Optimal power control of a three-shaft Brayton cycle based power conversion unit

---

A dissertation presented to

The School of Electrical and Electronic Engineering

North-West University

In partial fulfilment of the requirements for the degree

Magister Ingeneriae

in Computer and Electronic Engineering

by

Kenneth R. Uren

Supervisor: Prof. G. van Schoor

Assistant supervisor: Mr. C.R. van Niekerk

May 2005

Potchefstroom Campus

SUMMARY

The aim of this study is to develop a control system that optimally controls the power output of a Brayton cycle based power plant. The original design of the PBMR power plant is considered. It uses helium gas as working fluid. The power output of the system can be manipulated by changing the helium inventory to the gas cycle.

A linear model of the power plant is derived and modelled in Simulink®. This linear model is used as an evaluation platform for different control strategies. Four actuators are identified that are responsible for manipulating the helium inventory. They are:

- A booster tank
- A gas cycle bypass control valve
- Low-pressure injection at the low-pressure side of the system
- High-pressure extraction at the high-pressure side of the system

The control system has to intelligently generate set point values for each of these actuators to eventually control the power output. Two control strategies namely PID control and Fuzzy control are investigated in this study.

An optimisation technique called Genetic Algorithms is used to adapt the gain constants of the Fuzzy control strategy. This resulted in an optimal power control system for the Brayton cycle based power plant.

OPSOMMING

Die doelwit van hierdie studie is om `n beheerstelsel te ontwikkel wat die drywingsuitset van 'n Brayton siklus gebaseerde kragstasie optimaal sal beheer. In hierdie geval word die oorspronklike PBMR ontwerp beskou wat helium gas gebruik as werksvloeistof. Die drywinsuitset van die stelsel kan verstel word deur gas tot die siklus by te voeg of te onttrek.

'n Liniêre model van die kragstasie word afgelei en gemodelleer in Simulink®. Hierdie liniêre model word gebruik as 'n toetsplatform vir verskillende beheer strategieë. Vier aktueerders is geïdentifiseer wat verantwoordelik is vir die manipulasie van die gastoevoer tot die siklus. Hulle is as volg:

- 'n Aanjaagtenk
- 'n Gasomloop beheerklep
- Lae druk gasinspuiting by die lae druk kant van die stelsel
- Hoë druk gasontrekking by die hoë druk kant van die stelsel

Die beheerder moet op 'n intelligente wyse die verwysingsvlakke van die aktueerders bepaal om die drywingsuitset te beheer. Twee beheerstrategieë naamlik PID en Wasige beheer word ondersoek.

Genetiese Algoritmes word gebruik om die beheerkonstantes van die Wasige beheer strategie te optimeer. Deur van hierdie tegniek gebruik te maak kon 'n optimale drywingsbeheerstelsel afgelei word vir die Brayton siklus gebaseerde kragstasie.

ACKNOWLEDGEMENTS

I would like to firstly thank M-Tech Industrial and THRIP for funding this research and granting me the opportunity to further my studies.

I would also like to acknowledge the following people, in no particular order, for their contributions during the course of this project.

- Professor George van Schoor, my supervisor, for his guidance, advice and support that stood central to the success of this project.

- Mr. Carl van Niekerk, my co-supervisor, for his guidance, advice and support.

- My fiancé, Marisa van der Walt, for her love, support and understanding.

- My parents, Basie and Bertha Uren, and Marisa's parents, Piet and Marina van der Walt, for their love and support.

"Blessed are all who fear the Lord, who walk in his ways. You will eat the fruit of your labour; blessings and prosperity will be yours." Psalm 128:1-2

TABLE OF CONTENTS

NOMENCLATURE

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

GA	Genetic Algorithm
GAs	Genetic Algorithms
GFS	Genetic Fuzzy Systems
PID	Proportional, Integral and Derivative
PBMR	Pebble Bed Modular Reactor
HPT	High-pressure Turbine
HPC	High-pressure Compressor
LPT	Low-pressure Turbine
LPC	Low-pressure Compressor
PT	Power Turbine
GBPC	Gas Cycle Bypass Control Valve
FPID	Fuzzy Proportional, Integral and Derivative
FDSS	Fuzzy Decision Support System
FSs	Fuzzy Systems
FRBS	Fuzzy Rule Based Systems
KB	Knowledge Base
DB	Data Base
RB	Rule Base
MPS	Main Power System
PCU	Power Conversion Unit
RU	Reactor Unit
GUI	Graphical User Interface
AI	Artificial Intelligence
LPI	Low-pressure Injection
HPE	High-pressure Extraction
FLPI	Fuzzy Low-pressure Injection
FHPE	Fuzzy High-pressure Extraction

LIST OF SYMBOLS

e	Error
ce	Change in error
ie	Integral of error
u	Control variable
μ	Degree of membership
C	Capacitance
V	Volume
$p(t)$	Pressure
$v(t)$	Voltage
$i(t)$	Current
$q(t)$	Volumetric flow rate
N_{ind}	Number of individuals
L_{ind}	Number of parameters for optimisation

Chapter 1

Introduction

1.1 Background

1.1.1 Brayton cycle based power plant

In this study a power generation system will be considered that can produce up to 110 MW of electrical power. This system is called a module and can operate in a stand-alone mode, or as part of a power plant that can have more of these units.

This module is a graphite-moderated, helium-cooled reactor that uses the Brayton direct gas cycle to convert the heat, which is generated in the core by nuclear fission. The heat is transferred to the coolant gas (helium), and converted into electrical energy by means of a gas turbo-generator (See Figure 1-1) [1].

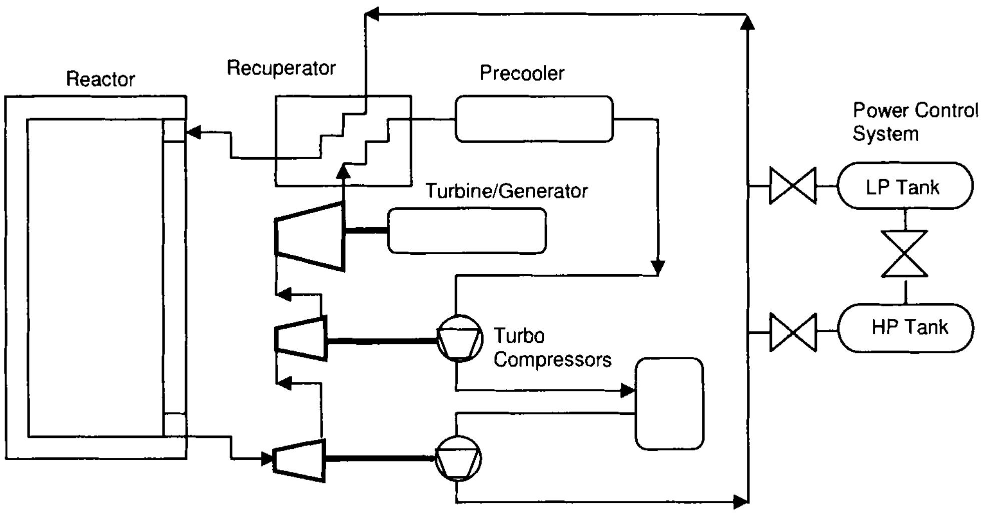


Figure 1-1 Power generation system layout [1]

The Brayton cycle consists of two isentropic and two isobaric processes as shown in Figure 1-2. Starting at 1, gas at a low pressure and temperature is compressed in an isentropic process to a higher pressure and temperature (2). From 2 to 3, the gas is heated in an isobaric (constant pressure) process to the maximum cycle temperature. From 3 to 4, the hot high-pressure gas is expanded isentropically in a turbine to a lower pressure and temperature. The cycle is completed from 4 to 1 by cooling the gas at constant pressure.

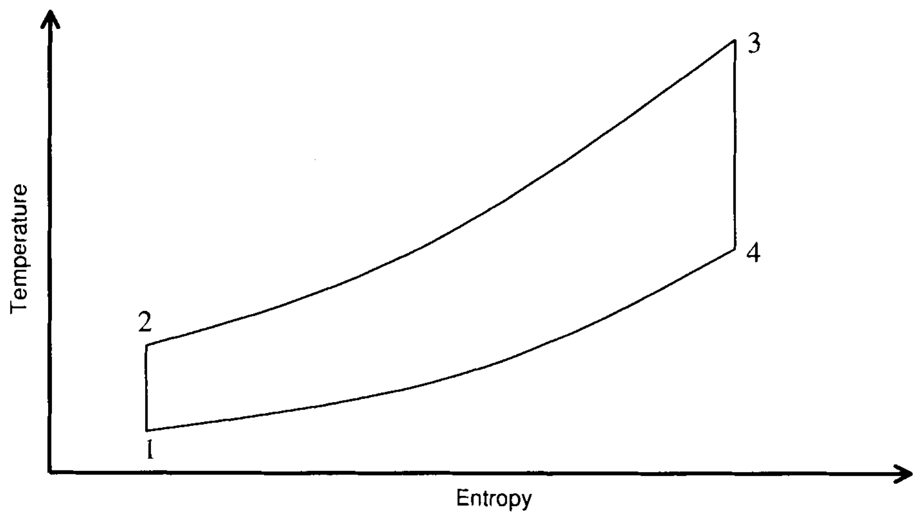


Figure 1-2 The Brayton cycle [1]

1.1.2 Power control

By adding to the gas inventory of the cycle, the electrical power generated will be increased and by removing inventory the power generated will be decreased. This is the primary method of controlling power.

The power control system constitutes four actuators that manipulate the power output of the power plant:

- A booster tank
- A gas cycle bypass control valve
- Low-pressure injection at the low-pressure side of the system
- High-pressure extraction at the high-pressure side of the system

The booster tank is used to increase the power level instantaneously. It can inject only a limited amount of helium at the high-pressure side of the system. Opening the gas cycle bypass control valve will reduce the power and closing it will increase the power.

By opening the gas bypass valve some of the helium that would normally pass through the reactor and turbines is re-circulated through the compressors.

Injection of gas at the low-pressure side of the system does not result in an instant increase in the power output of the system. The power first decreases and then starts to increase as shown in Figure 1-3. This phenomenon is called the non-minimum phase effect [2] and is undesirable. This effect can be avoided by closing the gas cycle bypass control valve while injecting helium at the low-pressure side of the system. Extraction of gas at the high-pressure side results in an instant decrease in the power of the system.

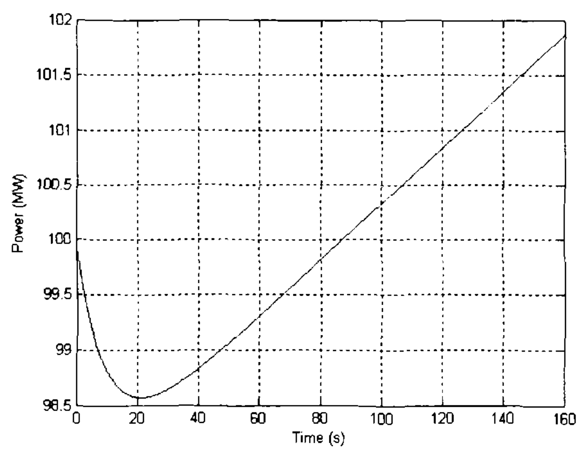


Figure 1-3 Non-minimum phase effect

1.2 Problem statement

A control system needs to be developed that optimally controls the power output of a Brayton cycle based power plant under normal load following conditions. Since it is a multiple variable problem with constraints a sophisticated control strategy is needed.

The control system has to generate set point values for the four helium manipulation actuators previously described. A conceptual diagram for the control system is shown in Figure 1-4. The controller has to use the power error along with other inputs to intelligently adjust the set point values of the actuators in order to control the power output of the plant.

Lastly there exists a need to optimise the power output of the system. A cost function has to be defined that will represent the performance of the system. An optimisation technique is needed to adapt the control system according to this cost function producing an optimal power control system.

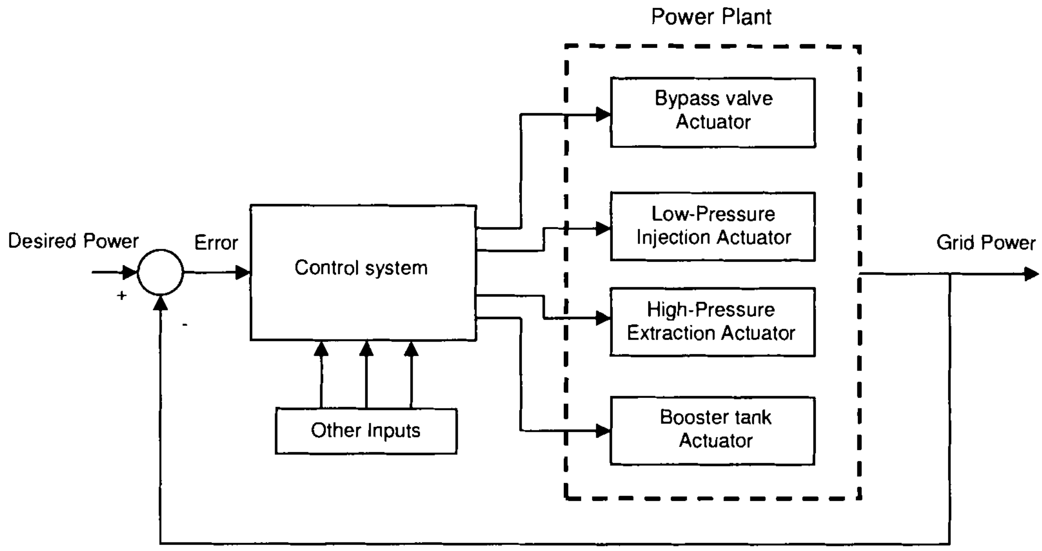


Figure 1-4 Power control system configuration

1.3 Issues to be addressed and methodology

1.3.1 Evaluation platform

A model of the power plant is needed to be able to test different control strategies. An existing linear Simulink® model (Simulink® is a software package used for modelling and simulating dynamic systems) of a Brayton cycle based power plant will be used. This model does not have a boosting capability. A booster tank model therefore has to be derived for the model to be complete.

1.3.2 PID control

This study should also give and indication of the level of sophistication needed to solve this control problem. A PID control strategy will be implemented first. This control strategy will be considered as a bench mark. Other control strategies will be compared with it to see if better control strategies can be found.

A PID control strategy will be devised comprising four subsystems each containing a PID controller. Each subsystem will be responsible for generating the correct set point value for each actuator (See Figure 1-5).

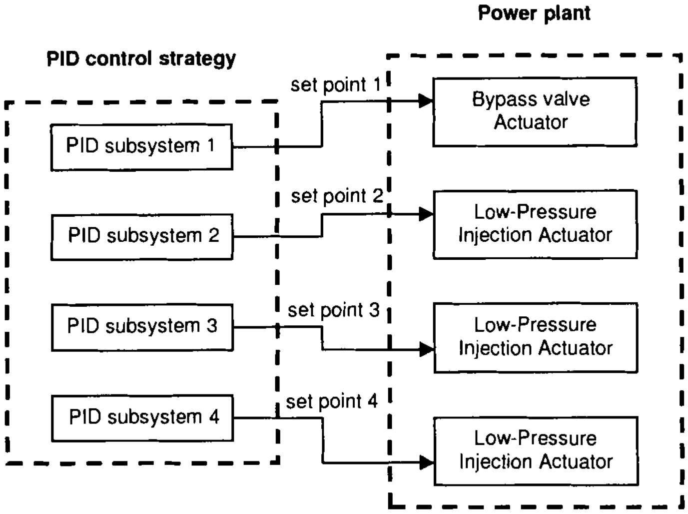


Figure 1-5 PID control strategy

This control strategy will then be tested on the linear model of the power plant using the Simulink® simulation environment.

1.3.3 Fuzzy control

A second control strategy will be investigated called Fuzzy control. Fuzzy control is a non-linear control technique and is therefore more sophisticated than for example PID control. This control technique has to be compared with PID control to see if better results can be obtained. Basically the same strategy as shown in Figure 1-5 will be followed except that each PID controller will be substituted with a Fuzzy controller in each subsystem. This strategy will also be simulated in Simulink®.

1.3.4 Optimised Fuzzy control

Both control strategies described above can be optimised using an optimisation technique. It was decided to optimise only the Fuzzy control strategy by means of a Genetic Algorithm (GA). There are many ways of adapting Fuzzy controllers with

Genetic algorithms [3] [4]. A specific Fuzzy control configuration called Fuzzy PID control seems to be the most attractive configuration for GA optimisation [5].

1.4 Overview of the dissertation

In chapter 2 the Brayton cycle based power plant is discussed in more detail. Specific attention is given to the plant operation and power output control of the system. The Fuzzy control technique is introduced and basic fundamentals such as Fuzzy sets and Fuzzy membership functions are discussed. Further a specific control configuration called Fuzzy PID control is described. This specific configuration allows a Fuzzy controller to emulate a PID controller under certain assumptions.

Finally the optimisation technique called the Genetic Algorithm (GA) is introduced. A biological overview of the GA is first given followed by descriptions of all its components. Also the field called Genetic Fuzzy systems (GFS) is described. This field is concerned with the hybridisation of GAs and Fuzzy systems.

The linear model of the Brayton cycle based power plant is described in chapter 3. The simulation software package, Simulink® is introduced followed by the derivation of the Simulink® linear model. A booster tank model is derived in this chapter to complete the linear model. Simulation results of the linear Simulink® model are given.

