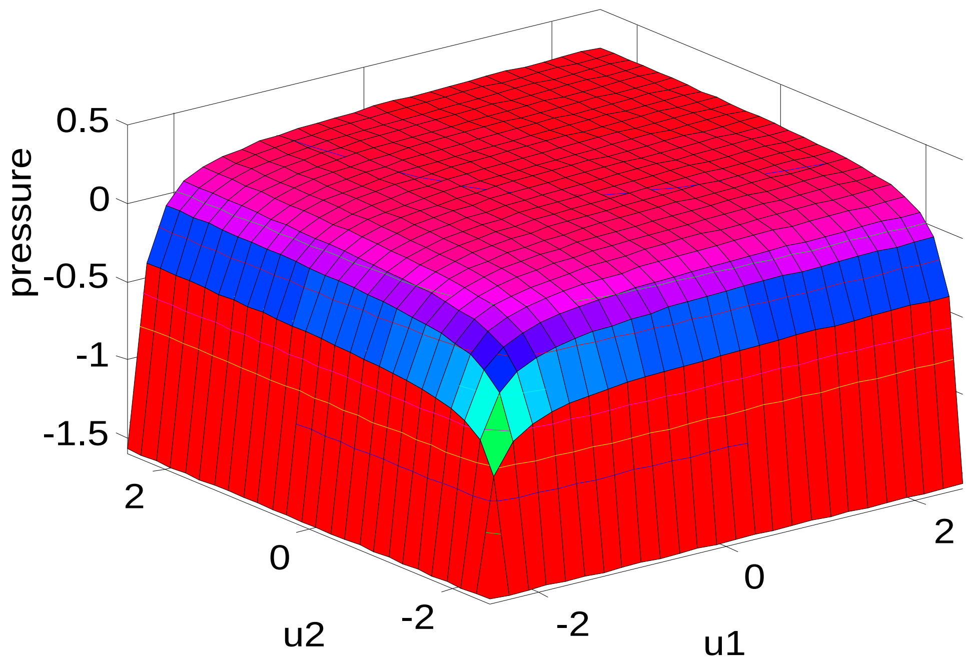

Figure 6.9 Control surface having the target point set to (-1.6,-1.6).

Starting at a point along the bottom left to top right diagonal on the top of the surface, the very small positive pressure error combined with the zero values of both of the parameters ‘Square’ and ‘Difference’ would mislead the controllers to believe that the target point is almost reached. In consequence the convergence towards the optimal point becomes slower starting at a point along this diagonal. The behaviour of the two controllers is shown in figures 6.10:

Views from above of the tracking of the optimal point on fig. 6.9:

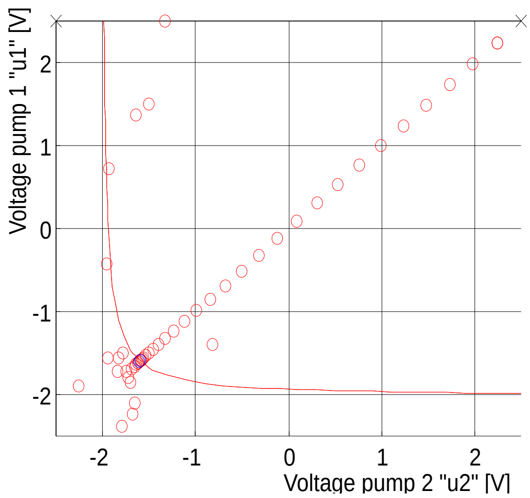

Figure 6.10 (a) 'Square' controller

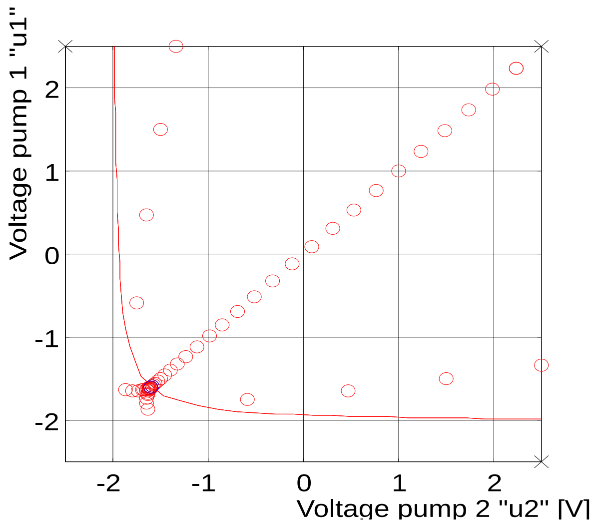

Figure 6.10 (b) 'Difference' controller.

The discernible characteristics in this situation are that whereas the 'Difference' controller for all of the starting positions is smoothly reaching for the target point, the other controller when moving in along the edges of the surface seems to jump erratically along the bend of the plateau. However, both of the controllers manage to track the target down. Again, traces of the ambiguity discussed in section 6.3.1 can be recognised from the two closely spaced points around position (1.5,-1.5) in figure 6.10 a).

6.3.4 Oscillations

Once more contemplating figure 6.9, we would expect problems as the voltage changes close to the optimal point could cause the operating point to fall abruptly from a small positive to a large negative pressure error. This could lead to a large compensation lifting the operating point back again thus returning it to the same point and causing an endless cycle i.e. oscillations. In order to avoid this phenomena the interaction between the fuzzy sets must be capable of slowing down the dynamics of the control process as the target is approaching. An example of this oscillatory behaviour showing the differences of the algorithms is given in figure 6.10.

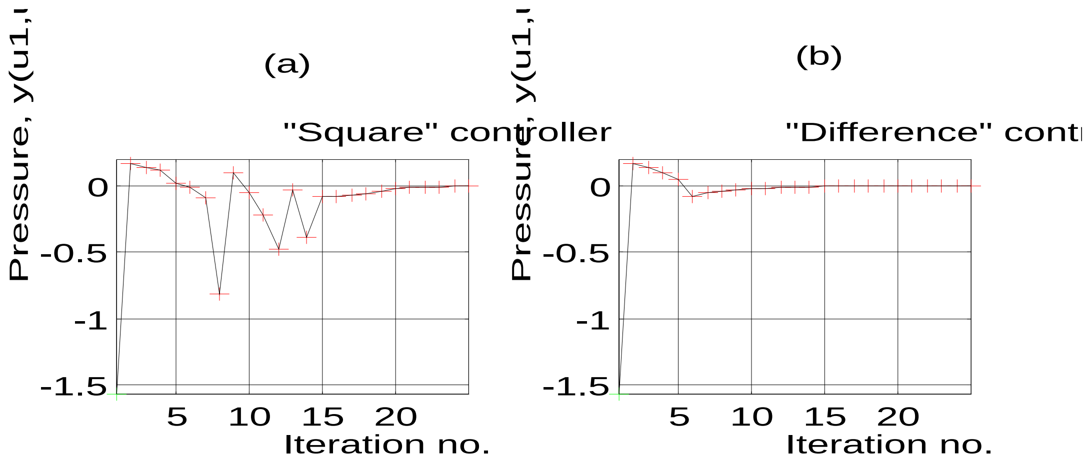

Figure 6.11 Oscillations in the two fuzzy control systems. Plot of the pressure level (e) time history when starting from the top left corner. (The tracking is shown in fig.6.10) : a) 'Square' controller, b) 'Difference' controller.

The disadvantage of having to smooth a system is that it often slows down the action of the controller. Hence a trade-off between oscillatory behaviour and speed often has to be made which in this particular system is reflected by the fact that a less oscillatory controller around the knee of the control surface leads to an increased number of smaller steps required to reach the target point. The fine tuning of fuzzy sets in order to overcome such problems can be found in [*5*].

Concluding remarks

In this chapter, a model of the physical system has been used for deducing rules for two fuzzy controllers. Performance criteria such as convergence and number of iterations requred to reach the optimal point were used in order to select one of the two fuzzy control strategies. As demonstrated in the comparisons the ‘Difference’ controller turned out to be superior and was chosen for the experiments in the following chapter.

7. Experimental implementation of the fuzzy controller

In this chapter the performance of the fuzzy controller is contrasted to the behaviour of the system constructed in Rioual's [*8*] project. The algorithms were run on a 486 DX 50 MHz PC and the experimental equipment which was constructed by Rioual is presented in appendix E. The fuzzy controller in spite of its simplicity turned out to be very successful and capable of achieving its predefined objectives impeccably.