Chapter 4 describes both the PID and Fuzzy control strategies in detail. A performance index is formulated that is used to compare the responses of these control strategies. The control strategies are simulated in the Simulink® environment and the results are discussed.

In chapter 5 the GA optimisation process of the Fuzzy control strategy is discussed. More detail on the specific GA configuration is given in terms of the chromosome format, operators and objective function. Results of the optimised Fuzzy control strategy are discussed.

Finally in chapter 6 concluding remarks are given and recommendations for future work are made.

Chapter 2

LITERATURE OVERVIEW

2.1 Three-shaft Brayton cycle based power station

2.1.1 Introduction

The typical three-shaft Brayton cycle based power station of the original PBMR (Pebble Bed Modular Reactor) design is a nuclear power station that can produce approximately 110 MW of electrical power. It is a graphite-moderated, helium-cooled reactor that uses the Brayton direct gas cycle to convert the heat, which is generated in the core by nuclear fission. Heat is transferred to the coolant gas (in this case helium), and converted into electrical energy by means of a gas turbo-generator [1].

2.1.2 Operation of the power plant

At full power conditions, helium enters the reactor at a temperature of approximately 500 °C and at a pressure of 70 bar. The helium moves downward between hot fuel spheres. It then picks up heat from the fuel spheres, which have been heated by the nuclear reaction. The helium leaves the reactor at a very high temperature of approximately 900 °C [1].

Next, the helium gas moves through the High-pressure Turbine (HPT) and drives the High-pressure Compressor (HPC) (See Figure 2-1). Then the helium moves through the Low-pressure Turbine (LPT), which drives the Low-pressure Compressor (LPC).

From the LPC the helium moves through the Power Turbine (PT), which drives the generator. At this stage the helium is still at a very high temperature. It passes through the recuperator where the heat is transferred between the high temperature helium from the PT and the low temperature helium returning to the reactor.

The helium is cooled down by means of a pre-cooler. This increases the density of the helium and improves the efficiency of the compressors. The LPC compresses the helium. From the LPC the helium is cooled down further when it passes through the intercooler. The helium is compressed further by the HPC and then passes through the recuperator that heats the helium up again. The helium then returns to the reactor.

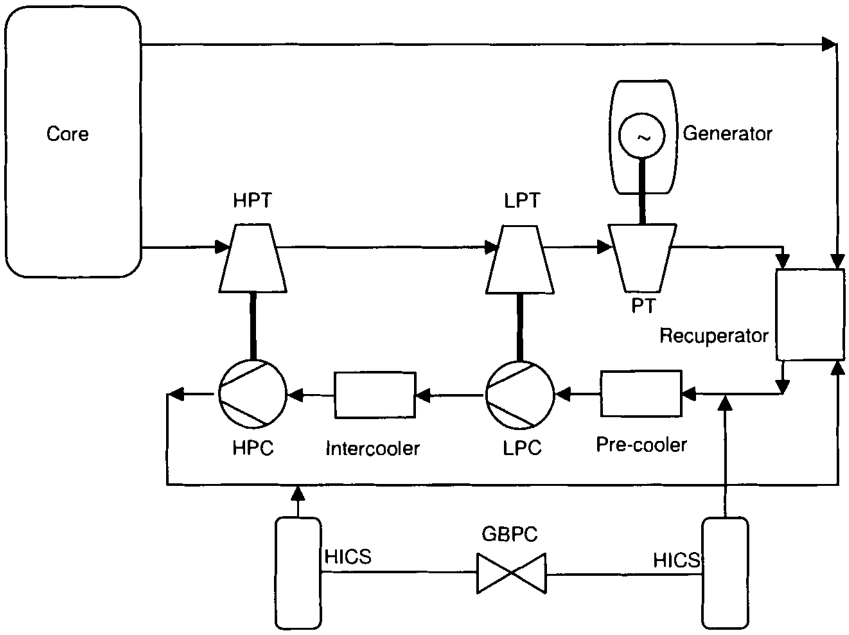


Figure 2-1 Detailed power plant layout

2.1.3 Power output control

Power output control is achieved by adding (or removing) helium to the circuit. This increases (or decreases) the pressures and mass flow rate without changing the gas temperatures or the pressure ratios of the system. The increased pressure and subsequent increased mass flow rate increases the heat transfer rate, thus increasing the power. Power reduction is achieved by removing gas from the circuit.

The power control system is supplied by a series of helium storage tanks ranging from low to high pressure to maintain the required gas pressure in the circuit. Adjustable stator blades on the turbo machinery and bypass flow are used to achieve short-term control [2].

In Figure 2-1 a more detailed system layout of the power plant is given. The Helium Inventory Control System (HICS) and the Gas Cycle Bypass Control Valve (GBPC) are also shown. These two mechanisms are used to adjust the helium inventory of the cycle.

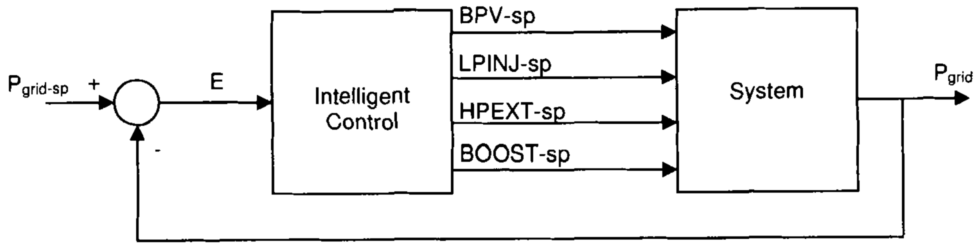
Helium is normally injected into the cycle at the inlet of the pre-cooler and removed at the manifold (Figure 2-1) [2]. A limited amount of helium can also be injected into the manifold by the HICS booster tank. Opening the GBPC will reduce the generated electrical power and closing it will increase the power. For small power level changes the GBPC allows the system to change from one power level to another very quickly. There are also no minimum phase effects.

The HICS and GBPC can be summarised into four helium inventory actuators as described previously, they are:

- A booster tank
- A Gas cycle bypass control valve
- Low-pressure injection at the low-pressure side of the system
- High-pressure extraction at the high-pressure side of the system

The difference between the required power set point and the actual electrical power generated determines whether helium should be injected or removed from the cycle. This power error is used by some type of controller to generate set point values for the four actuators (See Figure 2-2).

These set point values have to be intelligently adjusted, because their responses are interdependent. An intelligent control strategy is therefore needed to generate set point values that will result in smooth power responses and sufficient reserve capacity in stand by mode. A possible control technique that will be able to intelligently adjust the four set point values and satisfy the control demands is Fuzzy control.



BPV-sp	Gas Cycle bypass control valve set point	LPINJ-sp	Low pressure injection set point
HPEXT-sp	High pressure extraction set point	BOOST-sp	Booster tank set point
P_{grid}	Grid power (actual)	$P_{grid-sp}$	Grid power set point (desired)

Figure 2-2 Power control layout for the power plant

2.2 Fuzzy Control

2.2.1 Introduction to Fuzzy Control

Fuzzy logic is a non-linear control technique. It starts with and builds on a set of user-supplied human language rules. The fuzzy system converts these rules to their mathematical equivalents. Fuzzy logic can deal with problems with imprecise and incomplete data, and it can model non-linear functions of arbitrary complexity. In a case where a good plant model can't be found or the system is changing, Fuzzy control can produce a better solution than conventional control techniques [3] [5].

Fuzzy controllers consist of a number of conditional "if-then" rules. For the designer who understands the system, these rules are sometimes easy to write, and as many rules as necessary can be supplied to adequately describe the system.

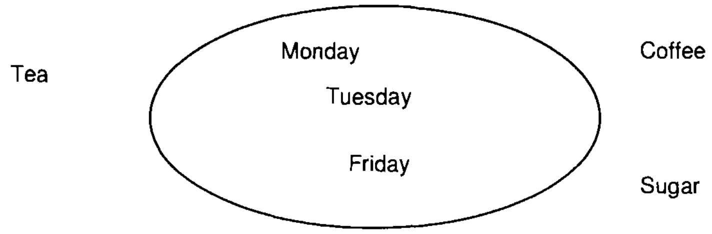
2.2.2 Foundations of Fuzzy logic

Two general concepts need to be introduced to grasp the concept of Fuzzy logic. These two concepts are Fuzzy sets and Fuzzy membership functions.

2.2.2.1 Fuzzy sets

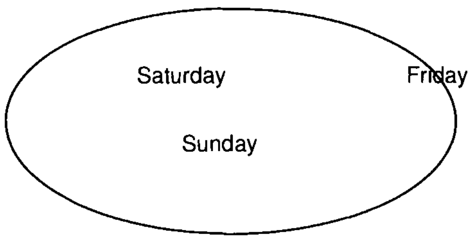
Fuzzy logic starts with the concept of a Fuzzy set. A Fuzzy set is a set without a crisp, clearly defined boundary. It can contain elements with only a partial degree of membership.

To describe a Fuzzy set further it is helpful to first explain what is meant by a classical set. A classical set is a set that completely includes or completely excludes any given element (see Figure 2-3). Consider the days of the week as a classical set. Monday, Tuesday, and Friday are included in this set. It excludes elements such as coffee, sugar, and tea.



Days of the Week
Figure 2-3 Classical set [6]

Consider the set of days comprising a weekend. In Figure 2-4 the set attempts at classifying the weekend days.



Days of the weekend
Figure 2-4 Fuzzy set [6]

Most would agree that Saturday and Sunday belong to the set weekend, but what about Friday? It feels like it can be part of the week and weekend sets. Classical sets would not normally tolerate this kind of thing. This leads to the statement that in Fuzzy logic, the truth of any statement becomes a matter of degree. Reasoning in Fuzzy logic is just a matter of generalizing the familiar yes-no (Boolean) logic. If "true" is given the numerical value of 1 and "false" the numerical value 0, Fuzzy logic also permits in-between values like 0.2 and 0.7453 [6]. For example:

Question:	Is Saturday a weekend day?
Answer:	1 (yes, or true)
Question:	Is Monday a weekend day?
Answer:	0 (no, or false)
Question:	Is Friday a weekend day?
Answer:	0.7 (for the most part yes, but not completely)
Question:	Is Sunday a weekend?
Answer:	0.95 (yes, but not quite as much as Saturday)

2.2.2.2 Fuzzy membership functions

A membership function is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The input space is sometimes referred to as the universe of discourse.

Consider a Fuzzy set called tall people. It includes all potential heights from 1.8 m to 2.5 m. The word "tall" would correspond to a curve that defines the degree to which a person is tall. If the set of tall people is given a crisp boundary of a classical set, we might say all people taller than 2 m are officially considered tall. But this is unreasonable to call one person short and another one tall when they differ in height by the width of a hair [6].

Figure 2-5 below shows a smoothly varying curve that passes from not-tall to tall. The output-axis is a number known as the membership value between 0 and 1. The curve is known as a membership function and is given the designation μ. The curve defines the transition from not-tall to tall.

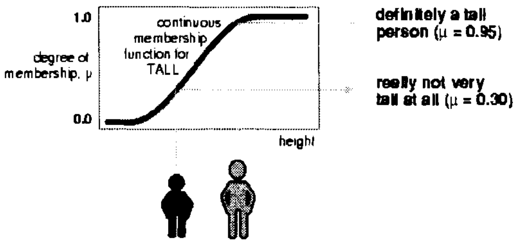


Figure 2-5 Membership function of tall [6]

2.2.3 A Fuzzy controller

A Fuzzy controller consists of a user interface, a rule base and an inference engine. A block diagram of a Fuzzy control system is shown in Figure 2-6. The user interface is often given as a diagram on a graphical screen showing the overall architecture of the control system. Or it can be given as a matrix. It is possible to see the Fuzzy set definitions by means of graphs.

The rule base is a collection of stored control rules in the following format:

$$R_i : \text{If } x \text{ is } A_i \text{ and } y \text{ is } B_i, \text{Then } z \text{ is } C_i$$

where x and y, the inputs, are measured variables, and z is the controller output or the control action; A_i, B_i, C_i are linguistic terms, such as 'low', 'medium' or 'high' [6]. The 'if' part, is called the premise or condition. The 'then' part is called the consequence or action. An if-then rule is mathematically speaking an implication. Fuzzy inference is the process of formulating the mapping from a given input to an output using Fuzzy logic.

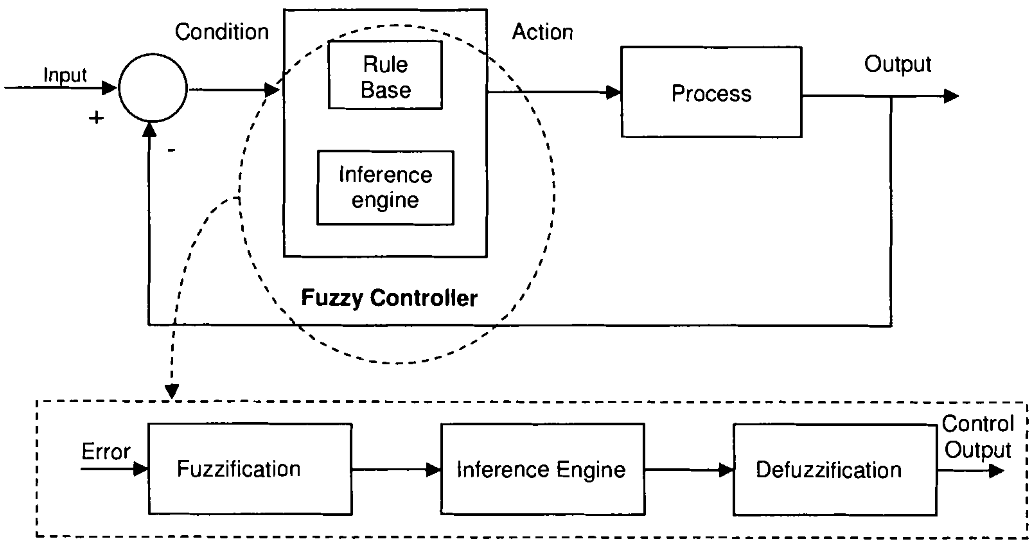


Figure 2-6 Components of a Fuzzy controller

The fuzzification comprises the process of transforming crisp values into grades of membership for linguistic terms of Fuzzy sets. The membership function is used to associate a grade to each linguistic term. The Fuzzy inference engine combines the facts obtained from the fuzzification with the rule base and conducts the fuzzy reasoning process. The transformation from a Fuzzy set to a crisp number is called defuzzification.

A Fuzzy controller contains many parameters that can be adapted to obtain a more optimal controller. There are a lot of optimisation techniques that can be used to optimise Fuzzy controllers. One of the most recent trends is to use Genetic Algorithms.

2.3 Design of Fuzzy PID controllers

2.3.1 Introduction

A Fuzzy controller can be regarded as a superset of linear controllers [7]. Hence the former can emulate the latter under certain assumptions. In this section the relationship between a PID controller and a Fuzzy controller will be investigated. A Fuzzy PID controller will be derived that emulates a PID controller.

2.3.2 Conventional PID controllers

In conventional PID controllers, the control variable $u(t)$ is defined in terms of deviations from or errors $e(t)$ between a reference value y_{ref} and the process output $y(t)$ (i.e. $e(t) = y_{ref} - y(t)$):

$$u(t) = G_p \cdot e(t) + G_i \int_0^t e(t)dt + G_d \cdot \frac{de(t)}{dt} \tag{2-1}$$

where G_p, G_i and G_d are proportional, integral and derivative gains, respectively.

In order to emulate a PID controller through a linear Fuzzy controller, the summation in the PID control equation has to be replaced by a Fuzzy rule base acting like a summation.

2.3.3 Fuzzy PID controllers

A Fuzzy PID (FPID) controller uses the variables error (e), change of error (ce), and integral of error (ie) in the antecedent of if-then rules and the control variable (u) as the consequent. A Fuzzy controller based on the Mamdani-type Fuzzy inferences would consist of rules having the form:

$$R_n : \text{if } (e \text{ is } A_{1,n}) \text{ and } (ce \text{ is } A_{2,n}) \text{ and } (ie \text{ is } A_{3,n}) \text{ then } (u \text{ is } B_n)$$

where n is the rule number and $A_{i,j}$ and B_n are Fuzzy sets [7].

A Fuzzy controller can be represented as an input-output mapping. In the general case it may result in a non-linear shaped control hyper surface. When three inputs (e, ce, ie) and one output (u) are considered, this mapping becomes:

$$u = f(e, ce, ie) \tag{2-2}$$

However assumptions need to be made to allow the Fuzzy rule base acting like a summation and resulting in a linear mapping:

$$u_t = G_p \cdot e_t + G_i \cdot ie_t + G_d \cdot ce_t \tag{2-3}$$

2.3.4 Assumptions for a Linear Fuzzy controller

A Fuzzy controller can be converted to a linear Fuzzy controller only by making certain assumptions with respect to the input universes, rules and membership functions and Fuzzy connectives:

- **Input universes**

  The input universes must be large enough for the inputs to stay within the limits (no saturation). Each input family should contain a number of terms, designed such that the sum of membership values for each input is 1. This can be achieved when the sets are triangular and cross their neighbouring sets at the membership value $\mu = 0.5$; their peaks thus being equidistant. Any input value can thus be a member of at most two sets, and its membership of each is a linear function of the input value [8].

- **Fuzzy rules and membership functions**
  The number of terms in each family determines the number of rules, as they must be the **and** combination (outer product) of all terms to ensure completeness. The output sets should preferably be singletons equal to the sum of the peak positions of the input sets. The output sets may also be triangles, symmetric about their peaks, but singletons make defuzzification simpler [8].

- **Fuzzy connectives**
  To ensure linearity, the algebraic product must be chosen for the connective **and**. Using the weighted average of rule contributions for the control signal (corresponding to the centre of gravity defuzzification), the denominator vanishes, because all firing strengths add up to 1 [8].

2.3.5 Fuzzy PID controller gains

The next step in the design procedure is to transfer the three gains (G_p, G_i and G_d) used in the conventional PID controller to four gains (FG_p, FG_i, FG_d and FG_u) that are necessary for tuning and scaling the FPID controller. The latter emulates the former if the following condition is met:

$$\begin{aligned} u_t &= G_p \cdot e_t + G_d \cdot ce_t + G_i \cdot ie_t \\ &= \left[FG_p \cdot e_t + FG_d \cdot ce_t + FG_i \cdot ie_t\right] \cdot FG_u \\ &= FG_p \cdot FG_u \cdot e_t + FG_d \cdot FG_u \cdot ce_t + FG_i \cdot FG_u \cdot ie_t \end{aligned} \tag{2-4}$$

Comparing the gains of the FPID controller with the gains of the conventional PID controller, the following relations can be derived [7] [8]:

$$FG_p \cdot FG_U = G_p \Rightarrow FG_u = \frac{1}{FG_p} \cdot G_p \tag{2-5}$$

$$FG_d \cdot FG_u = G_d \Rightarrow FG_d = FG_d \cdot \frac{G_d}{G_p} \tag{2-6}$$

$$FG_i \cdot FG_u = G_i \Rightarrow FG_i = FG_p \cdot \frac{G_i}{G_p} \tag{2-7}$$

These relations can be used when a PID controller exists already and an equivalent FPID controller needs to be derived. The FPID controller scheme is shown in Figure 2-7.

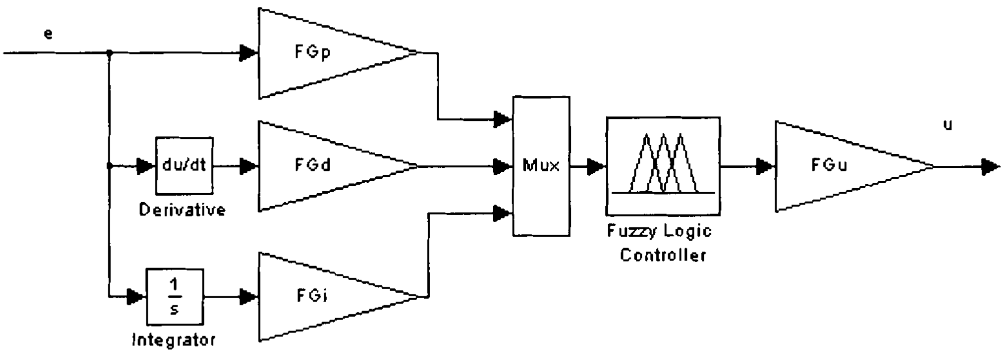


Figure 2-7 Fuzzy PID controller

A rule base with three inputs, however, easily becomes very big and rules concerning the integral action are troublesome [8]. It is therefore common to separate the integral action to form a Fuzzy PD+I controller as in Figure 2-8.

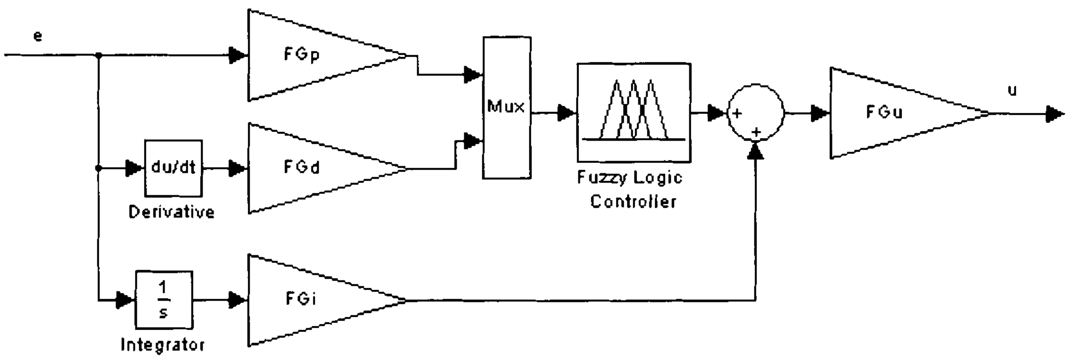


Figure 2-8 Fuzzy PD+I (FPD+I) controller

The controller function is thus split into two additive parts:

$$u = u_{FPD} + u_I = f_{FPD}(e, ce) + f_I(ie) \tag{2-8}$$

2.3.6 Construction of the Fuzzy rule base

The construction of a Fuzzy rule base for a FPD+I controller will now be illustrated. A standard universe of discourse (say [-100,100]) for both inputs: error and change of error. For the sake of simplicity, the same Fuzzy partition will be considered in both cases (see Figure 2-9, where 'N', 'AZ', and 'P' stand for 'Negative', 'About Zero' and 'Positive', respectively).

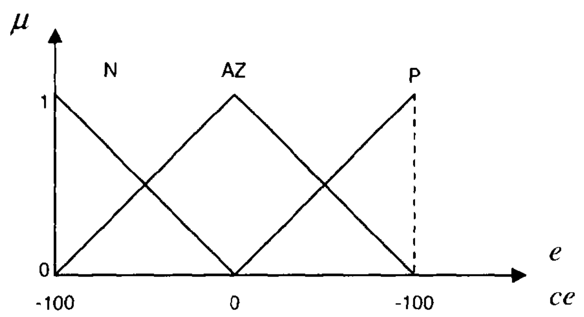


Figure 2-9 Fuzzy Partition on the universe of discourse [-100,100]

Due to the summation, the universe for the output variable u will be [-200,200]. According to the assumptions, Fuzzy singletons (whose positions are determined by the sum of the peak positions of the input sets) will be chosen as consequents for the Fuzzy rules (Figure 2-10).

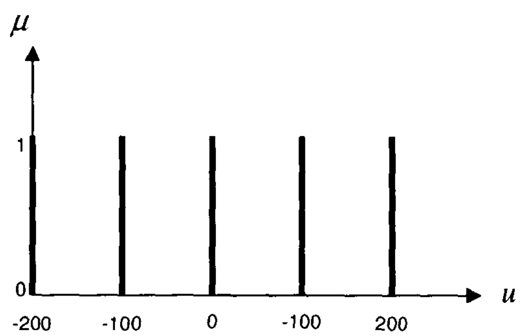


Figure 2-10 Singleton consequents

2.4 Genetic Algorithms

2.4.1 Introduction

Humans and animals are specialised and can very quickly adapt to their environment. Each animal has certain characteristics that allow it to operate very well in a certain environment. The question is where do these characteristics come from? All animals and humans have a unique genetic structure. This genetic structure gives individuals their specific character and it also determines how well they are adapted to their environment (also called the fitness of an individual).

Genetic Algorithms (GAs) are general search algorithms that imitate natural biological evolution. The idea is to evolve populations of individuals that are better adapted to their environment than the individuals from which they are created. GAs operate on a population of potential solutions applying the principle of survival of the fittest to produce successively better approximations to a solution. At each generation of a GA a new set of approximations is created by the process of selecting individuals according to their level of fitness and reproducing them using operators borrowed from natural genetics.

In the past few years there has been an increasing amount of interest of applying GAs in modern control system engineering [9]. Compared to traditional search and optimisation techniques the GA is more robust, global and generally more straightforward to use when there is little or no information about the process to be controlled. The GA does not require any derivative information or an estimation of the solution space. GAs have a stochastic nature (random search characteristic) which largely increases the likelihood of finding a global optimum.

Nowadays Fuzzy logic is increasingly used in decision-aided systems because it offers several advantages over other traditional decision-making techniques. The Fuzzy decision support systems (FDSS) can easily deal with incomplete and/or imprecise knowledge applied to either linear or non-linear problems [9].

These systems have successfully been applied to many different problems such as: predictive maintenance, tool conditions monitoring, job dispatching and tolerance allocation. Unfortunately, all these cases require and expert in order to manually construct, from his/her own expertise, the Fuzzy knowledge databases. Obviously, this learning process is lengthy.

Moreover, the quality of the resulting knowledge base depends greatly on the objectivity and the teaching capacity of the expert. Consequently, many research works have been conducted toward the automatic generation of Fuzzy knowledge bases. These works have first focused on different aspects of the automatic generation of fuzzy rules with either numerical methods or Genetic Algorithms (GA). Although the number of rules increases exponentially with the number of fuzzy sets, GAs appear to be the most promising learning tool.

2.4.2 Biological Overview

2.4.2.1 Chromosomes

All living organisms consist of cells. In each cell there is the same set of chromosomes. Chromosomes are strings of DNA and serve as a model for the whole organism. A chromosome consists of genes, (blocks of DNA). Each gene encodes a particular protein. Basically, it can be said that each gene encodes a trait (feature), for example eye color. Possible settings for a trait (e.g. blue, brown) are called alleles (feature value or feature setting). Each gene has its own position in the chromosome. This position is called a locus [9].

A complete set of genetic material (all the chromosomes) is called a genome and a particular set of genes in a genome is called genotype. The genotype is with later development after birth the base for the organism's phenotype (its physical and mental characteristics etc.)

2.4.2.2 Reproduction

During reproduction, recombination (or crossover) first occurs. Genes from parents combine to form a whole new chromosome. The newly created offspring can then be mutated. Mutation means that the elements of DNA are changed. These changes are mainly caused by errors in copying genes from parents. The fitness of an organism is measured by the success of the organism in its life (survival).

2.4.3 Basic description of a Genetic Algorithm

The algorithm begins with a set of solutions (represented by chromosomes) called a population. Solutions from one population are taken and used to form a new population. This is motivated by a hope that the new population will be better than the old one. Solutions that are selected to form new solutions (offspring) are selected according to their fitness - the more suitable they are the more chances they have to reproduce.

This is repeated until some condition (for example number of populations or improvement of the best solution) is satisfied.

2.4.3.1 Outline of a Genetic Algorithm

The outline of the basic GA is very general (See Figure 2-11). There are many parameters and settings that can be implemented differently in various problems. The first question to ask is how to create chromosomes and what type of encoding to choose.

The next question is how to select parents for crossover. This can be done in many ways, but the main idea is to select the better parents (best survivors) in the hope that the better parents will produce better offspring.

Generating populations from only two parents may cause you to loose the best chromosome from the last population. This is true, and so elitism is often used. This means, that at least one of a generation's best solutions is copied without changes to a new population, so the best solution can survive to the succeeding generation.

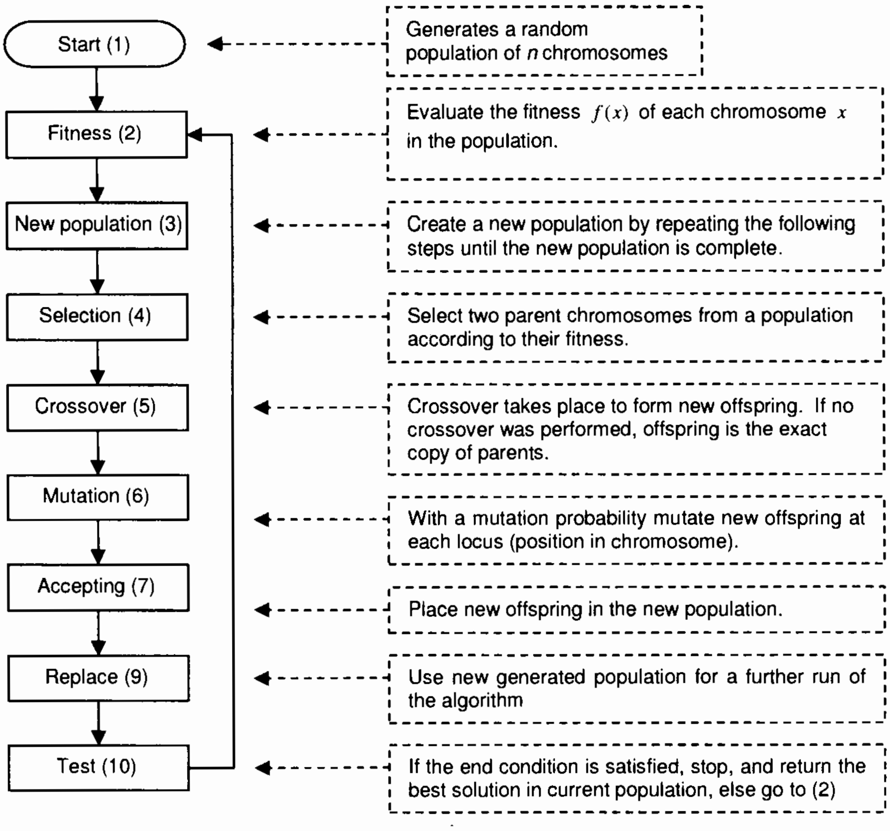


Figure 2-11 Basic outline of the GA

2.4.4 Major elements of Genetic Algorithms

As can be seen from the genetic algorithm outline, the crossover and mutation operators are the most important parts of the genetic algorithm. Mainly these two operators influence the performance. Before we can explain more about crossover and mutation, some information about chromosomes will be given.

2.4.4.1 Encoding of a chromosome

A chromosome should in some way contain information about the solution it represents. The most used way of encoding is a binary string. A chromosome then could look like this:

Chromosome 1	1101100100110110
Chromosome 2	1101111000011110

Table 2-1 Chromosome representation

2.4.4.2 Population size

Population size: How many chromosomes are in population (in one generation). If there are too few chromosomes, the GA will have too few possibilities to perform crossover on and only a small part of search space is explored. On the other hand, if there are too many chromosomes, GAs slow down. Research shows that after some limit (which depends mainly on encoding and the problem) it is not useful to use very large populations because it does not solve the problem faster than moderate sized populations.

2.4.4.3 Objective and fitness functions

The objective function is used to provide a measure of how well the individual (a particular solution, also called a chromosome) solves the problem. In the case of a minimisation problem, the fittest individuals will have the lowest numerical value of the associated objective function. This raw measure of fitness is usually only used as an intermediate stage in determining the relative performance of individuals in a GA.

The fitness function is used to transform the objective function into a measure of relative fitness. Thus, where f is the objective function, g transforms the value of the objective function to a non-negative number and F is the resulting relative fitness [10].

$$F(x_i) = g(f(x)) \qquad \textbf{(2-9)}$$

This mapping is always necessary when the objective function is to be minimised as the lower objective function values correspond to fitter individuals. In many cases, the fitness function value corresponds to the number of offspring that an individual can expect to produce in the next generation.

A commonly used transformation is that of proportional fitness assignment. The individual fitness $F(x_i)$ of each individual is computed as the individual's raw performance, $f(x_i)$ relative to the whole population, i.e.,

$$F(\mathrm{x}_i) = \frac{f(\mathrm{x}_i)}{\sum_{i=1}^{N_{ind}} f(\mathrm{x}_i)} \tag{2-10}$$

where N_{ind} is the populations size and x_i is the phenotypic value of individual i. Whilst this fitness assignment ensures that each individual has a probability of reproducing according to its relative fitness, it fails to account for negative objective function values.

A linear transformation (2-11) which offsets the objective function is often used prior to fitness assignment where a is a positive scaling factor if the optimisation is maximizing and negative if it is minimising [10].

$$F(\mathrm{x}) = a \times f(\mathrm{x}) + b \tag{2-11}$$

The offset b is used to ensure that the resulting fitness values are non-negative.

2.4.4.4 Selection

Selection models nature's *survival-of-the-fittest* principle. The aim is to ensure that fitter strings (solutions) receive a higher number of offspring, thereby getting a higher chance of surviving in the new generation. Selection strategies include Roulette wheel selection, tournament selection, Ranking selection, Boltzman selection [10], etc.

2.4.4.5 Roulette wheel selection methods

Many selection techniques employ a "roulette wheel" mechanism to probabilistically select individuals based on some measure of their performance [10]. A real-valued integer, called *Sum*, is determined as either the sum of the individuals' expected selection probabilities or the sum of the raw fitness values over all the individuals in the current population. Individuals are then mapped one-to-one into continuous intervals in the range [0,*Sum*].

The size of each individual interval corresponds to the fitness value of the associated individual. For example in Figure 2-12 the circumference of the roulette wheel is the sum of all six individual's fitness values. Individual 5 is the fittest individual and occupies the largest interval, whereas individuals 6 and 4 are the least fit and have correspondingly smaller intervals within the roulette wheel. To select an individual, a random number is generated in the interval [0,*Sum*] and the individual whose segment spans the random number is selected. This process is repeated until the desired number of individuals has been selected.

The basic roulette wheel selection method is stochastic sampling with replacement (SSR). Here, the segment size and selection probability remain the same throughout the selection phase and individuals are selected according to the procedure outlined above. SSR gives zero bias but a potentially unlimited spread. Any individual with a segment size greater than zero could entirely fill the next population.