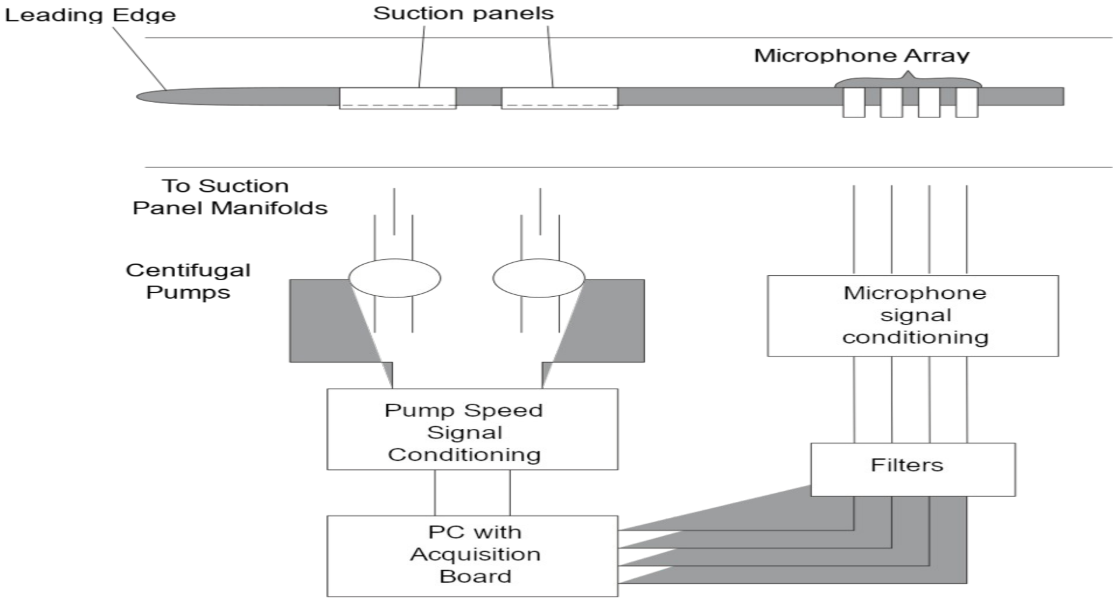

Figure E3 Sketch of the test rig and double channel equipment.

7.1 Previous work

In order to relate this thesis to the Ph.D. project [*8*] mentioned in the introduction, a brief description of that work is presented.

A part of that project concerned the implementation of an adaptive control system capable of controlling the position of the transition layer over an aerofoil whilst keeping the control effort a minimum. The controller uses a gradient descent method on a continuously updated linear model of the plant. The aim of the controller was to keep the transition layer on the wing profile in between the first and the fourth microphone, which are illustrated in appendix E, and at the same time having a minimal sum of the squares of the two suction pump voltages.

The problem was stated as:

Minimum pump effort $\quad \text{Min}\|u\|^2 \quad$ (7.1.1)

Error signal: (Constraint): $\quad e(\bar{u}) = r - y(\bar{u}) = 0 \quad$ (7.1.2)

Where u denotes an array of control voltages whose square is representative of the power consumption. The parameter r of eq. 7.1.2 represents the sum of the components of an array of desired pressures (eq. 7.1.3) which were deducted using a calibration routine [*8*].The indices of the x:s refer to locations of the pressure microphones see appendix E:

$$\overline{p_r} = [p_r(x_1), p_r(x_2), p_r(x_3) \, p_r(x_4)] \quad (7.1.3)$$

The plant was described as a function f(u) with additional white noise w(t):

$$y(\bar{u}) = f(\bar{u}) + w \quad (7.1.4)$$

The parameter y which is related linearly to the location of the transition layer was acquired using pressure microphones along the plate as illustrated in appendix E.

A traditional method of Lagrange multipliers, i.e. constrained optimisation, was developed leading to an iterative update of the pump voltages and the multipliers:

$$\bar{u}_{k+1} = \overline{u_k} - \mu\left(\nabla g(\overline{u_k}) + \nabla e(\overline{u_k})\right) \quad (7.1.5)$$

$$\lambda_k = \frac{e(\overline{u_k}) - \mu \nabla e(\overline{u_k})^T \nabla g(\overline{u_k})}{\mu\|\nabla e(\overline{u_k})\|^2} \quad (7.1.6)$$

However as the function f in the error function e by no means was known, a linear approximation of the function had to be made:

$$f(\overline{u_k}) = \theta_k^T \overline{u_k} + b \quad (7.1.7)$$

Furthermore, a general expression for the function to be minimised, g, containing the penalising multiplier R was constructed as:

$$g(\bar{u}_k) = \bar{u}_k^T R \overline{u_k} \quad (7.1.8)$$

As such a linear model as eq.7.1.7 is valid only locally, an on-line identification process had to be employed providing continuous updates of the parameters θ. Such an identification was achieved by using a recursive least squares (RLS) algorithm that provided new parameters at every iteration. In order to overcome some of the numerical instabilities during start-up the system identification routine was allowed to perform 50 iterations before the controller started tracking the optimal solution.

Experiments showed the following values of the error signal in eq.7.1.2:

Boundary layer turbulent already at x_1: $e = -2$

Transition in between x_1 and x_4: $e = 0$

Boundary level laminar at x_4: $e = 2$

The aim of the controller was to keep the location in between the microphones which corresponds to having an error signal close to zero.

The measured microphone pressures which consisted the outputs of the plant were RMS-weigthed for several choices of acquisition times. The shorter times leading to larger standard deviations equivalent to a significant noise level justify the noise term in eq.7.1.4. The earlier experiments also revealed that a special integral controller was needed to account for large fluctuations of the system as the linear updating no longer could cope with the identification of the plant. Thus the integral controller brought the plant back into the operating range of the RLS algorithm.

7.2 Evaluation of the performance of the fuzzy controller.

The control rules of the 'Difference' controller of chapter 6 were without modifications applied to the experimental equipment with remarkable success. Several initial conditions and pressure averaging times were tested.

The fulfilment of the objective of keeping the point of transition in between the first and the last (fourth) microphone was monitored in real time using an oscilloscope. This displayed the instantaneous pressure levels of microphones one and four enabling viewing of the progress of the controller as a decrease of the pressure level of the first microphone corresponds to a less turbulent flow. Thus, a transition point in between the first and the last microphones would result in a small signal from the first and a large signal, corresponding to turbulence, from the fourth pressure microphone.

In order to establish a suitable amount of smoothing of the input signal to the controller, different averaging times for the microphone pressure levels were investigated. An averaging time too short introduces too much of random noise because of the variations in the pressure level due to fluctuations in the flow and time needed for the system to stabilise. The shorter time tested lead to approximately the same performance in reaching the optimal point but unacceptable large oscillations around the desired pressure see fig 7.1: *Graphs showing the performance of the fuzzy controller. Initial pump voltages were (2.5,-2.5) using a pressure signal,* ***e averaging time of 0.1 s.***

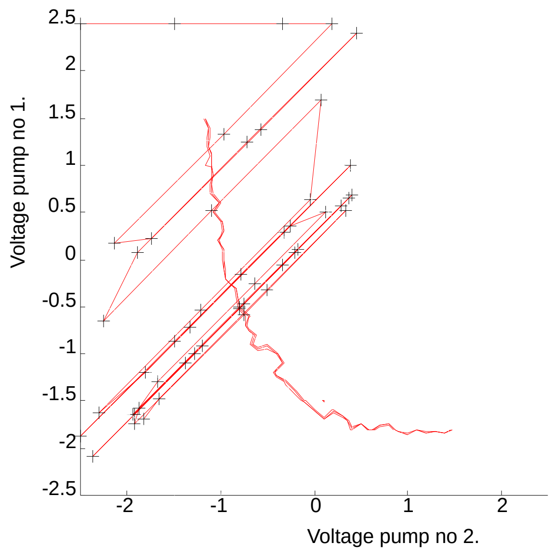

Figure 7.1 (a) averaging time of 0.1 s. The crescent shaped contour indicates the contour of desired pressure.

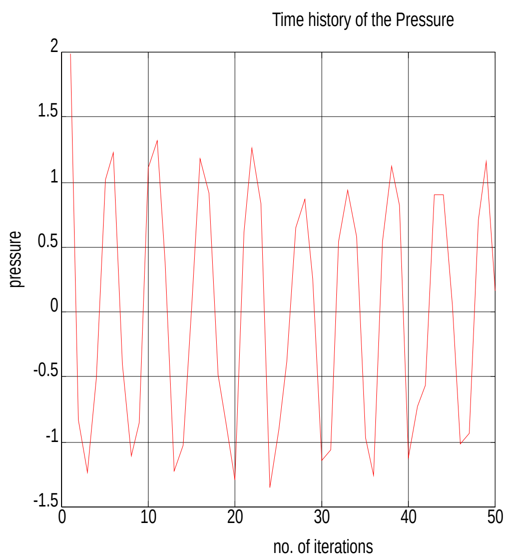

Figure 7.1 (b) averaging time of 0.1 s. Time pressure history

*Figures 7.2 Graphs showing the performance of the fuzzy controller. Initial pump voltages were (2.5,-2.5) using a pressure signal, **e averaging time of 1 s.***

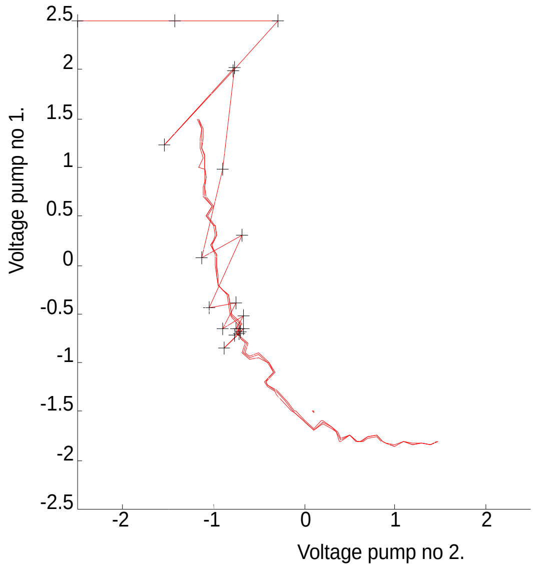

Figure 7.2 (a) averaging time of 1 s. The crescent shaped contour indicates the contour of desired pressure.