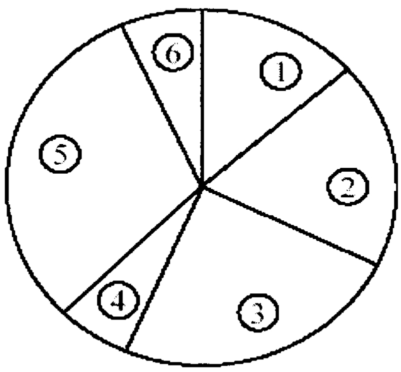


Figure 2-12 Roulette wheel selection [10]

2.4.4.6 Crossover

After we have decided what encoding we will use, we can proceed to crossover operation. Crossover operates on selected genes from parent chromosomes and creates new offspring. The simplest way to do that is to choose randomly some crossover point and copy everything before this point from the first parent and then copy everything after the crossover point from the other parent.

Crossover can be illustrated as follows: (| is the crossover point):

Chromosome 1	11010 | 00100110110
Chromosome 2	11011 | 11000011110
Offspring 1	11011 | 00100110110
Offspring 2	11010 | 11000011110

Table 2-2 Crossover illustration

There are other ways to make crossovers, for example more crossover points can be used. Crossover can be quite complicated and depends mainly on the encoding of chromosomes. A Specific crossover method made for a specific problem can improve the performance of the Genetic Algorithm.

2.4.4.7 Mutation

After a crossover is performed, mutation takes place. Mutation prevents the algorithm from falling into a local optimum. Mutation operation randomly changes the offspring resulted from crossover. In case of a binary encoding we can switch a few randomly chosen bits from 1 to 0 or from 0 to 1. Mutation can be illustrated as follows:

Original offspring 1	1101111000011110
Original offspring 2	1101100100110110
Mutated offspring 1	1100111000011110
Mutated offspring 2	1101101100110100

Table 2-3 Mutation illustration

The technique of mutation (as well as crossover) depends mainly on the encoding of chromosomes. For example when we are encoding permutations, mutation could be performed as an exchange of two genes.

2.5 Genetic Fuzzy System

Fuzzy systems (FSs) is any Fuzzy Logic-Based System, where Fuzzy Logic can be used either as the basis for the representation of different forms of system knowledge, or to model the interactions and relationships among the system variables. Fuzzy systems proved to be a very helpful tool for modelling complex systems where classical tools are too difficult or unsatisfactory [11].

Genetic algorithms provide robust search capabilities in complex spaces, offering a valid approach to problems requiring efficient and effective searching. A lot of papers and applications combining Fuzzy concepts and GAs have appeared lately. There is a particular interest on the use of GAs for designing Fuzzy systems. This field of interest is called Genetic Fuzzy Systems (GFSs).

Defining a FS automatically can be considered as an optimisation or search process. GAs is one of the best global search techniques and has the ability to explore and exploit a given operating space using available performance measures. GAs can find near optimal solutions in complex search spaces.

A priori knowledge may be in the form of linguistic variables, Fuzzy membership function parameters, Fuzzy rules, number of rules, etc. The code structure and independent performance features of GAs make them suitable candidates for incorporating a priori knowledge. These advantages have extended the use of GAs in the development of a wide range of approaches for designing Fuzzy systems. The Genetic Fuzzy concept is illustrated in Figure 2-13.

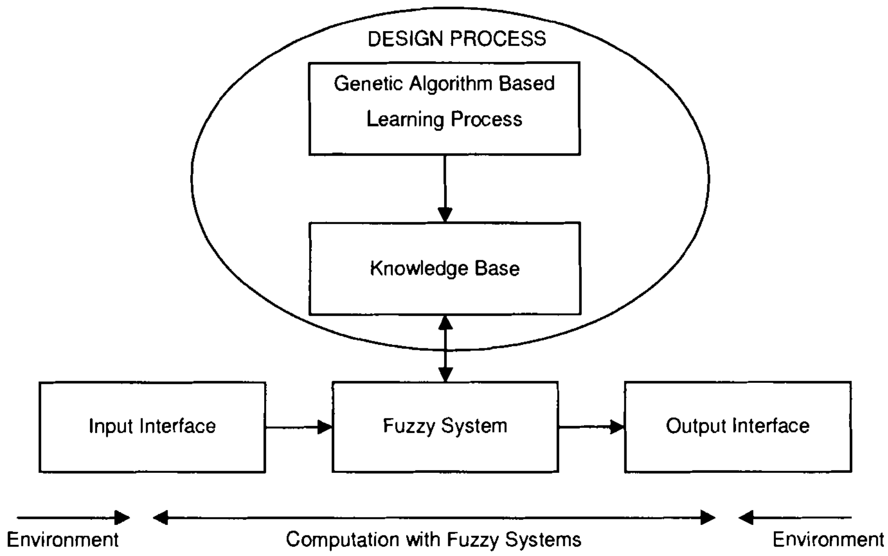


Figure 2-13 Genetic Fuzzy systems [11]

We will focus on Fuzzy Rule Based Systems (FRBSs). The Knowledge Base (KB) of FRBS comprises two components, a Data Base (DB), containing the definitions of the scaling factors and the membership functions of the Fuzzy sets specifying the meaning of the linguistic terms, and a Rule Base (RB), constituted by the collection of Fuzzy rules.

Figure 2-14 shows the structure of a KB integrated in a FS with fuzzification and defuzzification modules, as used in fuzzy control.

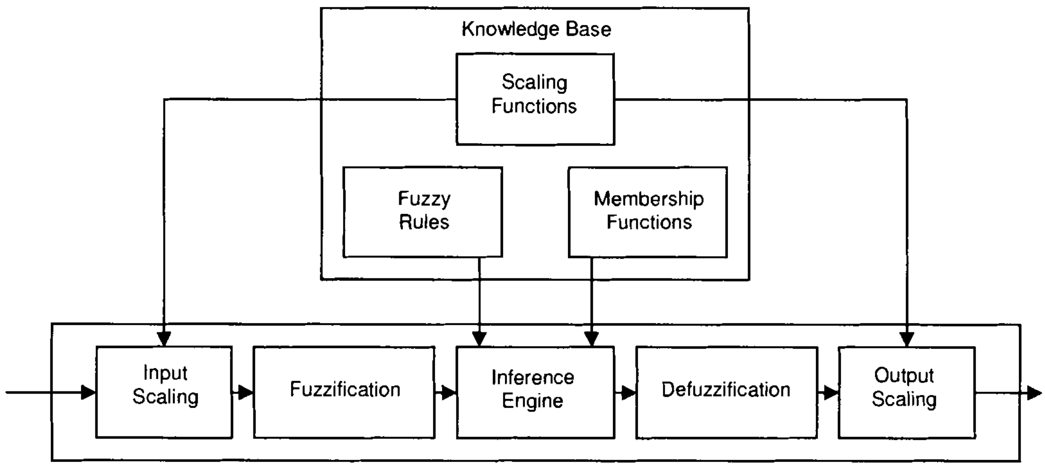


Figure 2-14 Structured knowledge base [11]

It is possible to distinguish between three different groups of Genetic Fuzzy Logic Controller design processes according to the KB components included in the learning process [11]. These ones are the following:

1) **Genetic definition of the Fuzzy Logic Controller Data Base (membership functions):** In this process each chromosome contains the parameters of all the membership functions of the specific Fuzzy system. This means that the membership function shapes are optimised during the learning process.

2) **Genetic derivation of the Fuzzy Logic Controller Rule Base:** In this case each chromosome contains an entire rule base. During the learning process different rule bases is tested to find an optimal rule base (optimal controller).

3) **Genetic learning of the Fuzzy Logic Controller Knowledge Base (membership functions and rule base):** Here the chromosome contains both the membership function parameters and rule base. The learning process optimises both the membership functions as well as the rule base.

2.6 Conclusion

The basic operation of a three-shaft Brayton cycle based power plant was discussed in this chapter. By looking at the power plant layout the four actuators was identified that manipulate the power output of the system. A Fuzzy control system was proposed to intelligently control the set points of these actuators. Fuzzy control proves to be a more robust control technique compared to other control techniques. It can easily form part of a hybrid system, meaning that a learning algorithm can be applied to the Fuzzy system to optimise its parameters.

A Fuzzy PID control technique was introduced in this chapter. Fuzzy controllers are non-linear controllers. These controllers can be regarded as a superset of linear controllers. Hence the Fuzzy controller can emulate a linear controller such as a PID when certain assumptions are made.

Genetic Algorithms is a random search algorithm that can be used to optimise the parameters of the Fuzzy controller. This training algorithm is attractive because it is not based on gradients like most training algorithms. Finally a Genetic Fuzzy controller was proposed to optimally control the power output of the system.

Chapter 3

LINEAR MODEL OF A BRAYTON CYCLE BASED POWER PLANT

3.1 Introduction

The Main Power System (MPS) of a Brayton cycle based power plant constitutes two parts namely the Reactor Unit (RU) and the Power Conversion Unit (PCU) [2]. The RU generates thermal energy by means of a nuclear reaction. This thermal energy is then converted to mechanical work and then to electric energy by the PCU.

The PCU plays a very important role in controlling the power output of the PBMR. A model of the PCU needs to be derived to be able to test different control techniques. In this chapter a linear model of the PCU will be described.

3.2 Description of the power conversion unit

3.2.1 Brayton cycle

The thermal energy of the core is extracted by means of a pressurised helium gas stream and transferred to the PCU. This thermal energy is converted by the PCU to electric energy using a Brayton cycle thermodynamic process. The ideal Brayton Cycle constitutes two isentropic and two isobaric processes shown in Figure 1-2.

The PBMR uses two mechanisms to improve the efficiency of the Brayton cycle. The heat rejected during cooling ((4) to (1)) is used to preheat the gas before it enters the heater. In this model of the Brayton cycle base power plant a multistage compression system with intercooling is used. This more efficient Brayton cycle is also referred to as a recuperative Brayton cycle.

3.2.2 The power conversion unit

The PCU utilises a recuperative Brayton cycle with helium as the working fluid. A schematic diagram of the PCU is shown in Figure 3-1, while the temperature-entropy of the cycle is shown in Figure 3-2. At (1), helium at a relatively low pressure and temperature is compressed by a Low Pressure Compressor (LPC) to an intermediate pressure (2), after which it is cooled in an intercooler to state (3). The intercooling between the two multistage compressors improves the overall cycle efficiency. A High Pressure Compressor (HPC) then compresses the helium to state 4. From (4) to (5), the helium is preheated in the recuperator before entering the reactor that heats the helium to state (6).

After the reactor, the hot high-pressure helium is expanded in a High Pressure Turbine (HPT) to state (7) after which it is further expanded in a Low Pressure Turbine (LPT) to state (8). The HPT drives the HPC while the LPT drives the LPC. After the LPT, the helium is further expanded in the power turbine to the pressure at (9), which is approximately the same as the pressure at (10) and (1).

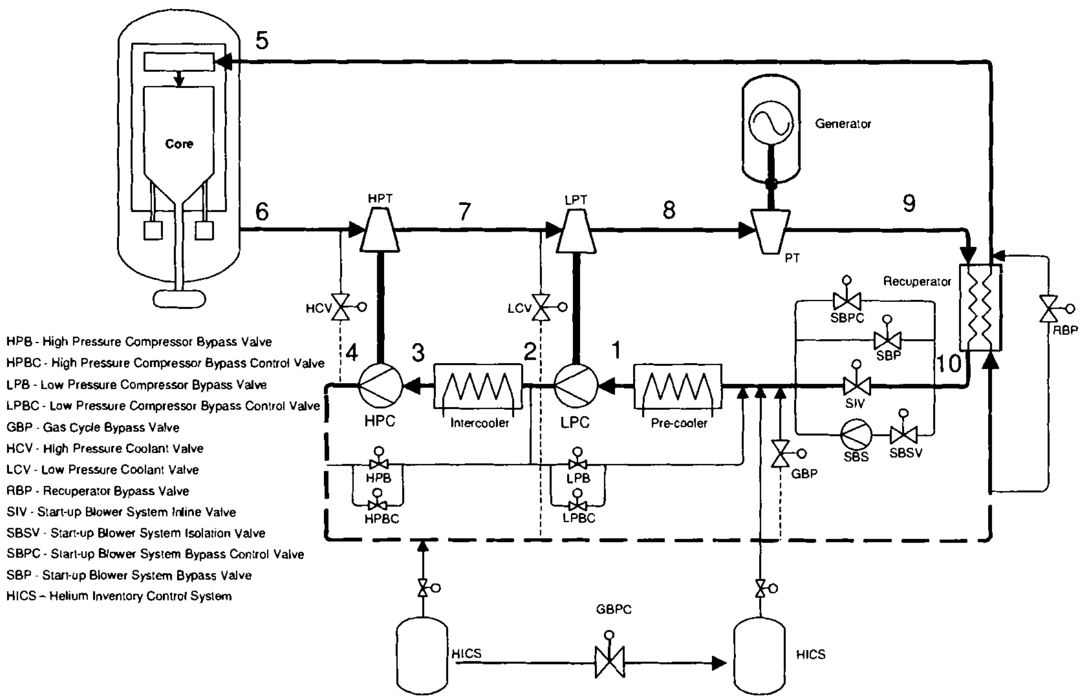


Figure 3-1 Schematic layout of the PCU [2]

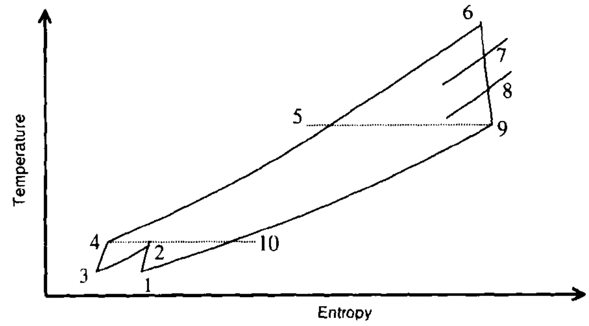


Figure 3-2 Temperature-Entropy diagram of the recuperative Brayton cycle [2]

From (9) to (10), the helium is cooled in the recuperator. The heat taken from the helium at (9) to (10) is equal to the heat transferred to the helium at (4) to (5). The recuperator uses heat from the cooling process that would otherwise be lost to the ultimate heat sink to heat the gas before it enters the reactor, thereby reducing the heating demand on the reactor and increasing the overall plant efficiency.

3.2.3 Power control

In Figure 3-1 the Helium Inventory Control System (HICS) and the Gas Cycle Bypass Control Valve (GBPC) are also shown. These two mechanisms are used to adjust the helium inventory of the cycle. By increasing the inventory the electrical power generated will be increased and by decreasing the inventory the power will be decreased.

Helium is normally injected into the cycle at the inlet of the pre-cooler and removed at the manifold (Figure 3-1) [2]. A limited amount of helium can also be injected into the manifold by the HICS booster tank. Opening the GBPC will reduce the generated electrical power and closing it will increase the power.

3.3 Linear models

3.3.1 Linear model of the turbo-machines

Linear models need to be derived for the turbines and the compressors in the power plant system. It is not in the scope of this chapter to go into detail of the operation of compressors and turbines. A basic model of a compressor or turbine is given in Figure

3-3. This model implements a linear algorithm that simulates either a compressor or a turbine. For a detailed description of the algorithm refer to [12].

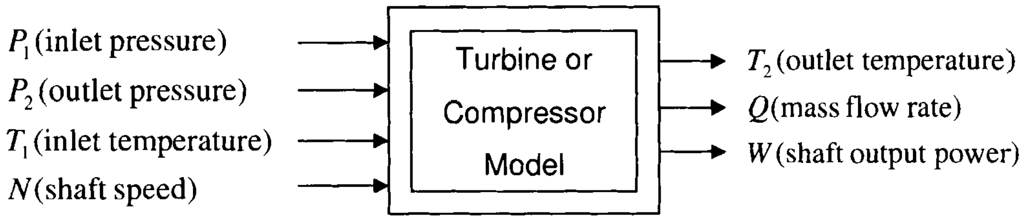


Figure 3-3 Turbine/Compressor model [12]

3.3.2 Linear model of the volumes in the system

It is possible to model certain mechanical and thermodynamic systems with electrical equivalent systems. In [12] it is shown that for fluids, a volume is analogous to a capacitance in an electric circuit, pressure is analogues to voltage and volumetric flow rate (or mass flow rate) is analogous to current. A summary is given in Table 3-1.

Thermodynamic parameter	Electrical parameter
Volume, V	Capacitance, C
Pressure, $p(t)$	Voltage, $v(t)$
Volumetric flow rate, $q(t)$	Current, $i(t)$

Table 3-1 Electric equivalence for thermodynamic systems

Figure 3-4 further demonstrates the analogy. It should be noted that the analogy between a volume and capacitance holds provided that there are no large changes in the temperature of the fluid during a transient.

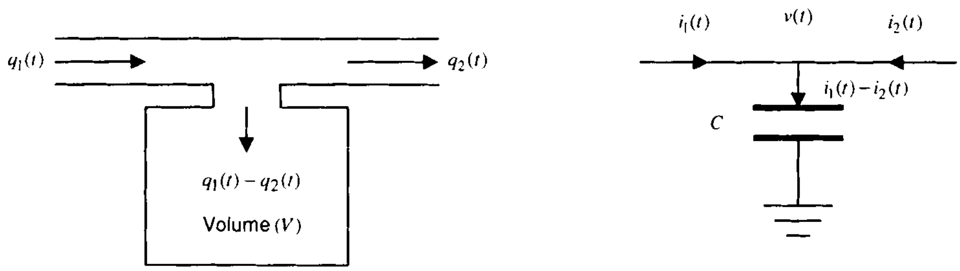


Figure 3-4 Analogy between volume and capacitance

The equations used in the analogy are the continuity equation, (3-1) and Kirchhoff's current law, (3-2).

$$q_1(t) - q_2(t) = V\frac{d}{dt}p(t) \tag{3-1}$$

$$i_1(t) - i_2(t) = C\frac{d}{dt}v(t) \tag{3-2}$$

3.4 Linear model of the PCU

In the previous paragraphs linear models were derived for the components of the PCU. A complete linear model of the PCU can now be constructed. From the schematic layout of the PCU in Figure 3-1 a simplified linear model can be derived (see Figure 3-5).

For modelling purposes the HICS tanks and their inlet/outlet valves are modelled as mass flow sources. Active control on these valves is assumed to ensure that constant mass flow from the HICS into (or out of) the PCU is ensured. Similarly, the mass flow rates through the compressor bypass valves are modelled as constant flow rate sources, because of the active control on these valves.

It is assumed that the effect of the manifold, reactor and HP (High Pressure) side of the recuperator can be modelled simply as a volume (capacitance) between the HP compressor and the HP turbine. The volume between the compressors is not considered to be negligible because the temperature is very low at this point (slightly above the sea water temperature). It is also assumed that a significant capacitance resides between the outlet of the power turbine and the inlet to the low-pressure compressor. This is because there is significant volume between these components and temperature is a mixture of low and high temperature.

Figure 3-5 shows the significant capacitances C_{LP}, C_{MP} and C_{HP} (the subscripts in this case indicate low pressure, medium pressure and high pressure capacitances). The bypass valves and HICS tanks are modelled with mass flow sources where q gives the perturbation from the steady state mass flow at normal power operation.

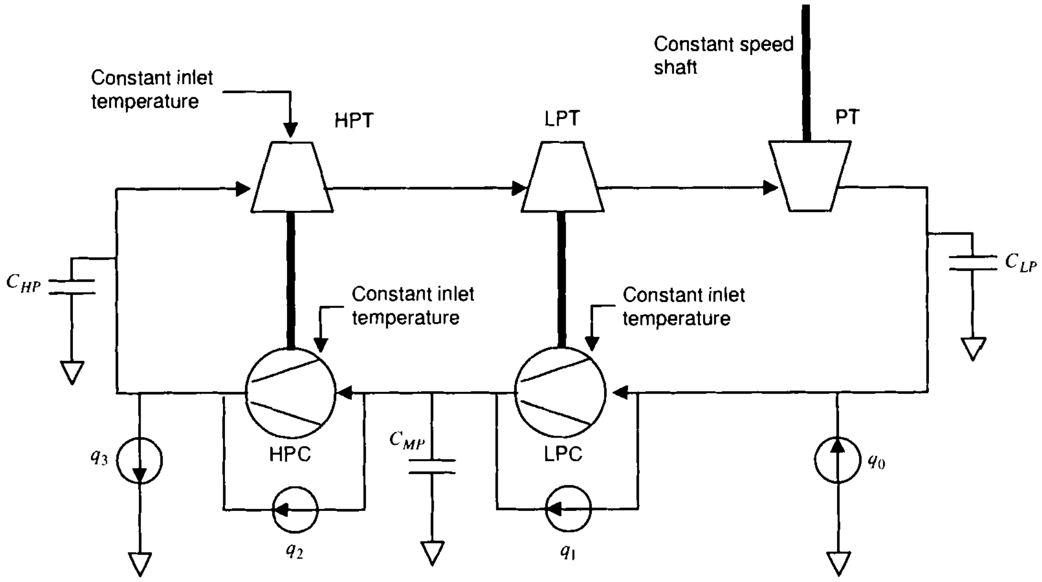


Figure 3-5 Simplified linear model of the PCU [12]

By assuming constant inlet temperature to the high temperature turbine implies that the reactor is modelled as a perfect heat source. That is, as the mass flow increases or decreases in the system, the outlet temperature remains constant. The mass flow source q_0 represents the perturbation in the mass flow being injected from the HICS into the low-pressure side of the circuit and the source q_3 represents the perturbation in helium extraction from the high-pressure side of the circuit to the HICS. The sources q_1 and q_2 represent the perturbation in flow through the compressor bypass valves.

3.5 Simulink® linear model of the PCU

3.5.1 Introduction to Simulink®

Simulink® is a software package used for modelling and simulating dynamic systems. It supports linear and non-linear systems, modelled in continuous time, sampled time, or a hybrid of the two.

For modelling, Simulink® provides a graphical user interface (GUI) for building models as block diagrams, using click-and-drag mouse operations. With this interface, models can be drawn up just as one would with pencil and paper. After the model has been defined, it can be simulated and plots of the results can be generated.

3.5.2 Modelling the PCU in Simulink®

A Simulink® model of the PCU already exists and is shown in Figure 3-6. At the top the three models of the turbines HPT, LPT and PT are shown.

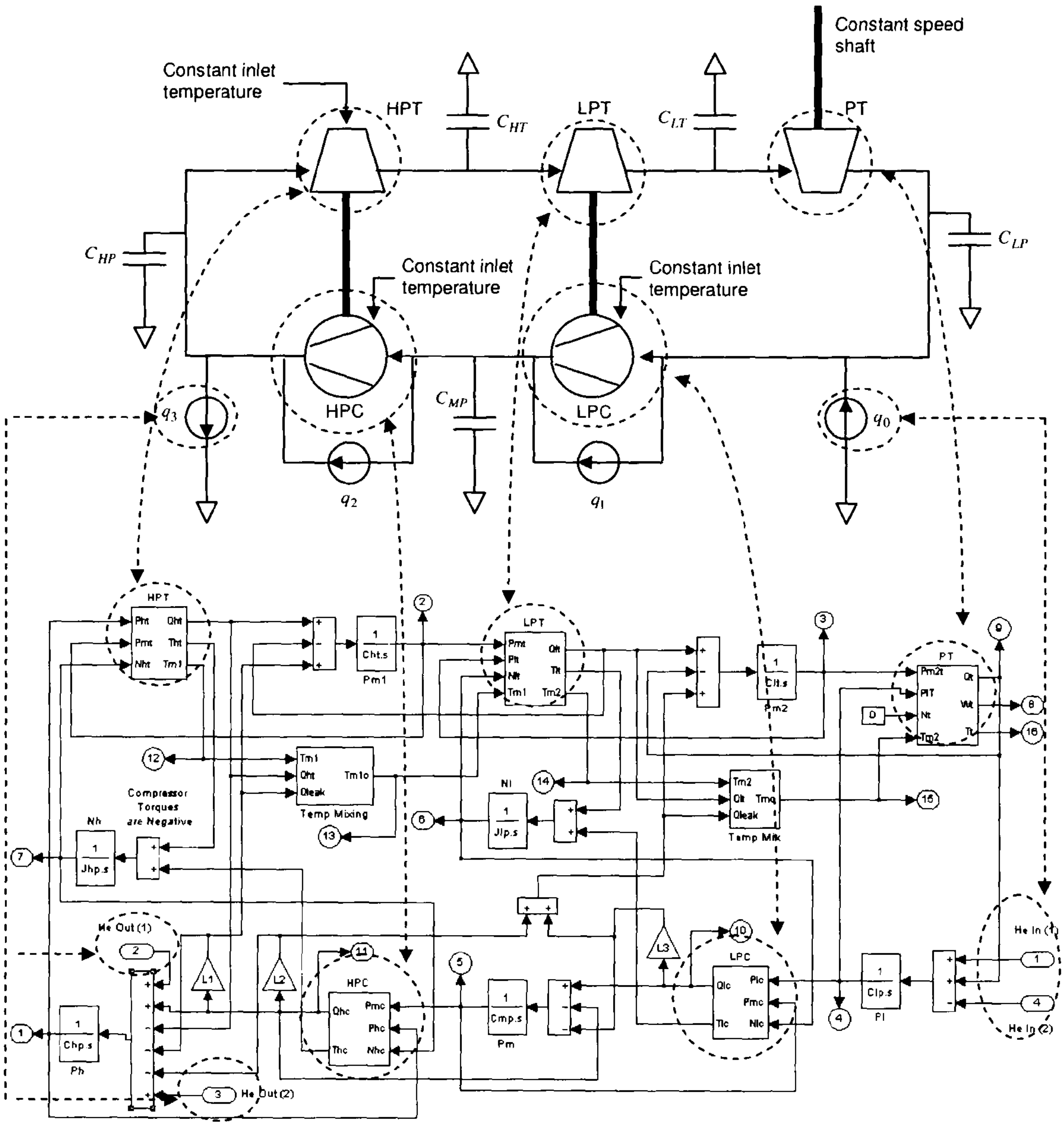


Figure 3-6 Simulink® model of the PCU

At the bottom there is a model for the low-pressure compressor (LPC) and a model for the high-pressure compressor (HPC). The power turbine model block (PT) has an output labelled (Wt) and numbered (8). This output represents the power output of the system.

At the bottom right of the figure there are two inputs numbered (1) and (4). These inputs represent the mass flow source, q_0. Helium is injected through (1) and (4) from the HICS into the low-pressure side of the circuit. At the bottom left of the figure there are two inputs numbered (2) and (3). These inputs represent the mass flow source q_3. Helium is extracted by means of the inputs (2) and (3) from the high-pressure side of the circuit. In the Simulink® model the volumes between the turbines (C_{HT} and C_{LT}) as well as the leak flows and pipe losses at the outlets of the compressors and turbines are taken into consideration in this model.

3.6 Simulation of the PCU model

The model as described is part of work done by J.F Pritchard [12]. This model does not have a high-pressure injection capability. In this section the adjustments made to the PCU model will be discussed. The following paragraphs will be concerned with the booster tank model that was added to the PCU model to give it a high-pressure injection capability.

The Simulink® model has four actuators to manipulate the power output of the power plant at normal load following conditions. If there is a sudden change in the power demand the system must adjust to meet that power demand. The four actuators will be briefly explained.

3.6.1 High pressure injection (Boosting)

Helium gas is injected at the high-pressure side of the system to increase the mass flow in the system. Injection of gas at high-pressure side results in an instant increase in the power output of the system.

Let the perturbation of helium injected at input (3) in Figure 3-6 be equal to one. It is assumed that all other inputs are zero and that the current power output is 100 MW. The power due to boosting is given in Figure 3-7.

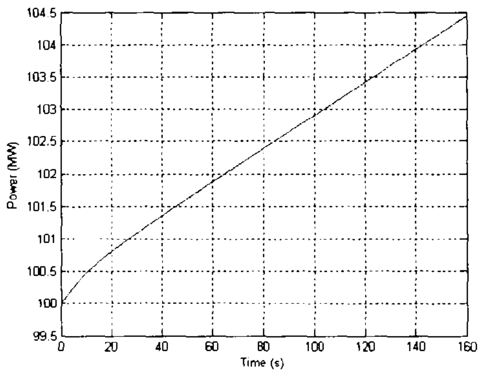


Figure 3-7 Power output during boosting

3.6.2 Low pressure injection

Helium is injected at the low-pressure side of the system to increase the mass flow in the system. Injection of gas at the low-pressure side does not result in an instant increase in the power output of the system. The power will in fact first drop before it starts to increase. This is called the non-minimum phase effect and is undesirable. Let the perturbation of helium injected at input (1) in Figure 3-6 be equal to one. All the other inputs are assumed to be zero and the power output at the beginning of the simulation is 100 MW. The change in power due to helium injection at the low-pressure side is given in Figure 3-8.

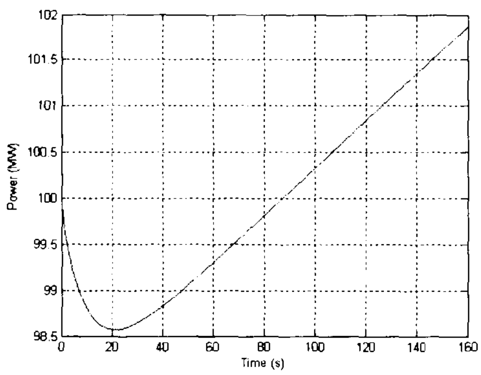


Figure 3-8 Power output during low pressure injection

3.6.3 Gas bypass

In the real world system a gas bypass valve is used for bypassing. Opening the gas bypass valve will reduce the generated electrical power and closing it will increase the power. By opening the gas bypass valve some of the helium that would normally pass through the reactor and turbines is re-circulated through the compressors. The power turbine power is reduced owing to the decrease in mass flow rate and the compressors using proportionately more of the available thermal energy. The power can instantaneously be increased or decreased by using the bypass valve countering the non-minimum phase effect resulting from injecting helium at pre-cooler inlet.

In the Simulink® model gas bypassing is implemented by injecting gas at the low-pressure side and extracting the same amount of gas at the high-pressure side.

Figure 3-9 shows the power output by setting the gas bypass valve perturbation to one. This means that the valve is opened and effectively the power will drop.

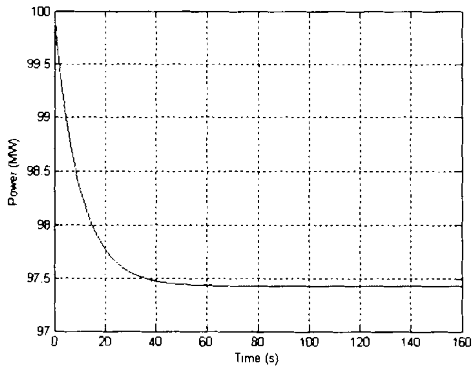


Figure 3-9 Power output due to opening the bypass valve

Figure 3-10 shows the power output by setting the gas bypass valve perturbation to minus one. This means that the valve is closing and the power will increase.

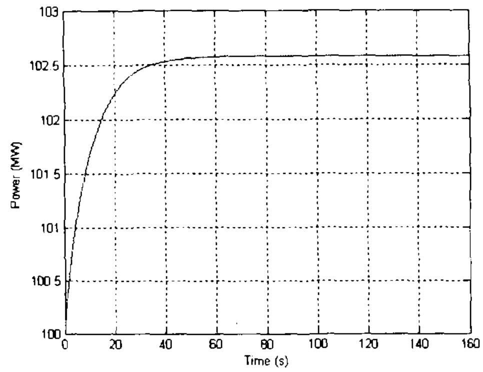


Figure 3-10 Power output due to closing the bypass valve

3.6.4 High pressure extraction

Helium gas is extracted at the high-pressure side of the system to decrease the mass flow in the system. Extraction of gas at the high-pressure side results in an instant decrease in the power of the system. Figure 3-11 shows the power output by setting the high-pressure extraction perturbation to one.

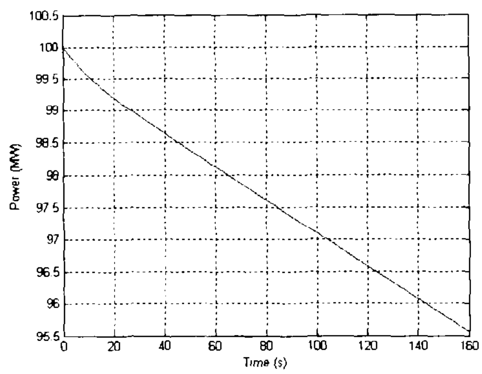


Figure 3-11 Power output due to high pressure extraction

3.6.5 Booster tank introduction

A booster tank makes high-pressure injection possible. Gas is stored in a tank at a higher pressure than that of the system. When a sudden increase in the power demand arises gas can be injected at the high-pressure side of the system. This will result in an instant increase in the power output.

The booster tank has one limitation though. It can only inject a limited amount of gas over a certain period of time. If its pressure drops to a predefined limit, the boosting capability will be disabled. The booster tank's pressure must then be increased to a predefined level before the boosting capability can be restored.

3.6.6 Booster tank system block

A booster tank system block is created (Figure 3-12). This block has two inputs and two outputs. It takes the current pressure of the system at the high-pressure side as an input (SysPres). The rate at which gas must be injected (mass flow) at the high-pressure side during the boosting operation is taken as the second input (Boost).

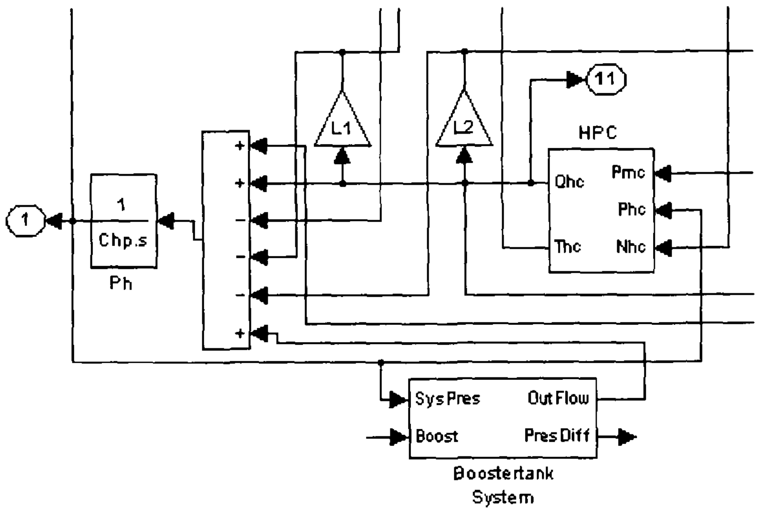


Figure 3-12 Booster tank sub-system

The first output of the booster tank sub-system is the current mass flow out of the booster tank (OutFlow). Secondly the difference between the pressure in the booster tank and the pressure at the high-pressure side of the system is calculated. The pressure difference (PresDiff) is then the second output of the booster tank block. The booster tank has three components called: *Booster tank control*, *Booster tank pressure control* and *Booster tank* (see Figure 3-13).