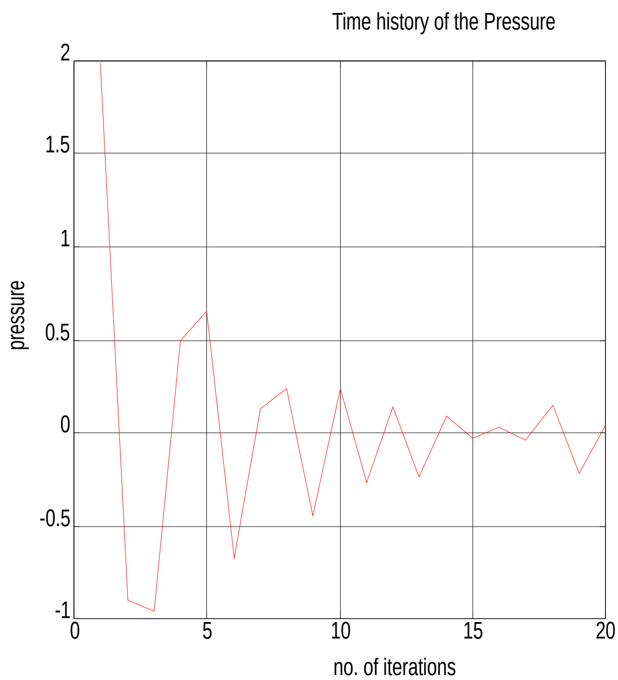

Figure 7.2 (b) averaging time of 1 s. Time pressure history

*Figures 7.3 Graphs showing the performance of the fuzzy controller. Initial pump voltages were (2.5,-2.5) using a pressure signal, **e averaging time of 2 s.***

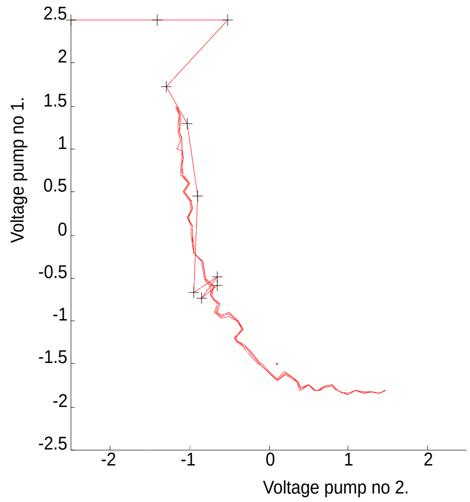

Figure 7.3 (a) averaging time of 2 s. The crescent shaped contour indicates the contour of desired pressure.

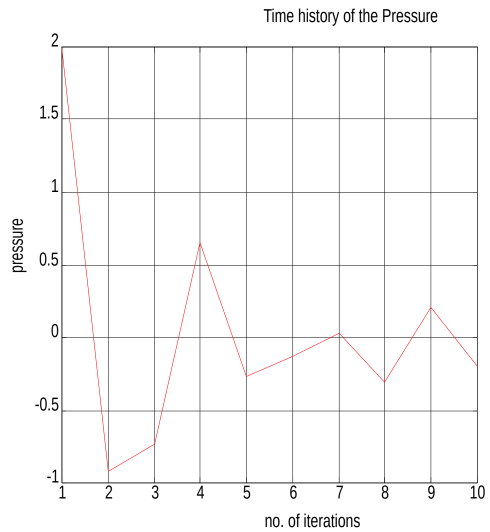

Figure 7.3 (b) averaging time of 2 s. Time pressure history

Comparison of the different averaging times in terms of oscillations around the target pressure level versus the speed of the controller yielded as a compromise a suitable averaging time of one second. The performance in terms of oscillations should be compared to that of the simulations for the same starting position presented in figure 6.8 c) of chapter 6.3.2. Having established a suitable averaging time tests were carried out from several starting positions in order to detect a possible instability of the system. However, in all of the situations tested the controller proved successful. An example of the behaviour of the controller for a starting position close to the target, initial pump voltages (0.5, 0.0), is given in figures 7.4.

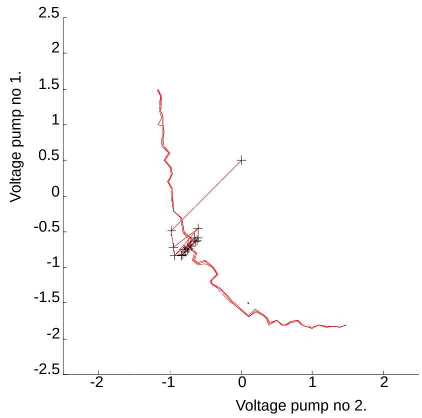

Figure 7.4 (a) Avg time 1 s, starting close to the target point. Contour.

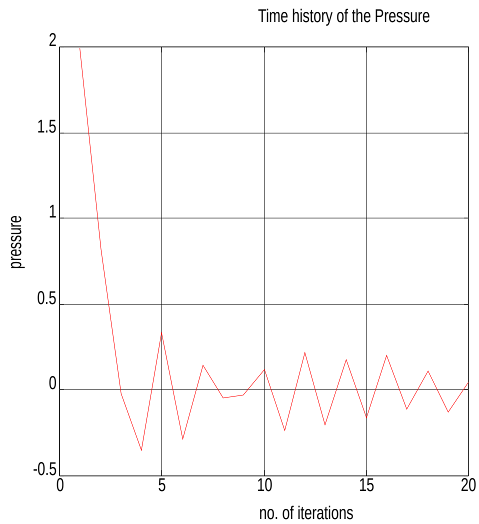

Figure 7.4 (b) Avg time 1 s, starting close to the target point. Pressure.

All of the results in graphs 7.1-4 were taken using an intermediate wind tunnel speed corresponding to a voltage of 165 volts. However, as in reality the speed of the air flow varies greatly the flexibility of the controller is also of great importance. One way of testing this is to investigate how the controller works during a significantly higher flow velocity corresponding to finding an optimal point situated closer to the top right corner of the control surface. This was carried out by turning the control knob of the wind tunnel fan up to its maximum voltage of 240 volts, according to its specifications equivalent to a mean flow speed of 23 m/s. Yet again, the fuzzy controller handled also this situation without problems. In this situation the limitations of the pumps were noticeable as those struggled to provide adequate suction rates.

In all of the situations presented, as the controller was homing in on the target point the suction rates were firstly adjusted to provide the correct point of transition and secondly moving closer to each another to finally move in phase around the final voltage.

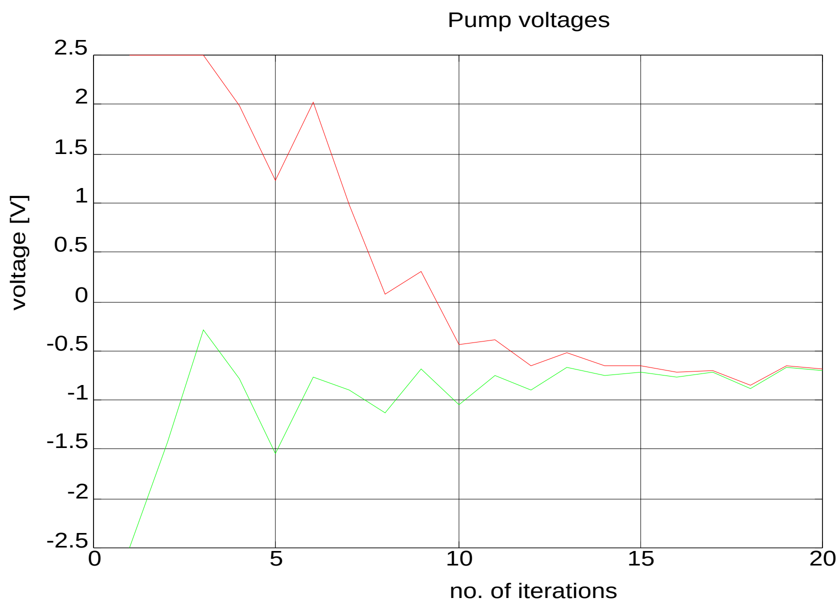

Figure 7.5 Plot of the two pump voltages resulting from a starting point at (2.5,-2.5).

The fuzzy controller was capable of finding the optimal point (i.e. the point at which the error signal e = 0 and the two pump voltages had the same value) within 5-10 iterations. Having established a suitable averaging time the controller proved capable of finding this point from all possible starting situations and having a residual oscillatory level of less than [-0.25 - + 0.25] units around the zero error signal level.

7.3 Comparison with the steepest descent method

Number of iterations to reach the target point

The fuzzy controller manages to reach the target point in less than ten iterations which is close to the value of 10-15 reported by Rioual [*8*]. Furthermore no initial identification iterations are needed for the fuzzy controller which immediately starts to operate on the predefined plant model.