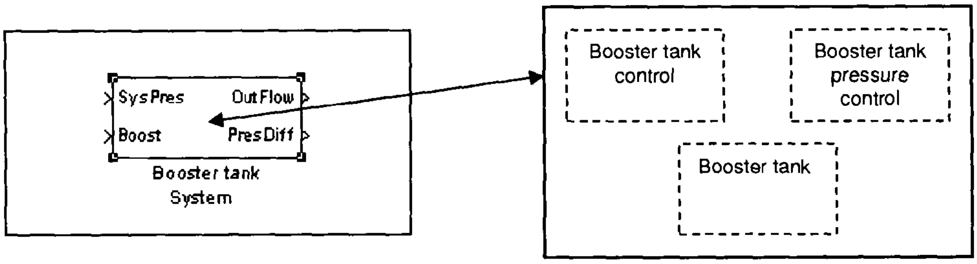


Figure 3-13 Three components of the booster tank system

A detailed layout of the booster tank system is given in Figure 3-14.

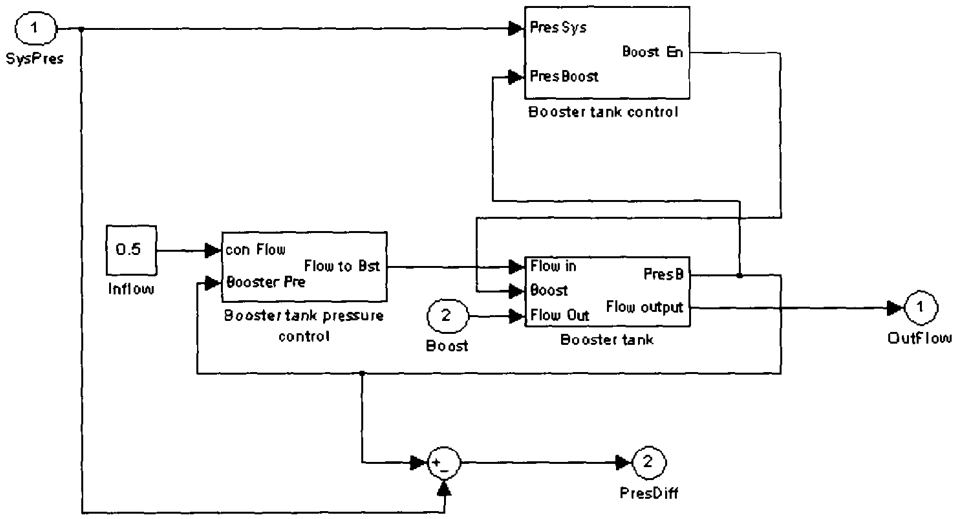


Figure 3-14 Detail of the booster tank system block

3.6.6.1 Booster tank control component

The booster tank control component has two inputs and one output as shown in Figure 3-14. It takes the current pressure of the high-pressure side of the system (PresSys) as a first input. As a second input the current pressure in the booster tank (PresBoost) is taken. The output is a boost-enabling (Boost En) signal. This signal enables boosting if the pressure of the booster tank is higher than that of the system pressure. This signal is also dependent on the booster tank pressure level. If the pressure is lower than a certain limit, boosting is also not allowed.

The detail of the booster tank control block is shown in Figure 3-15. The value of the pressure at the high-pressure side of the system is compared with the pressure in the booster tank by means of the relational operator block. If the pressure in the system is lower than the pressure in the booster tank, the output of the relational operator will be one. This means that boosting can be enabled.

The relay checks the pressure of the booster tank. When the pressure in the booster tank is above a predefined set point an output of one is generated, which enables boosting. When the pressure is below a predefined set point the relay will give out a zero value. Boosting is then disabled.

The outputs of the relational operator and the relay is multiplied so that the boosting enabling signal (Boost En) is dependant on the pressure difference between the system and the booster tank, and also on the pressure in the booster tank itself.

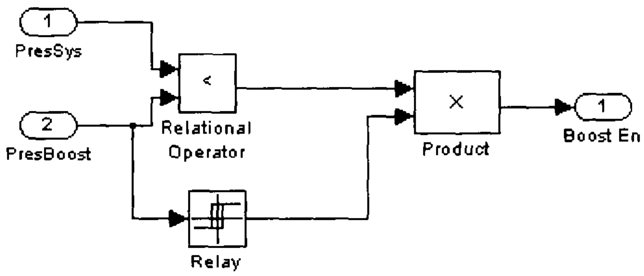


Figure 3-15 Detail of the booster tank control component

3.6.6.2 Booster tank pressure control component

The pressure in the booster tank cannot be allowed to become more than a predefined maximum pressure. The inflow of helium into the booster tank must therefore be controlled to ensure the booster tank is not over pressurised. When the booster tank reaches its maximum pressure set point, the flow to the booster tank has to be disabled.

A detailed layout of the booster tank pressure control component is given in Figure 3-16.

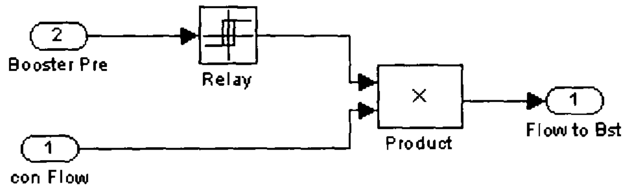


Figure 3-16 Detail of the booster tank pressure control component

The pressure in the booster tank (Booster Pre) is checked by the relay. If the pressure reaches the maximum set point the relay will give out a zero value. This value is multiplied with the helium flow to the booster tank (con Flow). The resulting flow to the booster tank (Flow to Bst) is then zero. When the maximum set point is not reached the output value of the relay is one and helium flow to the booster tank is enabled.

3.6.6.3 Booster tank component

The booster tank component is the block that models the dynamics of the booster tank. A detailed layout of the booster tank component is given in Figure 3-17.

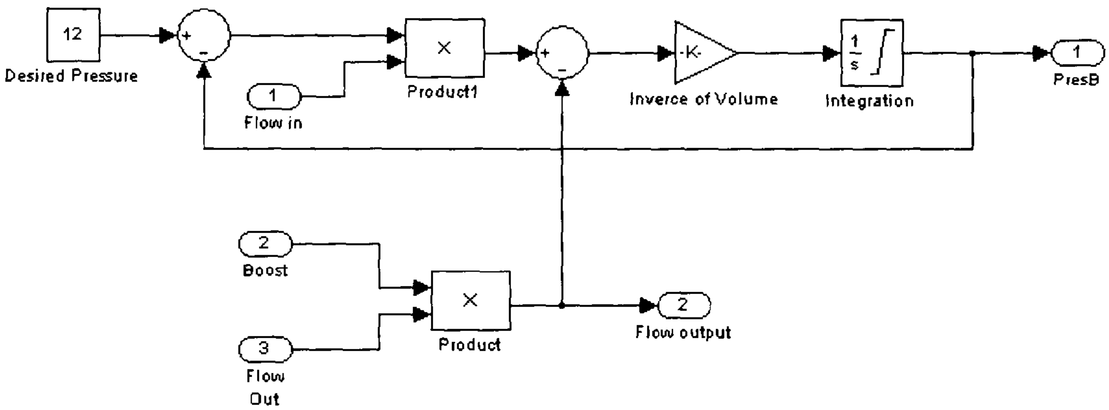


Figure 3-17 Detail of the booster tank component

A constant block specifies the desired pressure in the booster tank. The actual pressure in the booster tank is subtracted from the desired pressure to obtain the pressure error in the booster tank. The error is multiplied by the amount of helium flowing into the booster tank (Flow in). From this value the amount of helium leaving the booster tank (Flow output) is subtracted. Next (3-1) is applied to obtain the actual pressure in the booster tank.

Input (3) in Figure 3-17 represents the amount of helium flowing out of the booster tank when boosting is enabled. Input (2) states whether the booster tank is capable of boosting or not and has a value of one and zero respectively.

3.6.6.4 Booster tank simulation

When boosting is not enabled the booster tank will be pressurised up to its maximum pressure value. The helium flow to the booster tank will then be disabled. Figure 3-18 shows how the pressure in the booster tank increases and settles at its desired pressure level. The initial pressure value of the booster tank is assumed to be 10 bars and the desired pressure value is 12 bars.

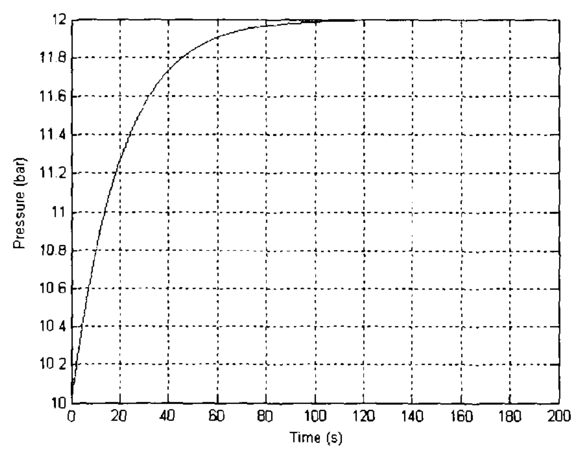


Figure 3-18 Pressure in the booster tank

In the next case the booster tank is enabled. In Figure 3-19 the flow out of the booster tank is shown. This is the amount of helium injected into the system at the high-pressure side of the system.

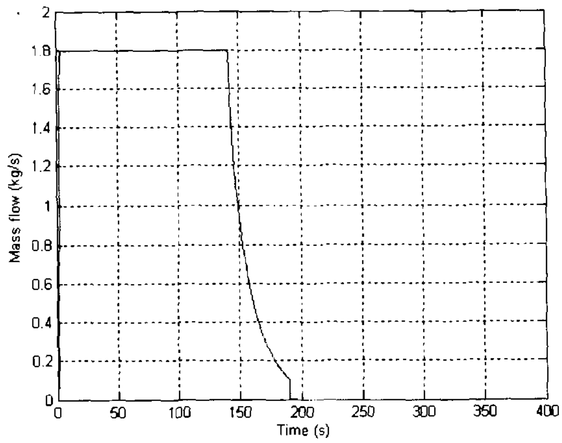


Figure 3-19 Helium flow out of booster tank

Figure 3-20 shows the pressure in the booster tank during boosting. When boosting is finished the pressure in the booster tank increases until its desired pressure level is reached.

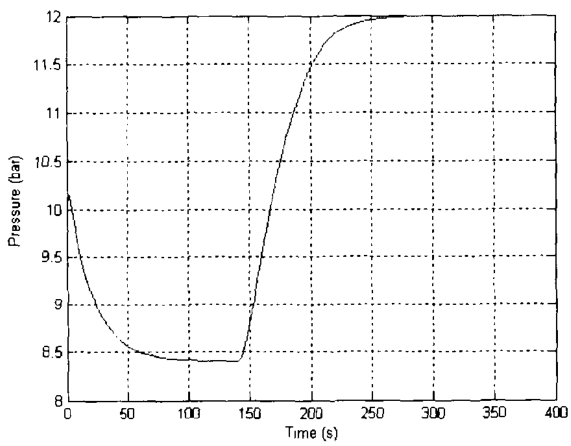


Figure 3-20 Pressure in booster tank

3.7 Conclusion

In this chapter the basic operation of a Brayton cycle based power plant was discussed. Using electrical equivalents for mechanical and thermodynamic parameters a linear model for this power plant was constructed. This linear model was then converted to a Simulink® model making it possible to simulate this model on a computer. Simulink®

provides a graphical user interface for building models as block diagrams, using click-and-drag operations.

The four actuators that manipulate the power output of the power plant at normal load following conditions were simulated to determine their effect on the power output. A booster tank model was added to the linear model. The booster tank makes high-pressure injection possible and allows an instant increase in the power when needed. This model of the power plant is now complete and will be used as a simulation platform. It will be very useful to evaluate the performance of different controllers.

Chapter 4

System Control

4.1 Introduction

The aim of this chapter is to investigate two control strategies for the power control of a three-shaft Brayton cycle based power plant. The first control technique that will be investigated is PID control. This type of control is well known and very widely used by engineers. The second control technique can be classified as an intelligent control technique. Fuzzy control is part of the field known as Artificial Intelligence (AI). This controller incorporates human reasoning strategies to generate control outputs.

The linear model of the power plant derived in the previous chapter will be used as a simulation platform to test these two control strategies. The results will be used to determine which control strategy is more suited for this problem. It is also the aim of this chapter to introduce the reader to Fuzzy control, which is a very interesting and promising control technique.

4.2 PID control

As previously discussed the power output of the system is manipulated by means of four actuators: the gas bypass valve, low-pressure injection, high-pressure extraction and boosting. It was therefore decided to use four PID controllers to control the power output. Each PID controller generates a control output for a specific actuator. This proposed PID control strategy is shown in Figure 4-1.

The actual power output of the system (P) is subtracted from the desired power reference (P_{ref}) to obtain the power error (e_p). The power error is the input to the PID controller numbered 1. This PID controller generates the bypass valve set point (*BPV-sp*).

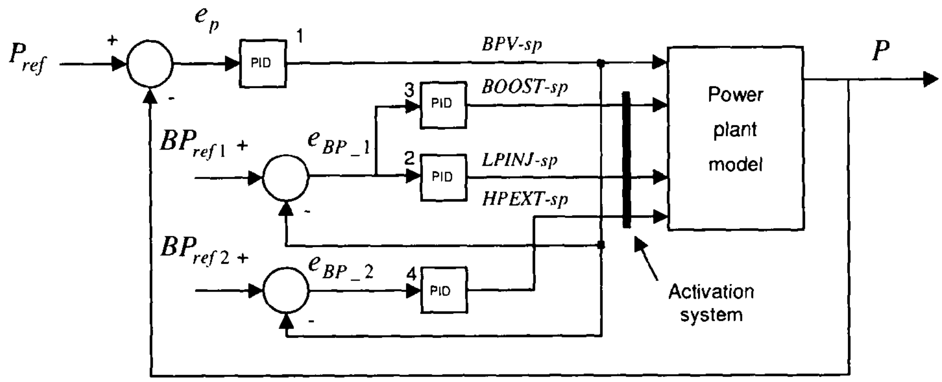


Figure 4-1 Conceptual diagram of the PID control strategy

The *BPV-sp* is subtracted from the first bypass valve reference (BP_{ref1}) to obtain a bypass valve error (e_{BP_1}). This bypass valve error is the input to two other PID controllers numbered 2 and 3 that generate the set point values (*LPINJ-sp, BOOST-sp*) for low-pressure injection and boosting respectively. The activation system allows the *LPINJ-sp* and *BOOST-sp* to be ported to the system only when $e_p > 0$ MW and $e_p > 5$ MW respectively.

The *BPV-sp* is also subtracted from a second bypass valve reference (BP_{ref2}) to obtain a second bypass valve error (e_{BP_2}). This bypass valve error is passed to the PID controller numbered 4 that generates the set point value (*HPEXT-sp*) for high-pressure extraction. The *HPEXT-sp* is only ported to the system when $e_p < 0$ MW.

The bypass valve is mainly used to control the power output of the system, but it must operate between predefined boundaries for the system to have a certain amount of reserve capacity. Therefore, two bypass valve references BP_{ref1} and BP_{ref2} are defined to ensure that the bypass valve operates between predefined boundaries.

When the power output is increased and the bypass valve starts to operate outside its boundary value there will be a bypass valve error (e_{BP_1}). The set point values for the actuators, boosting and low-pressure injection, will be generated according to e_{BP_1}. This will restore the bypass valve to operate within its boundary values.

In the case where the power output is decreased and the bypass valve again starts to operate outside its boundary value there will be a bypass valve error (e_{BP_2}). The set point value for the actuator, high-pressure extraction, will be generated according to e_{BP_2}. This will restore the bypass valve to operate within its boundary values.

4.2.1 PID control and logic block

A PID control and logic block was set up in Simulink® (Figure 4-2) that contains the PID controllers for the four actuators that manipulate the helium inventory of the system. It also contains other logic that facilitates the control process, ensuring the desired responses are obtained.

In Figure 4-2 the inputs and outputs of the control block are shown. The block's inputs are the power error (Error P), the desired power output (Pref), the output power (P), the initial power error (Init P Error), the pressure difference between the booster tank and the high-pressure side of the power plant (Press Diff). The initial power error is the power difference at the time zero of the simulation. The outputs of the block are the set points to the four helium inventory actuators: bypass valve set point (BPV-sp), low-pressure injection set point (LPINJ-sp), high-pressure extraction set point (HPEXT-sp) and the boost set point (BOOST-sp).