Oscillations

As the previous section shows, the fuzzy controller with the averaging time chosen to one second oscillates around the desired zero error signal with an amplitude of 0.25 units. For comparison the steepest descent controller with an averaging time of 0.5 seconds has an oscillatory amplitude of approximately the same value. However, the oscillatory amplitude is dependent more upon the gain of the controllers rather than the method used. A high gain, which in the case of the fuzzy controller corresponds to having larger values on the universe of discourse of the sets governing the change in voltages, leads to faster tracking of the optimal point. The drawback of having too high a gain, i.e. an oscillatory system, becomes critical for very steep control surfaces as explained in section 6.3.4. A suggestion on how to overcome this problem is given in chapter 8.2.

Integral control

Rioual reports that the need for a separate integral controller could be overcome by carefully adjusting the parameters of the steepest descent method. As the fuzzy algorithm is designed using the complete error surface no such problems were encountered.

Concluding remarks

However successful the fuzzy controller turned out to be it should be pointed out that the gradient search method is slightly more sophisticated as it is capable of finding optimal points located off the diagonal indicating equal pump voltages. This capability is desirable when the control surface is not symmetric with respect to the diagonal a situation which may occur in many applications. However, for the purpose of this thesis which was to design a simple fuzzy controller this particular type of control surface was dealt with powerfully. Suggestions for development of the fuzzy controller are given in the following chapter.

8. Conclusions-Further work

8.1 Conclusions

The aim of providing a comparison between fuzzy and optimal control was carried out in a simplified way, illustrating the difference in cost for several simulation times. The results indicated similarities of the two methods for the shorter control times whereas the optimal controller was clearly more advantageous for the longer control times.

An investigation of the control surface obtained from experimental identification of the plant led to the construction of two fuzzy control strategies. Simulations of the practical system led to the selection of one of the strategies for the experiments.

The objective of monitoring a time averaged variable rather than the instantaneous value itself was met with great success. Throughout the project in simulations as well as in the experiment the fuzzy controller proved to be capable of handling the non-linearities of the control problem. The fuzzy controller did not need a separate integral control law to account for large steps upon the control surface. However, in order to ensure less oscillations suggestions are made to modify the control laws as the optimal point is approached.

Furthermore the simplicity of the formulation of the fuzzy control rules has not lead to any difficulties in finding the optimal point as defined. Comparison with the steepest descent method showed that the fuzzy controller in spite of its simplicity was performing very well in similar situations.

The rough model of the plant given as a starting point was enough to construct a controller capable of managing the physical system. This feature demonstrated the inherent capability of fuzzy logic of ability to capture information from vague and imprecise knowledge.

8.2 Suggestions for further work

Smoothing of the average value of pressure

An investigation of the possibility to extract more reliable values of the error signal without the drawbacks of long averaging times.

Modification of the fuzzy controller (gradient)

A case more directly related to the fuzzy control algorithm is how to modify this in order to find an optimal point off the symmetry diagonal. One plausible approach could be to incorporate another variable representing the difference between the error signal for two subsequent iterations. This method bears a certain resemblance to a gradient method as the controller could be made to move in the direction of the optimal point with step sizes deduced by fuzzy rules. However, problems may be encountered in terms of local minima and indeed the definition of the optimal point itself.

Self tuning fuzzy sets

As discussed in the previous chapter there is a trade-off between speed and residual oscillations. A powerful countermeasure would be to have an adaptivity in the fuzzy controller dependent upon the shape of the local error surface. For instance an approach in which the strength of the outputs decrease as the controller approaches the optimal point could prove very useful. This would allow a rapid control far away from the optimal point becoming gradually slower as the optimal point is reached which is crucial for very steep surfaces that could be expected in applications. One way of monitoring the distance to the optimal point would be to measure the variance of the error signal e. As this signal decreases the self-tuning controller would be less sensitive to sudden spikes and random disturbances due to the finite averaging time of the pressure signals.

The whole idea of self tuning fuzzy sets inevitably leads to the resent research in combining the adaptivity of neural networks with the manner in which information is stored in fuzzy sets. I firmly believe that the key to success in constructing controllers for more complex systems in the near future lies within the strength of neuro-fuzzy systems.

List of appendices

A: Representation of fuzzy sets and rules

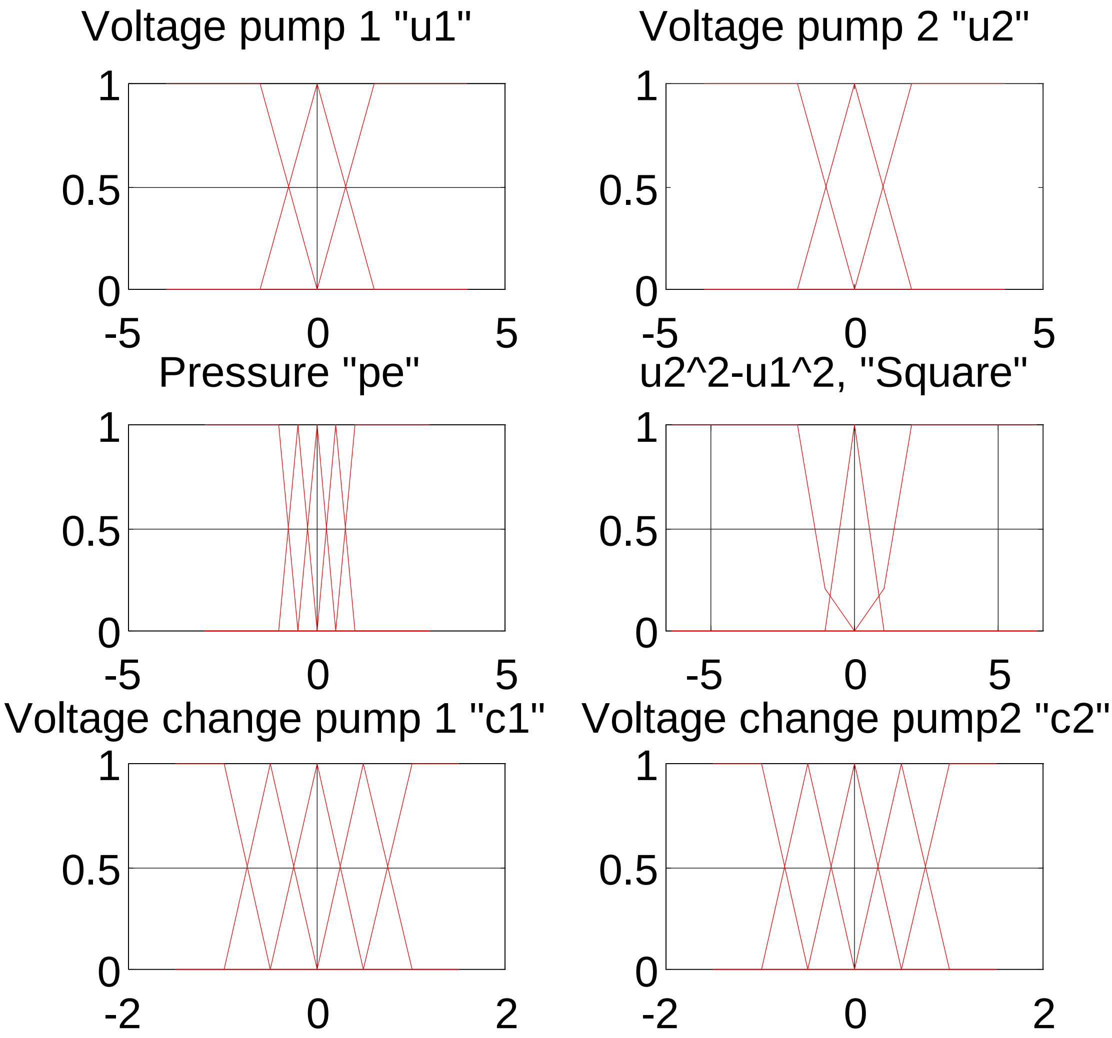

Figure A.1 Fuzzy sets for the "Square" controller

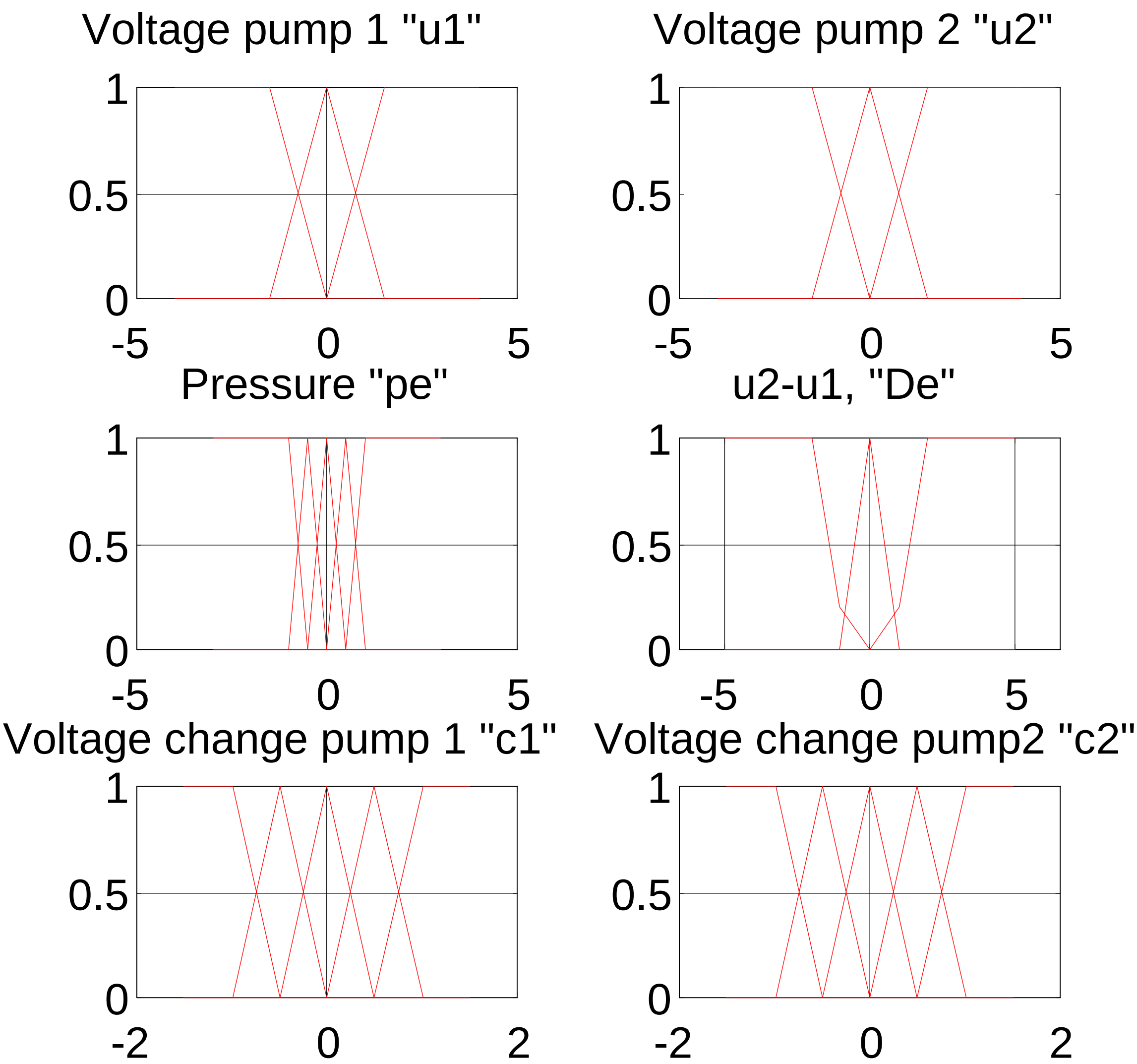

Figure A.2 Fuzzy sets for the "Difference" controller

%RULETABLE8.M

%A list of all the rules of the type:

%IF pe is PB and u1 is PB THEN change in u1 is NB

%Uses the parameter De (Difference)

*

peNB:c1PB

peNM:c1PM

pePM:c1NM

pePB:c1NB

peZ_,SqNB:c1NB

peZ_,SqPB:c1Z_

peZ_,SqZ_:c1Z_

peNB:c2PB

peNM:c2PM

pePM:c2NM

pePB:c2NB

peZ_,SqPB:c2NB

peZ_,SqNB:c2Z_

peZ_,SqZ_:c2Z_

B: Programs for the simulations of the fuzzy controllers

```
%SIFUCTLD.M
%Main program simulating the fuzzy controller

nos=input('Enter no of iterations');
u1new=input('Enter starting value for u1');
u2new=input('Enter starting value for u2');
% Choose the desired target point
% (+2.5 lifts the values up to the interval [0-5])
u1opt=-2+2.5;
u2opt=-2+2.5;
%Pressure level at target
a=0;
% Equation p5 blue ISVR report
alfa1=u1opt^3;alfa2=u2opt^3;
c=u1opt^2+u2opt^2;
b=0.2;
% Generate the first change of the control action
[u1di,u2di]=genchd(a-atan(b*(alfa1/(u1new+2.5)+alfa2/(u2new+2.5)-c)),u1new,u2new,lmx5);
figure(1);
clf;
figure(1);
w=0:0.1:pi/2;
R=[0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 6.5 7];
hold on;
for i=1:length(R),
        plot(R(i)*cos(w)-2.5,R(i)*sin(w)-2.5,'k:');
end;
kontur;
hold on;plot(u2new,u1new,'go');
axis([-2.5 2.5 -2.5 2.5]);
plot(u2new+u2di,u1new+u1di,'ro');
figure(2);
clf;
plot(1,a-atan(b*(alfa1/(u1new+2.5)+alfa2/(u2new+2.5)-c)),'g+');
hold on;
axis([1 nos -2 2]);
figure(1);
for i=2:nos,
        if sign(u1new+u1di)==(1|0)
                u1new=min(2.5,u1new+u1di);
        else
                u1new=max(-2.5,u1new+u1di);
        end;
        if sign(u2new+u2di)==(1|0)
                u2new=min(2.5,u2new+u2di);
        else
                u2new=max(-2.5,u2new+u2di);
        end;
        u1=u1new+2.5;
        u2=u2new+2.5;
        [u1di,u2di]=genchd(a-atan(b*(alfa1/(u1)+alfa2/(u2)-c)),u1new,u2new,lmx5);
        if i~=nos        % Plot the location on the control surface
                figure(1);
                plot(u2new,u1new,'ro');
```

```
        end

figure(2);          %Plot the pressure level
        plot(i,a-atan(b*(alfa1/(u1)+alfa2/(u2)-c)),'r+');
                ylabel('Pressure');
                xlabel('Iteration no.');
end;
figure(1);
plot(u2new,u1new,'bo');%Plot the final value as a black ring
grid on;
xlabel('Voltage pump 2 "u2" [V]');
ylabel('Voltage pump 1 "u1" [V]');
%Display final values
disp('final error');
disp(a-atan(b*(alfa1/(u1new+2.5)+alfa2/(u2new+2.5)-c)));
disp('final u1 & u2');
disp(u1new);
disp(u2new);
disp('final Sq');
disp(u2new^2-u1new^2);
```

%GENCHD.M

```
%Creates the change of control action (Function,loop)
%Reads the ruletable and calculates crisp output values
%for the changes in pump voltages

function [u1di,u2di]=genchd(pe,u1,u2,lmx)
global  SqNB SqPB SqZ_ peNB pePB peZ_ u1NB u1PB u1Z_ u2NB u2PB u2Z_ c1NB c1PB c1Z_
c2NB c2PB c2Z_ ;
global c1NM c1PM c2NM c2PM
global peNM pePM

fid=fopen('rultab8.m');

%Calculate matrix containing degree of membership in all of the sets in lmx
msmtr=genmmd(pe,u1,u2,lmx);
%read the rules in rultab
a=fscanf(fid,'%[^*]s');
a=fscanf(fid,'%[*]s');
fl=[];u1s=zeros(1,7);u2s=zeros(1,7);ms=[];
i=0;

while feof(fid)~=1,
i=0;
        while fl~='c',%Do until a consequent is next
                i=i+1;
                [a fl ]=rread(fid);
                %Construct array containing degree of fulfilment of
                %the antecedents%GENMMD.M
%Extracts the matrix cont. mship values in the sets
```

GENMM.M

```
function [msmtr]=genmm(pe,u1,u2,lmx);
global peNB peZ_ u1NB u1Z_ u2NB u2Z_ pePB u1PB u2PB SqNB SqNM SqZ_ SqPM SqPB c1NB
c1Z_ c2NB c2Z_c1PB c2PB u1PM
global c1NM c1PM c2NM c2PM
global peNM pePM
Sq=u2-u1;
```

```
for i=1:length(lmx),%For all the of the sets within lmx
        if ~strcmp(lmx(i,1),'c')
                [msmtr(i) crisp]=mship(eval(lmx(i,:)),eval(lmx(i,1:2)));
        end;  %if
end;


                ms(i)=msmtr(findi(a,lmx));
        end;
        %Deduce crisp output value
        if fl=='c',
                [a fl]=rread(fid);
                p=eval(a);
                %Truncate the output set at the level
                %of the minimum mship value of the antecedents
                q=[min(min(ms),p(1,:));p(2,:)];
                ms=[];
                %Separate pump 1 from pump 2
                if a(1:2)=='c1'
                        u1s=[max(q(1,:),u1s(1,:));p(2,:)];
                elseif a(1:2)=='c2'
                        u2s=[max(q(1,:),u2s(1,:));p(2,:)];
                end;
        end;    %while fl~='c'
ms=[];
end;    % while not EOF
u1di=defuzz(u1s);
u2di=defuzz(u2s);
fclose(fid);
```

% MSHIP.M

```
% Calculates a membership value from a fuzzy set
function [grade,crisp]= mship(set,crisp)
[smaller,larger]=localize(set,crisp);
grade=intpol(set,crisp,smaller,larger);
```

%FINDI.M

```
%Returns the row index of text matrix lmx corresponding
%to where the string str is stored


function i=findi(str,lmx);
i=1;
while (strcmp(lmx(i,:),str) ~=1)&(i<length(lmx)),
        i=i+1;
end
```