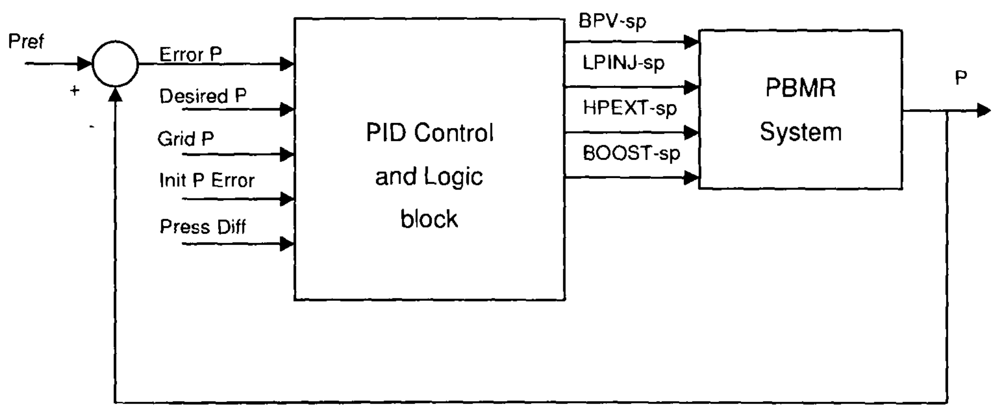


Figure 4-2 Conceptual diagram of the PID control and logic block

The PID control and logic block constitutes four subsystems. Each subsystem contains a PID controller that is responsible for generating the desired set point value. These four subsystems are shown in Figure 4-3. Each of these subsystems will be discussed individually.

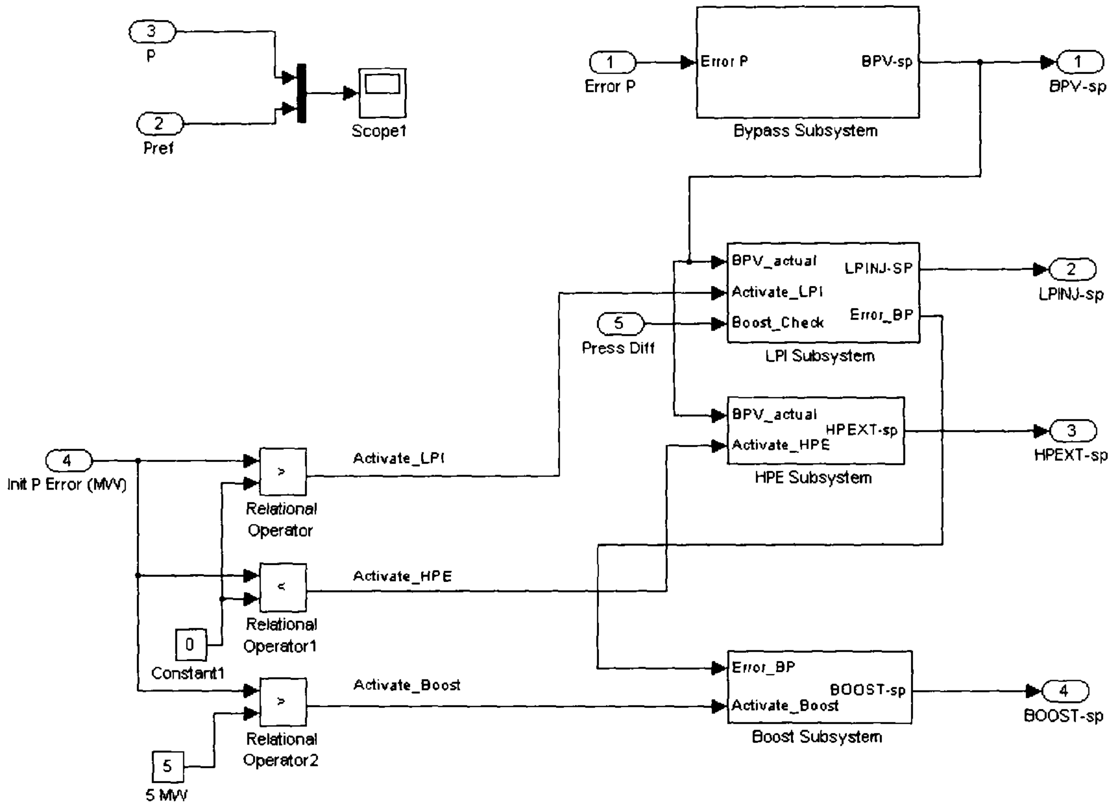


Figure 4-3 Four subsystems of the PID control and logic block

4.2.2 PID Bypass subsystem

The bypass subsystem generates the desired set point of the bypass valve according to the power error. The detail of the bypass subsystem is shown in Figure 4-4. It has one input and one output. The input is the power error (Error P) and the output is the bypass valve set point (BPV-sp).

The PID controller in Figure 4-4 takes the power error and generates a control output. A saturation block limits the output of the PID controller. The output of the saturation block is multiplied with a predefined constant. The rate of the signal must be limited to conform to system specifications. A first order low pass filter smoothes the control output signal. This signal passing through the filter is called the bypass valve set point (BPV-sp).

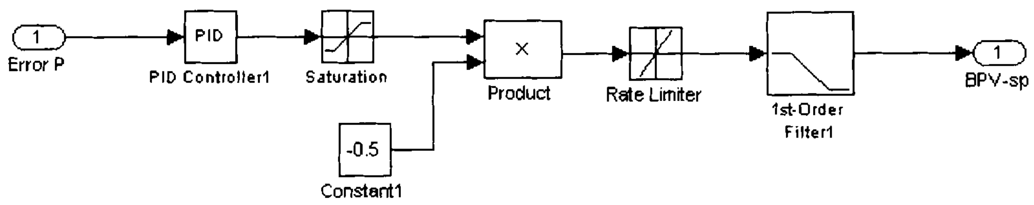


Figure 4-4 Detail of PID bypass subsystem

4.2.3 PID Low-Pressure Injection (LPI) subsystem

The LPI subsystem (See Figure 4-5) calculates the desired low-pressure injection set point (LPINJ-sp). It has three inputs and two outputs. The inputs are the mass flow through the bypass valve (BPV_actual), an activation signal (Activate_LPI) and a pressure difference signal from the booster tank. The outputs are the low-pressure injection set point (LPINJ-sp) and the bypass valve set point error (Error_BP).

The bypass valve set point depends on the pressure in the booster tank. If the pressure in the booster tank is low then the set point of the bypass valve must be higher. This allows the system to have reserve capacity in case the power needs to be increased when the booster tank is depleted. The (Boost_Check) input is used to check if the booster tank is at the right level.

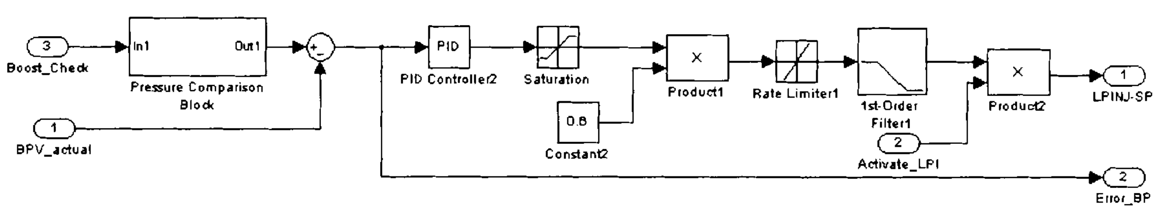


Figure 4-5 Detail of the PID LPI subsystem

The pressure comparison block calculates the correct set point for the bypass valve according to the pressure in the booster tank. The PID controller takes the bypass valve error as an input. It then calculates the LPI set point. An activation signal (Activate_LPI) is multiplied with the LPI set point. This signal has a value of one when the low-pressure injection is activated and a zero value when it is deactivated.

4.2.4 PID High-Pressure Extraction (HPE) subsystem

The HPE subsystem in Figure 4-6 calculates the desired high-pressure extraction set point. It takes as inputs the value of the bypass valve (BPV_actual) and an activation signal (Activate_HPE). The output is the high-pressure extraction set point.

High-pressure extraction decreases the power output. The bypass valve set point can then assume a lower value than in other cases to increase system efficiency. The actual value of the bypass valve (BPV_actual) is subtracted from a reference value (BPV_ref) to obtain a bypass valve error.

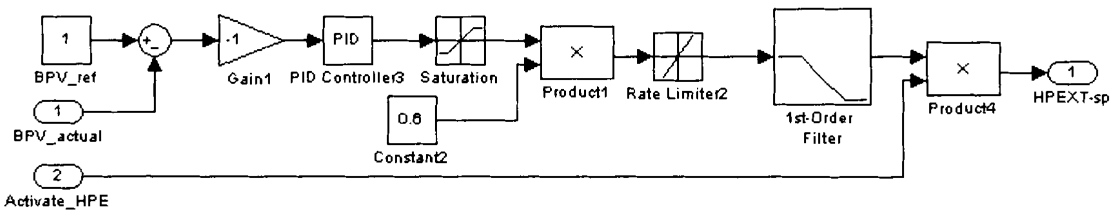


Figure 4-6 Detail of the PID HPE subsystem

The bypass valve error is multiplied by a negative value because extraction is a mirror image of injection. The PID controller then generates the high-pressure extraction set point (HPEXT-sp).

An activation signal (Activate_HPE) is multiplied with the high-pressure extraction set point. The activation signal has a value of one when high-pressure extraction is activated and a zero value when it is deactivated.

4.2.5 PID Boost subsystem

The boost subsystem in Figure 4-7 calculates the desired boost set point. It takes the bypass valve error (Error_BP) as an input as well as an activation signal (Activate_Boost) and generates the boosting set point (BOOST-sp).

The PID controller takes the bypass valve error (Error_BP) as input and calculates the boost set point (BOOST-sp). An activation signal (Activate_Boost) is multiplied with the boost set point. This signal has a value of one when boosting is activated and a zero value when it is disabled.

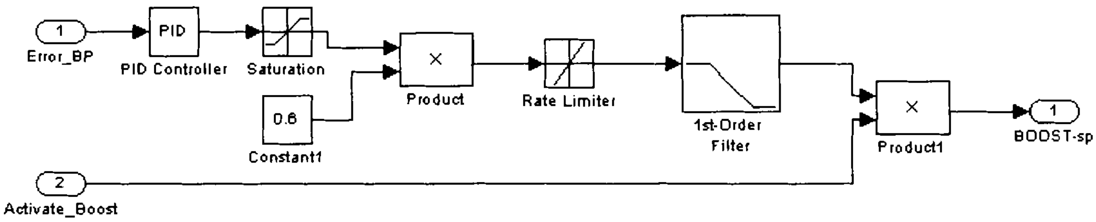


Figure 4-7 Details of the PID Boost subsystem

4.2.6 Activation signals

According to the initial power error (Init P Error) certain subsystems have to be activated (See Figure 4-3). When the initial power error is positive the low-pressure injection subsystem is activated, when the initial power error is negative the high-pressure extraction subsystem is activated. Boosting is only activated when the initial power error is positive and more than a specified value, in this case more than 5 MW.

The bypass valve subsystem is always activated and works together with the other subsystems to control the power output. This activation system ensures that there is no conflict between the outputs of the subsystems. For example it is not desirable to have the LPI subsystem inject helium and the HPE subsystem to extract helium at the same time.

4.3 Fuzzy control

The same control strategy as shown in Figure 4-1 is used in this case, but the PID controllers are substituted with Fuzzy controllers.

4.3.1 Fuzzy control and logic block

The Fuzzy control and logic Simulink® block contains the Fuzzy controllers that generate the set point values for the four helium inventory actuators. Figure 4-8 shows the inputs and outputs of the Fuzzy control and logic block.

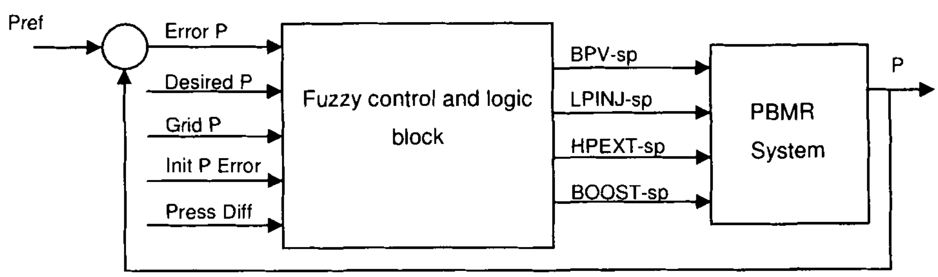


Figure 4-8 Conceptual diagram of the Fuzzy control and logic block

The detail of the Fuzzy control and logic block is identical to that of the PID control and logic block complete with its four subsystems as shown in Figure 4-3. In the following paragraphs a more detailed description of the Fuzzy controllers and the four subsystems will be given.

4.3.2 Fuzzy controller detail

It was decided to use a Fuzzy PD and I (FPD+I) controller as discussed in chapter 2. Scaled values of the error (e) and the change in error (ce) is used to generate a control output (*uF*). A scaled value of the integral of the error (ie) is added to this control output to generate the final control output (*u*) of the FPD+I controller (See Figure 4-9).

The scaling constants are defined as follows: Fuzzy proportional gain (GE), Fuzzy derivative gain (GCE), Fuzzy integral gain (GIE) and Fuzzy control gain (GU).

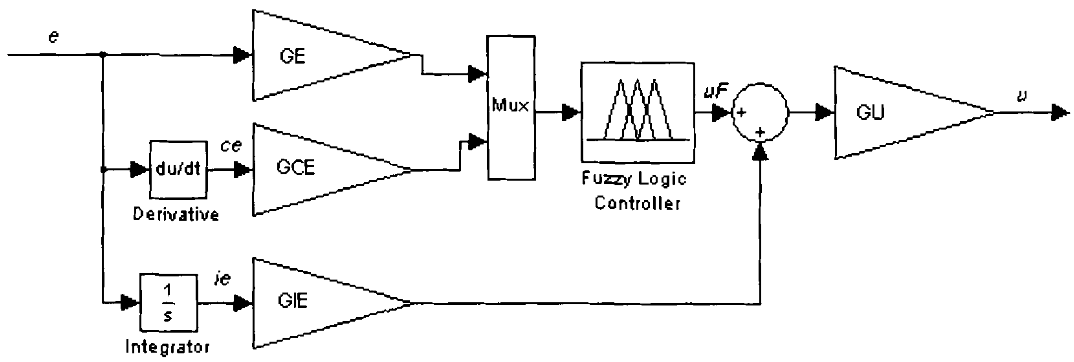


Figure 4-9 Fuzzy PD+I controller

4.3.2.1 Membership function definition

A Mamdani inference system is used. Consider a universe of discourse[−E,E]. The membership functions for the inputs and output are defined as shown in Figure 4-10 and Figure 4-11 respectively.

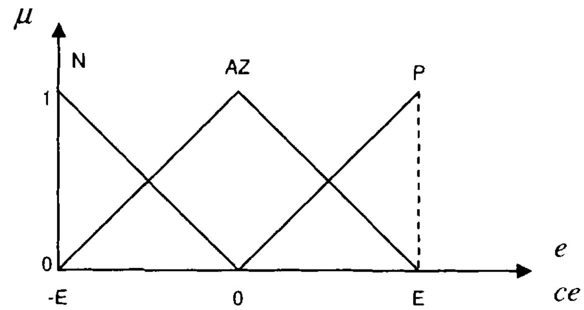


Figure 4-10 Input membership function definition

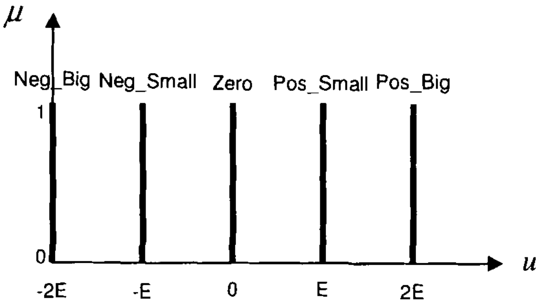


Figure 4-11 Output membership function definition

The input membership functions for both the error (e) and change of error (ce) are triangular. The input space is partitioned in three Fuzzy sets called negative (N), about zero (AZ) and positive (P).

Singleton membership functions are chosen to define the output control variable u. The output space is partitioned in five Fuzzy sets called Negative big (Neg_Big), Negative small (Neg_Small), Zero (Zero), Positive small (Pos_Small) and Positive big (Pos_Big).

4.3.2.2 Rule base definition

The rule base of the Fuzzy controllers consists of nine rules. These rules link two inputs namely the error and the change of the error to a control output. The rules are defined as follows:

R1: If (Error is Negative) and (Change of error is Negative) then (Control is Neg_Big)
R2: If (Error is Negative) and (Change of error is About Zero) then (Control is Neg_Small)
R3: If (Error is Negative) and (Change of error is Positive) then (Control is Zero)
R4: If (Error is About Zero) and (Change of error is Negative) then (Control is Neg_Small)
R5: If (Error is About Zero) and (Change of error is About Zero) then (Control is Zero)
R6: If (Error is About Zero) and (Change of error is Positive) then (Control is Pos_Small)
R7: If (Error is Positive) and (Change in error is Negative) then (Control is Zero)
R8: If (Error is Positive) and (Change in error is About Zero) then (Control is Pos_Small)
R9: If (Error is Positive) and (Change in error is Positive) then (Control is Pos_Big)

4.3.3 Fuzzy bypass subsystem

The detail of the Fuzzy bypass subsystem is given in Figure 4-12. The power error and its derivative are inputs to the Fuzzy controller. The integral of the error is added to the output of the Fuzzy controller. The output of the controller is limited to conform to system specifications.