% LOCALIZE.M

```
% Finds out if a value is a member of a fuzzy set,
% Returns the surrouning values of the universe of discourse if true
% Otherwise it returns [-100 -100]

function [smaller,larger]=localize(set,crisp)
equal=find(set(2,:)==crisp);
if isempty(equal),

        tmps=find(set(2,:) < crisp);
        tmpl=find(set(2,:) > crisp);
```

```
		if (1~=(isempty(tmps) | isempty(tmpl))),
			smaller=tmps(length(tmps));
			larger=tmpl(1);
		else
			smaller=-100;
			larger=-100;
%else negative indices
		end
else
		smaller=equal;
		larger=equal;
end
```

INTPOL.M

```
function [grade]=intpol(set,crisp,smaller,larger)
if (smaller==-100), grade=0, else
		if (smaller~=larger),
		%disp(set(2,larger));disp(set(2,smaller));
		grade=set(1,smaller)+(set(1,larger)-set(1,smaller))/(set(2,larger)-set(2,smaller))*(crisp-
set(2,smaller));
		else grade=set(1,smaller);
	end,
end
```

%DEFUZZ.M

```
% Defuzzifies an output set by calulating a crisp
% value by the centre of area method
function [crispoutput]=defuzz(A);
if trapz(A(2,:),A(1,:))~=0 %Calculate centre of gravity
		crispoutput=trapz(A(2,:)',(A(1,:).*A(2,:))')/trapz(A(2,:),A(1,:));
else%Takes care of zero output sets
		crispoutput=0;
end;
```

%RREAD.M

```
%Reads the rules from the ruletable in
%(RULTAB.M)
%global fid;
function [str,fl]=rread(fid)
%global fid;
%fid=fopen('rultab.m');
%a=fscanf(fid,'%[^*]s');
%a=fscanf(fid,'%[*]s');
%for i=1:10,
		str=fscanf(fid,'%[^,\n\r:]s');
		%str=fgetl(fid);
		%disp(a);
		%disp(size(a));
		b=fscanf(fid,'%[,\n\r:]s');
		%b=fgetl(fid);
if b==':',
		%disp(b);
		fl='c';
end;
		%disp(size(b));
%end;
```

C: Programs for the comparison of the optimal and the rule based controllers

%PFIN.M

```
%Simulates and solves the variance problem
%with a fixed end condition p(tf)=Ptf
%Secant method employed

zn_1=input('initial guess for z(0)?');
[t,x]=ode45('opt1',t0,tf,[zn_1,6]);
fzn_1=x(length(t),2)-3;
zn=input('second guess for z(0)?');
[t,x]=ode45('opt1',t0,tf,[zn,6]);
fzn=x(length(t),2)-3;
global A0 ALFA Q PD COST

while abs(fzn)>1e-4,
        tmp=secz0(zn,zn_1,fzn,fzn_1)
        zn_1=zn;
        zn=tmp;
        [t,x]=ode45('opt1',t0,tf,[zn,6]);
        fzn_1=fzn;
        fzn=x(length(t),2)-3;
end
```

%OPT1.M

```
%Contains the state space representation necessary for the MATLAB fcn ODE.45
%xdot(1)=z1dot
%xdot(2)=p1dot
function xdot=opt1(t,x)
global A0 COST PD Q
xdot=[2*x(1)*A0-2/COST*x(2)*x(1)^2+x(2)*2-2*PD;-2*x(2).*A0+2/COST*x(1).*x(2).^2+Q];
%xdot=[x(1);x(2)];


figure(1);
subplot(2,1,1)
plot(t,x(:,2))
title(grphtit(A0,COST,Q,PD))
text((t0+tf)/5,max(x(:,2))/2,maxval(t,x))

subplot(2,1,2)
plot(t,-1/COST*x(:,2).*x(:,1))
title('control (ALFA)')
xlabel('time [s]')
ham;
```

%SECZO.M

```
%Secant function to find a new initial value for z
function newz0=secz0(zn,zn_1,fzn,fzn_1)
newz0=zn-(zn-zn_1)/(fzn-fzn_1)*fzn;
```

%GRPHTIT.M

```
function [diverse]=grphtit(A0,COST,Q,PD)
diverse=['  A0=' num2str(A0) '  COST=' num2str(COST) '  PD=' num2str(PD) '  Q=' num2str(Q)];
```

%FUA0COP.M

```
%Simulates and plots the function specified
%"Fuzzy, System parameter A0 constant"
[t,x]=ode45('fua0co',t0,tf,p0);
global A0 ALFA Q PD COST konstant
%subplot(2,1,1)
figure(3);
plot(t,x)
grid on;
title('variance (p)')
%hold on
%subplot(2,1,2)
%plot(t,-1/COST*x(:,2).*x(:,1))
%title('control (ALFA)')
%xlabel('time [s]')
%hold on
```

%FUA0CO.M

```
%Switches the control action ALFA +/- konstant
%Constant system parameter A0
%pdot(1)=p1dot
function pdot=fua0co(t,p)
global A0 COST PD Q konstant
if p(1)>PD
        pdot=-2*(A0+konstant)*p(1)+Q;
elseif p(1)<PD
        pdot=-2*(A0-konstant)*p(1)+Q;
else
        pdot=-2*A0*p(1)+Q;
end
```

%HAM.M

```
%Calculates and plots the Hamiltonian for the variance problem
hamil=[];

global COST A0 Q PD

for i=1:1:length(t),
        hamil(i)=(-(PD-(x(i,2)))^2+((x(i,2)*x(i,1)))^2/COST-2*A0*(x(i,2)*x(i,1))+(x(i,1))*Q);
end
figure(2);
clf;
plot(t,hamil);
```

%INTCOST.M

```
%Calculates the cost over the interval
%for the optimised variation problem

function expense=intcost(t,x)
global PD COST
expense=trapz(t,(PD-x(:,2)).^2+1/COST*(x(:,2).*x(:,1)).^2)
figure(3);
plot(t,(PD-x(:,2)).^2+1/COST*(x(:,2).*x(:,1)).^2);
grid on;
ylabel('integrand in cost fcn');
xlabel('t');
```

D: Deduction of the Lyapunov equation

Starting from the expression for a randomly driven state space system,

$$\dot{x} = \mathrm{A}x + \mathrm{B}w \tag{D.1}$$

x is n x 1

w is n x m

assuming that w is a noise vector with

$$\mathrm{E}[w] = 0 \tag{D.2}$$

$$\mathrm{E}\left[w(t_1)w^T(t_2)\right] = Q(t_1)\delta(t_1 - t_2) \tag{D.3}$$

a solution for x(t) is obtained as

$$x(t) = \Phi(t, t_0)x(t_0) + \int_{t_0}^{t} \Phi(t, \lambda)\mathrm{B}(\lambda)w(\lambda)d\lambda \tag{D.4}$$

where **Φ** is the fundamental matrix

The mean value of x

Assuming that

$$\mathrm{E}\left[x(t_0)\right] = 0 \tag{D.5}$$

then

$$\mathrm{E}[x(t)] = 0 \tag{D.6}$$

since

$$\mathrm{E}[w(t)] = 0 \tag{D.7}$$

Second order properties:

Define the covariance matrix

$$P(t_1, t_2) = E\left[x(t_1)\, x^T(t_2)\right] \tag{D.8}$$

The zero-lag covariance matrix P(t) is obtained when $t_2 = t_1 = t$.
Starting from the expression of the zero-lag covariance matrix:

$$P(t) = E\left[x(t)\, x^T(t)\right] \tag{D.9}$$

we can obtain an expression for its change with time:

$$\dot{P} = E\left[\dot{x}(t)\, x^T(t)\right] + E\left[x(t)\, \dot{x}^T(t)\right] \tag{D.10}$$

substituting from eq D.1 yields

$$\dot{\mathrm{P}} = \mathrm{E}\left[(\mathrm{A}x + \mathrm{B}w)x^T\right] + \mathrm{E}\left[x(\mathrm{A}x + \mathrm{B}w)^T\right] \tag{D.11}$$

Considering only the first term in eq. D.11 yields:

$$\dot{\mathrm{P}} = \mathrm{AE}\left[xx^T\right] + \mathrm{BE}\left[wx^T\right] \tag{D.12}$$

The first term in eq. D.12 is to be recognised from eq D.9 as

$$\mathrm{AP} \tag{D.13}$$

and the expectation value in the second term of eq. D.12,

$$E\left[wx^T\right]=E\left[w(t)x^T(t_0)\Phi^T(t,t_0)+w(t)\int_{t_0}^{t} w^T(\lambda)B^T(\lambda)\Phi^T(t,\lambda)d\lambda\right] \qquad \text{(D.14)}$$

The first term in eq. D.14 disappears due to uncorrelation and the second term

$$\mathrm{E}\left[w(t)\int_{t_0}^{t} w^T(\lambda)\mathrm{B}^T(\lambda)\Phi^T(t,\lambda)d\lambda\right]=\mathrm{E}\left[\int_{t_0}^{t} w(t)w^T(\lambda)\mathrm{B}^T(\lambda)\Phi^T(t,\lambda)d\lambda\right] \qquad \text{(D.15)}$$

Substituting from eq. D.3 yields:

$$\int_{t_0}^{t} Q(t)\delta(t-\lambda)\mathrm{B}^T(\lambda)\Phi^T(t,\lambda)d\lambda \qquad \text{(D.16)}$$

Performing the integration and considering that only half of the dirac pulse is enclosed:

$$\frac{1}{2}Q(t)\mathrm{E}\left\{\mathrm{B}^T(t)\Phi^T(t,t)\right\}=\frac{1}{2}Q(t)\mathrm{E}\left[\mathrm{B}^T(t)\mathrm{I}\right]=\frac{1}{2}Q(t)\mathrm{E}\left[\mathrm{B}^T(t)\right] \qquad \text{(D.17)}$$

Substituting this result back into the second term of eq. D.12 we find:

$$\mathrm{BE}\left[wx^T\right]=\frac{1}{2}BQE\left[B^T(t)\right] \qquad \text{(D.18)}$$

Assuming that B is time-invariant this is equivalent to;

$$\frac{1}{2}\mathrm{B}Q\mathrm{B}^T \qquad \text{(D.19)}$$

Thus, the first term of eq. D.11 can be written as:

$$\mathrm{AP}+\frac{1}{2}\mathrm{B}Q\mathrm{B}^T \qquad \text{(D.20)}$$

Similarly, considering the part of the expectation value of the second term in eq. D.11:

$$E\left[xw^T\right]=E\left[\Phi\left(t,t_0\right)x\left(t_0\right)w^T(t)+\int_{t_0}^{t}\Phi(t,\lambda)B(\lambda)w(\lambda)d\lambda w^T(t)\right] \tag{D.21}$$

The first term is zero for the same reason as eq. D.14, and the second term

$$\mathrm{E}\left[\int_{t_0}^{t}\Phi(t,\lambda)\mathrm{B}(\lambda)w(\lambda)d\lambda w^T(t)\right]=\int_{t_0}^{t}\mathrm{E}\left[\Phi(t,\lambda)\mathrm{B}(\lambda)\right]Q(\lambda)\delta(\lambda-t)d\lambda \tag{D.22}$$

Integrating and considering that only half of the dirac pulse is enclosed:

$$\frac{1}{2}\mathrm{E}\left[\Phi(t,t)\mathrm{B}(t)\right]Q(t)=\frac{1}{2}\mathrm{E}\left[\mathrm{IB}(t)\right]Q(t)=\frac{1}{2}\mathrm{E}\left[\mathrm{B}(t)\right]Q(t) \tag{D.23}$$

Which, after assuming time-invariant B and substituting back into the second term of eq. D.12 yields:

$$\mathrm{BE}\left[xw^T\right]=\frac{1}{2}\mathrm{B}Q\mathrm{B}^T \tag{D.24}$$

Substituting eq. D.20 and D.24 whilst recognising the first part of the second term in eq. D.11 as the transpose of eq. D.13 into eq. D.11 yields:

$$\dot{\mathrm{P}}=\mathrm{AP}+\frac{1}{2}\mathrm{B}Q\mathrm{B}^T+\mathrm{PA}^T+\frac{1}{2}\mathrm{B}Q\mathrm{B}^T \tag{D.25}$$

Adding gives the final expression for the time dependency of the variance:

$$\dot{P}=\mathrm{AP}+\mathrm{PA}^T+BQB^T \tag{D.26}$$

In situations where the covariance is constant the left side of eq. D.16 disappears and we obtain the Lyapunov equation:

$$\mathrm{AP}+\mathrm{PA}^T+BQB^T=0 \tag{D.27}$$

E: Description of the experimental equipment (From Rioual [8])

In this appendix pictures illustrating the wing profile and the experimental equipment as given by Rioual are presented. For technical data see Rioual.

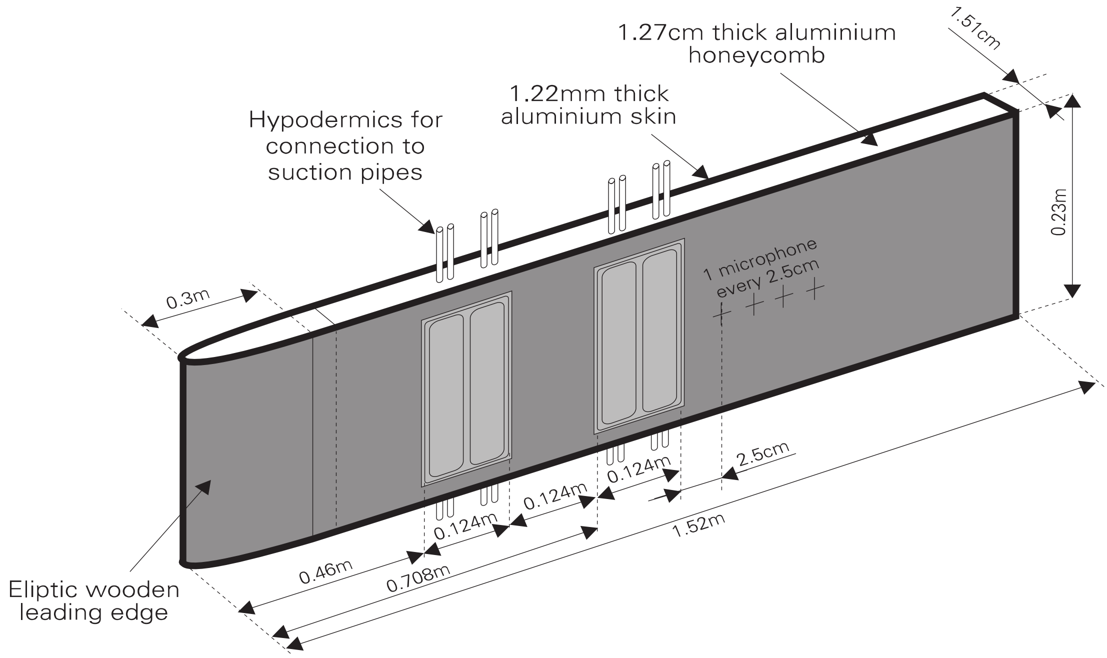

Figure E1 Location of suction panels and microphones on the plate.

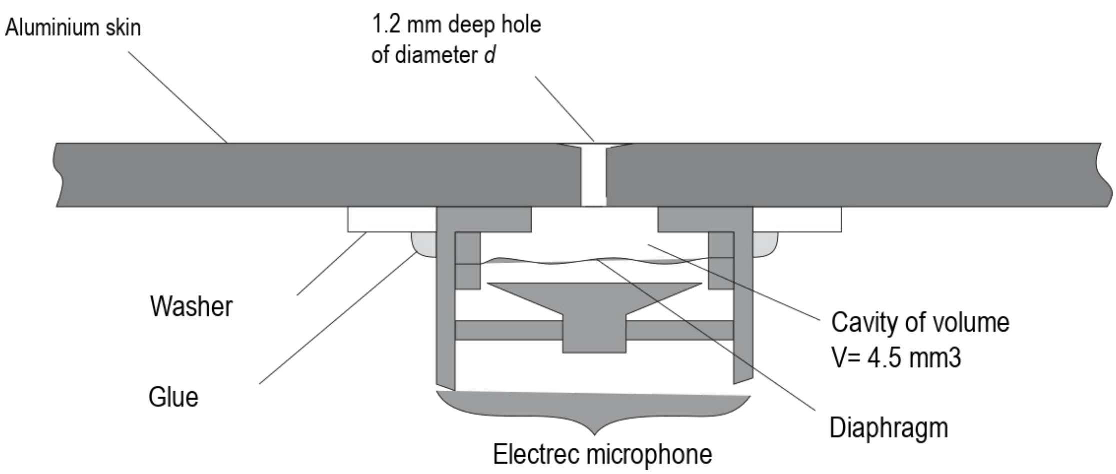

Figure E2 Mounting of a microphone and corresponding sizes of cavity constriction

Experimental equipment

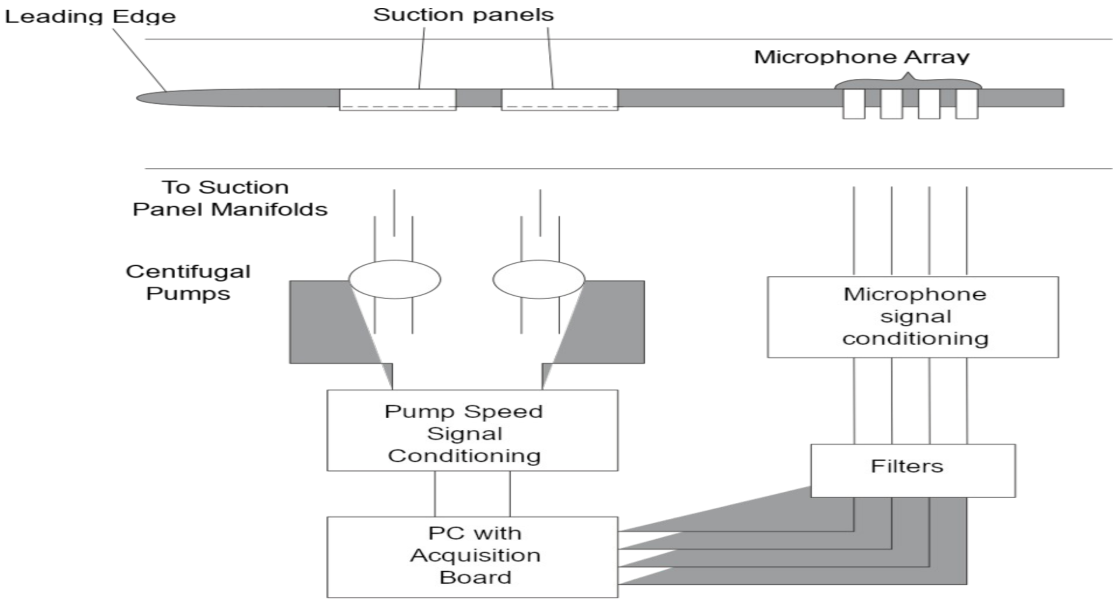

Figure E3 Sketch of the test rig and double channel equipment.

F: MATLAB program for the control of the experimental rig

FUCTRL.M

```
%*************************************************************************
% Controls the pump voltages by applying fuzzy control to a preidentified
% error surface . The program uses the fuzzy environment of appendix B
%*************************************************************************

%********** First initialization *********
clc;
time=1;
v1=input('Initial voltage pump 1 (0.0): ');
v2=input('Initial voltage pump 2 (0.0): ');
nin=input('Number of iterations: ');

start

save temp
clear
load temp

not=2;
load stdy3md.mat;
samprate=4000;
nop=samprate*time;
datum=ones(nop,4);
pres=ones(1,4);
y=zeros(1:nin);
e=0;
rand('uniform');
vs=[v1*ones(nin,1) v2*ones(nin,1)];
vsi = [v1 v2];
R=[1 1.5 2 2.5 3 3.5 4 4.5];
loR=length(R);

des=[.2 .33 .66 .8];
r=sum(des);

clc;
hold off;
clg;
%********** Start update of suction rate according to pressure distribution
contour(e175_3md,[-0.01 0 0.01],volt1,volt2,'r');
axis([-2.5 2.5 -2.5 2.5]);
axis('square');
title('Evolution of the two suction voltages on the error surface.');
xlabel('Voltage pump no 2.            ');
ylabel('Voltage pump no 1.');
hold on;
plot(volt1,volt2,'w');
for i=1:loR,
        P2=-2.4:0.025:(-2.5+R(i));
        P1=sqrt( R(i)^2 - (P2+2.5).^2 ) - 2.5;
        plot(P2,P1,'w');
end;
```

```
pause;
simultio(vtest(vs(1,:)));
pause(2);

k=0;
while k<(nin-1),
        k=k+1;
        % Make acquisition and calculate error signal:
        [datum,f]=snap(4,samprate,nop,4);
        pres=sqrt(mean(datum.^2))./cal;
        y(k+1)=sum(pres);
        e = r-y(k+1);

        % Calculate and output new voltages...
        vs(k+1,:)=fuzz(e,vs(k,:),lmx5);
        simultio(vtest(vs(k+1,:)));

        % Plot results:
        plot(vs(k+1,2),vs(k+1,1),'+w');

        if rem(k,200)==0,
                save temp;
                clear;
                load temp;
        end;
end;

xlabel('                              ... Program terminated ...');

save temp y vs r cal
clear
load temp

hold off;
```

FUZZ.M

```
%Main function calculating new
%voltages using fuzzy logic
%and the rules in ruletab8.m

function [vnew]=fuzz(e,vold,lmx5)
[v1di,v2di]=genchd(e,vold(1),vold(2),lmx5);

%Make sure voltages are within the range [-2.5 2.5]
if sign(vold(1)+v1di)==(1|0)
        vnew(1,1)=min(2.5,vold(1)+v1di);
else
        vnew(1,1)=max(-2.5,vold(1)+v1di);
end;
if sign(vold(2)+v2di)==(1|0)
        vnew(1,2)=min(2.5,vold(2)+v2di);
else
        vnew(1,2)=max(-2.5,vold(2)+v2di);
end;
```

G: List of references

JOURNALS & CONFERENCE PAPERS (C)

1 Sutton R.'Use of fuzzy logic in the implementation of control algorithms'
J.IERE Vol.55.p.358-67.Oct. 1985

2 Chiu S.'Control of roll & moment for a flexible wing aircraft'
IEEE Control Systems.p.42-8. June 1991

3 Dubois D.'Fuzzy sets: A survey of engineering applications'
Comp.& Chem. Eng'Vol.17.p373-80. 1993

4 Chiang W.'Dynamic response of structures with uncertain parameters'
Prob. eng. mech.Vol.2.p.82-91. 1987

5 Heger A.'Use of fuzzy logic in signal processing'
Am. nucl. Soc. 1993 annual.p.135-6.June 1993C

6 Yao J.'Uncertainties in structural identification & control'
F. Eng. tow. hum friendly Sys.p.844-9.Nov.1991C

7 Yeh E.'Fuzzy control for active suspension design'
Proc. of the intell. veh. 92'.p.92-7.June-July 1992C

8 Schwartz D.'Applications of fuzzy sets and approximate reasoning'
Proceedings of the IEEE Vol.82,no.4,p.482-98.1994

9 Rioual J.'Adaptive Boundary Layer Control with Distributed Suction'
ISVR, Univ. of Southampton. Contract rep.no.93/33 Sept. 1993

10 Rioual J.'The automatic control of boundary layer transition via a double suction panel'VPI Conf. 1993C

11 Hammond J.K.'Digital and Kalman filtering'
ISVR, Univ. of Southampton p.3:1-3:6

BOOKS (*bold, italics)*

1 White David A. and Sofge Donald A. (edited by) 'Handbook of intelligent control' 1992.NY:Van Nostrand Reinhold

2 Novák V.' Fuzzy sets and their applications' 1989. Bristol: IOP publishing Ltd.

3 Breuer J.' The theory of sets' .1958.USA: Academic press

4 Zadeh L. et al 'Fuzzy sets and their applications to cognitive & decision processes' 1975.New York:Academic press

5 Cox Earl 'The fuzzy systems handbook'.1994.London:AP

6 Hocking Leslie M.'Optimal Control. An introduction to the Theory with Applications.'1991.Oxford:Clarender Press.

7 Banks Stephen P.'Control systems engineering'[p.321-372] 1986.London:Prentice-Hall.

8 Rioual Jean-Luc 'Adaptive boundary layer control with distributed suction' 1994-5(To be presented).Ph.D.-Thesis.University of Southampton.

9 Pohl Peter et al 'Lärobok i numeriska metoder' 1983.Stockholm:THS AB

10 Press W. 'Numerical recipes, the art of scientific computing'. *1989. Cambridge:CUP*

H: The programming environment

A listing of the programmes used for the fuzzy and optimal controllers is attached in appendices B, C and F. A summary of the operating characteristics is given below.

H.1 Representation of the fuzzy algorithms (see chapter 4)

The programs described in this section is included in appendix A.

Fuzzy sets were stored in 2 by N arrays containing the degree of membership and the universe of discourse respectively. It is imperative that the universe of discourse for a group of sets operating on the same variable are identical. Otherwise problems in the defuzzification function will occur. The membership values of the in between the sets defined points on the universe of discourse were obtained by means of linear interpolation [INTPOL.M].

Decomposition of the combining AND operators was obtained by keeping track of the membership values of the antecedents and truncating the output fuzzy set at the smallest value of these .

The union of several output rule sets concerning the same variable was achieved by a Max. operation [GENCH.M].

Finally, a defuzzification [DEFUZZ.M] of the fuzzy output set was obtained by the means of the centroid method using the MATLAB functions for trapezoidal integration. The sets were stored as global variables for easy access throughout the workspace. A text matrix containing the names of the set served as a catalogue for the decomposition function .

The rules were stored in a text file [RULTAB.M] where the antecedents were separated from each other by commas which indicated an AND operation. The consequent was separated from the antecedent by a semi-colon. The first two letters of the sets indicates the corresponding variable whereas the last two determines the set concerned. For instance peNB means PRESSURE ERROR is NEGATIVE BIG

H.2 Solution of the optimisation problem

In order to find the optimal solution according to chapter 3, the differential equations had to be solved for initial conditions obtained by the iterative secant method. The differential equation solver ODE45 of MATLAB was employed for the numerical results. The complete program for obtaining the optimal solution, [PFIN.M] (appendix C) is included with subroutines in the program listings.

H.3 Simulations of the rule based controller

The program [FUA0COP.M] (appendix C) directly makes use of ODE45 for the rule based controller and presents the trajectory graphically.

H.4 The cost function and the Hamiltonian

As the routine ODE45 has solved an equation, an array containing the time history of the variables involved results. This was used for calculating such parameters as the cost function (sec. 3.2) and the Hamiltonian (sec 3.3). The two programmes are listed in appendix C as [INTCOST.M] and [HAM.M].

H.5 Programs for the control of the test rig.

The programs described in this section is included in appendix F.

The main program [FUCTRL.M] interfacing with the signal acquisition box is a modified version of the original steepest descent controller developed by Rioual [8]. In this program a function [FUZZ.M] applying fuzzy logic rules is called upon as many times as demanded by the operator during set-up of the run.