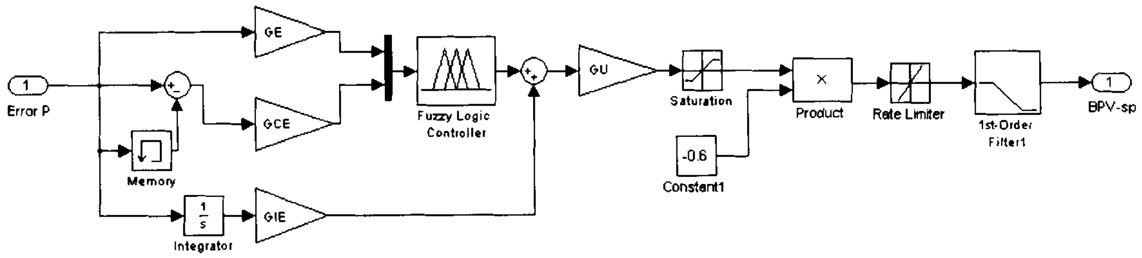


Figure 4-12 Detail of the Fuzzy bypass subsystem

4.3.4 Fuzzy low-pressure injection (FLPI) subsystem

Figure 4-13 shows the detail of the FLPI subsystem. The reference for bypass valve error is generated according to the pressure in the booster tank. Again the output is limited to conform to system specifications.

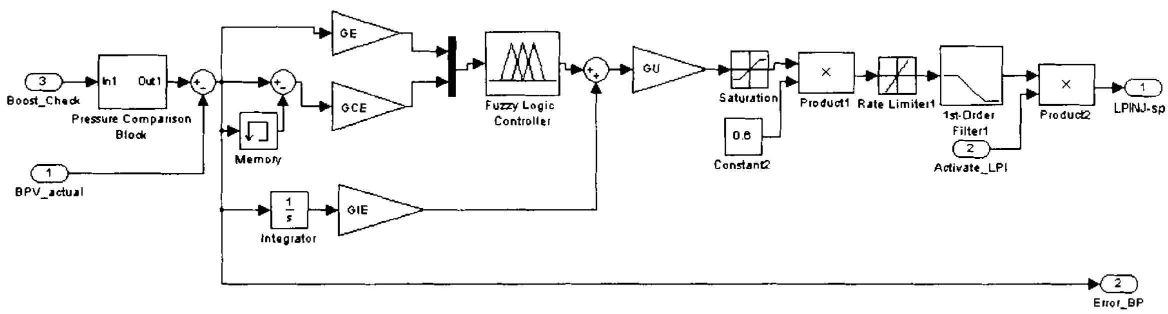


Figure 4-13 Detail of the Fuzzy low-pressure injection subsystem

4.3.5 Fuzzy high-pressure extraction (FHPE) subsystem

Figure 4-14 show the detail of the FHPE subsystem. The bypass valve reference value (BPV_ref) is subtracted from the bypass valve set point (BPV_actual) to get the second bypass valve error.

High-pressure extraction is the opposite of low-pressure injection. The bypass valve error therefore needs to be multiplied with a value of negative one. This will allow high-pressure extraction to be activated correctly.

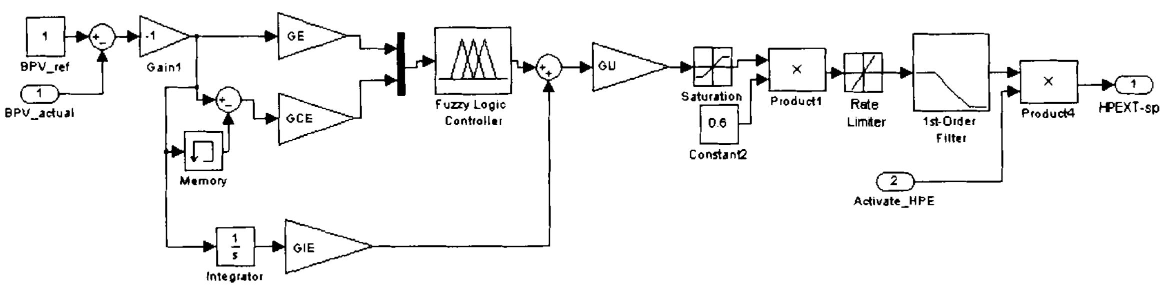


Figure 4-14 Detail of the Fuzzy high-pressure extraction subsystem

4.3.6 Fuzzy boost subsystem

The Fuzzy boost subsystem is shown in Figure 4-15. The bypass valve error, derivative of the error and the integral of the error is used to generate the boost set point.

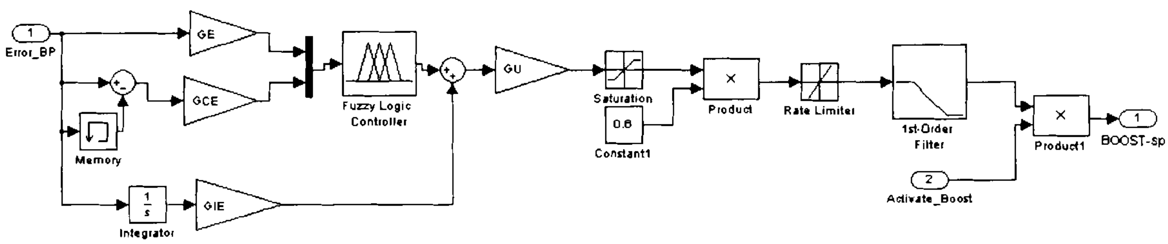


Figure 4-15 Detail of the Fuzzy boost subsystem

4.4 Determination of control constants

Due to the complexity of the system the values of the controller constants for both the PID controllers and FPD+I controllers are chosen by trial and error. The constant values for each of the controllers are summarised in tables 4-1 and 4-2.

PID controller no.	*P-gain*	*I-gain*	*D-gain*
1 (BVP-sp)	4	0	1.5
2 (LPINJ-sp)	3	0	1.5
3 (HPEXT-sp)	3	0	1.5
4 (BOOST-sp)	2	0	1.5

Table 4-1 Constant values for the PID controllers

The PID and FPD+I constant relations (2-5, 2-6 and 2-7) in chapter 2 were used at first to determine the constant values for the FPD+I controllers from the existing PID controllers. It was not successful in this case and it was therefore decided to choose the constant values of the FPD+I by means of trial and error.

FPD+I controller no	GE	GCE	GIE	GU
1 (BVP-sp)	0.2	5	0	0.2
2 (LPINJ-sp)	10	8	0	0.2
3 (HPEXT-sp)	5	3	0	0.2
4 (BOOST-sp)	40	80	0	0.1

Table 4-2 Constant values for the FPD+I controllers

The values for the integral constants are chosen to be zero. It was shown by simulation that proportional and derivative control was adequate to control the system satisfactorily. By keeping the gain constants the same and assigning the integral gains a value of 0.1 the power output of the system for a reference of 102 MW is simulated. In Figure 4-16 it is clear that the added integral control results in large overshoots, which is undesirable.

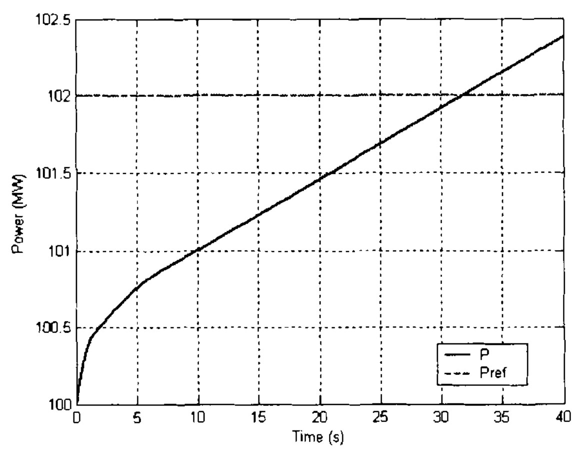


Figure 4-16 Power output of PID control with I-gains = 0.1

4.5 Results

Both the PID and Fuzzy control strategies are implemented in the Simulink® environment. The Simulink® environment will now be used as a test platform for both strategies. Responses generated from each control strategy will be investigated and compared.

Certain scenarios are generated specifically to investigate the role of each actuator in generating the desired power level. To be able to investigate all the actuators, three scenarios are needed: a small power increase scenario, a large power increase scenario and a power decrease scenario. The role of the bypass valve and low-pressure injection are investigated in the first scenario. In a large power increase scenario the role of boosting can be investigated. Lastly in the power decrease scenario, high-pressure extraction is investigated. These three scenarios are simulated for each control strategy and summarised in Figures 4-17, 4-18 and 4-19 respectively.

The power response for each control strategy is compared by means of the ITAE performance index as given by (4-1)

$$ITAE = \int_0^T t\left|e(t)\right|dt \tag{4-1}$$

where T is the total simulation period and $e(t)$ is the power error at time step t. It was decided to normalise the power error because the power error can have a very large number. Equation (4-1) then becomes the following:

$$ITAE = \int_0^T t\left|\frac{e(t)}{100\times e^6}\right|dt \tag{4-2}$$

The performance for each control strategy and scenario is measured. The strategy with the lowest value of the performance index has the best performance.

In the first scenario (Figure 4-17) the power level is increased from 100 MW to 102 MW. For the power to increase the mass flow through the bypass valve needs to decrease; therefore the set point of the bypass valve shows negative mass flow perturbations. Positive perturbations of helium mass flow for low-pressure injection results in helium being injected at the low-pressure side of the system, also contributing to the increase in power.

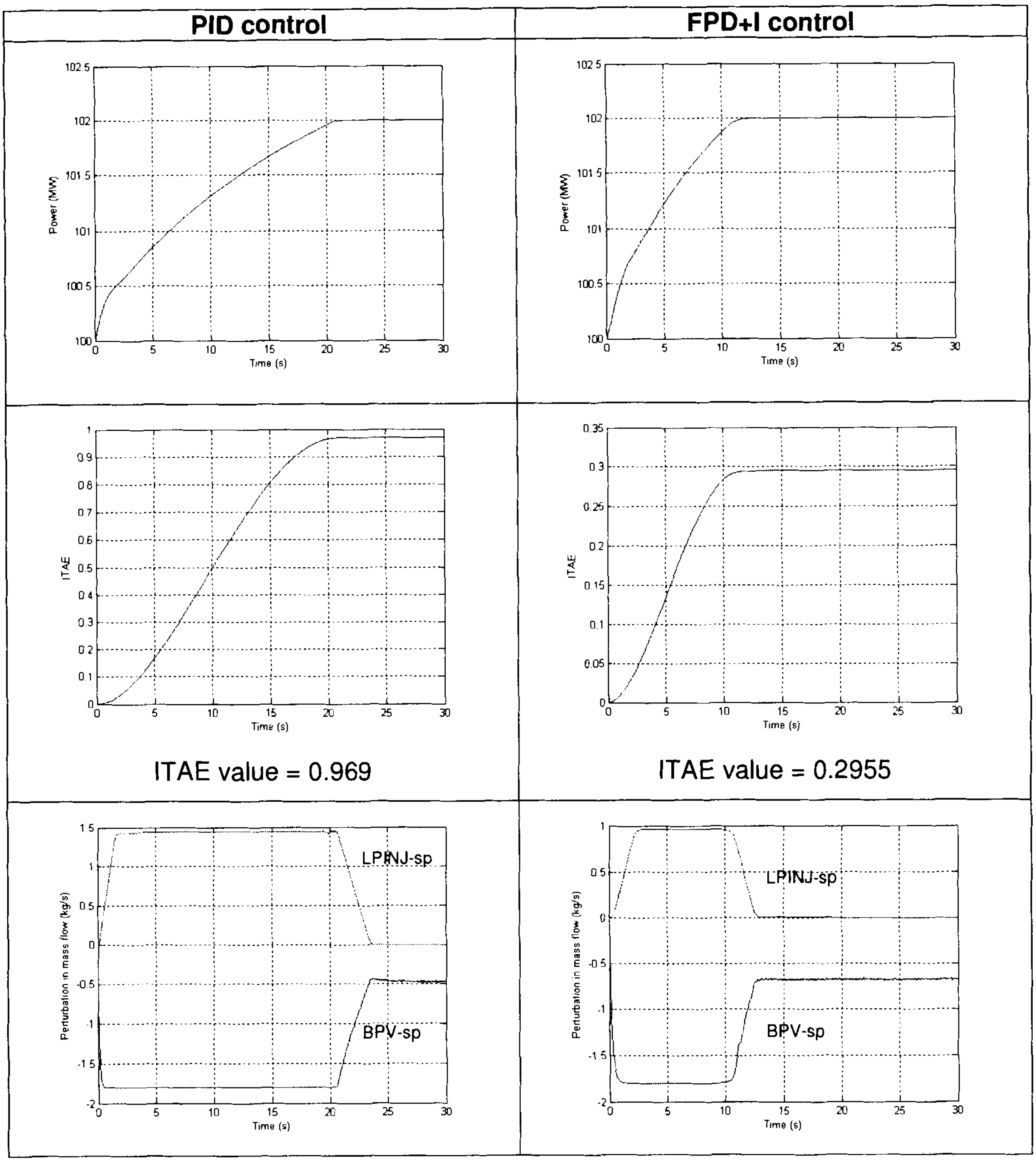


Figure 4-17 Results of a small power increase

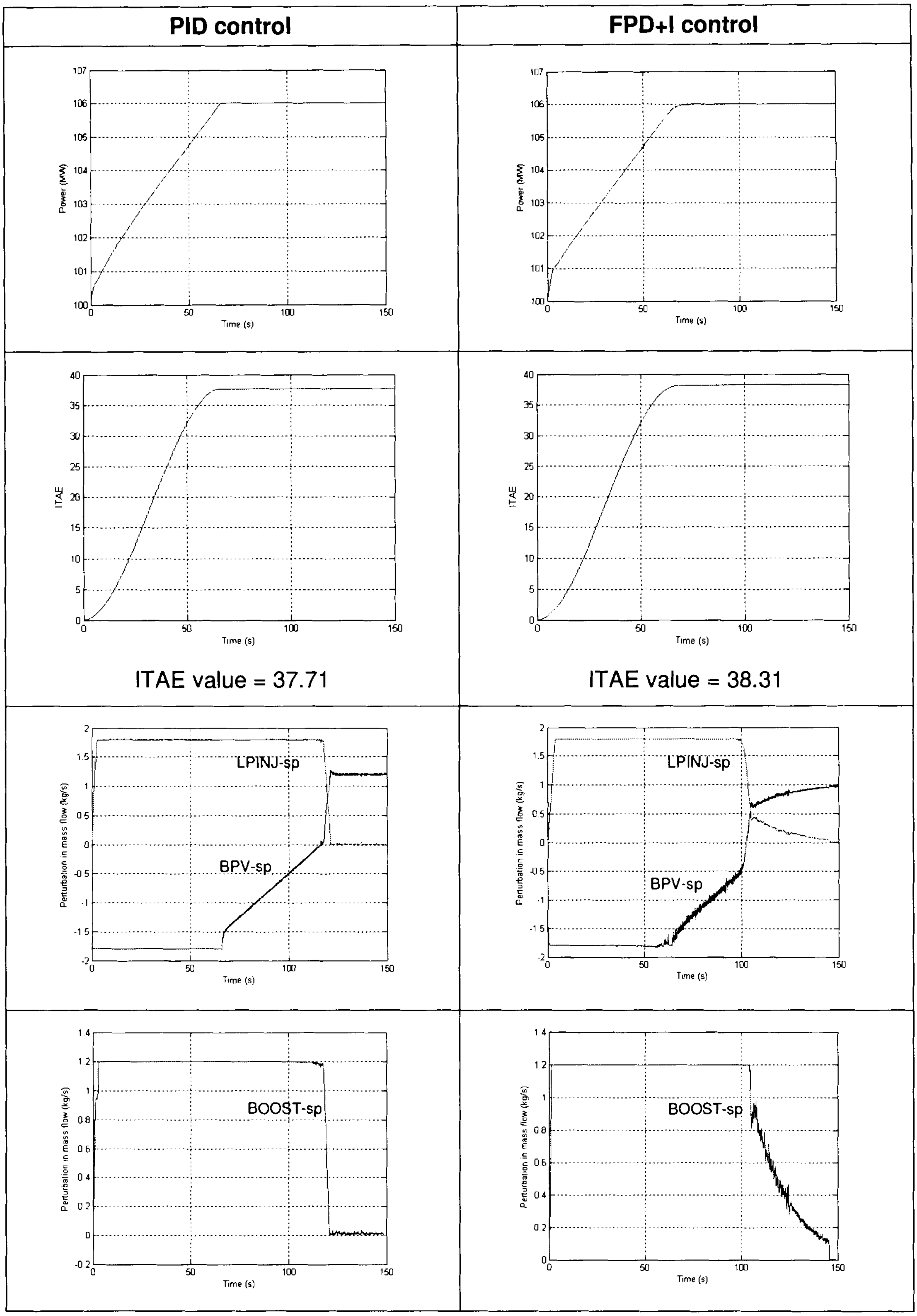


Figure 4-18 Results for a large power increase

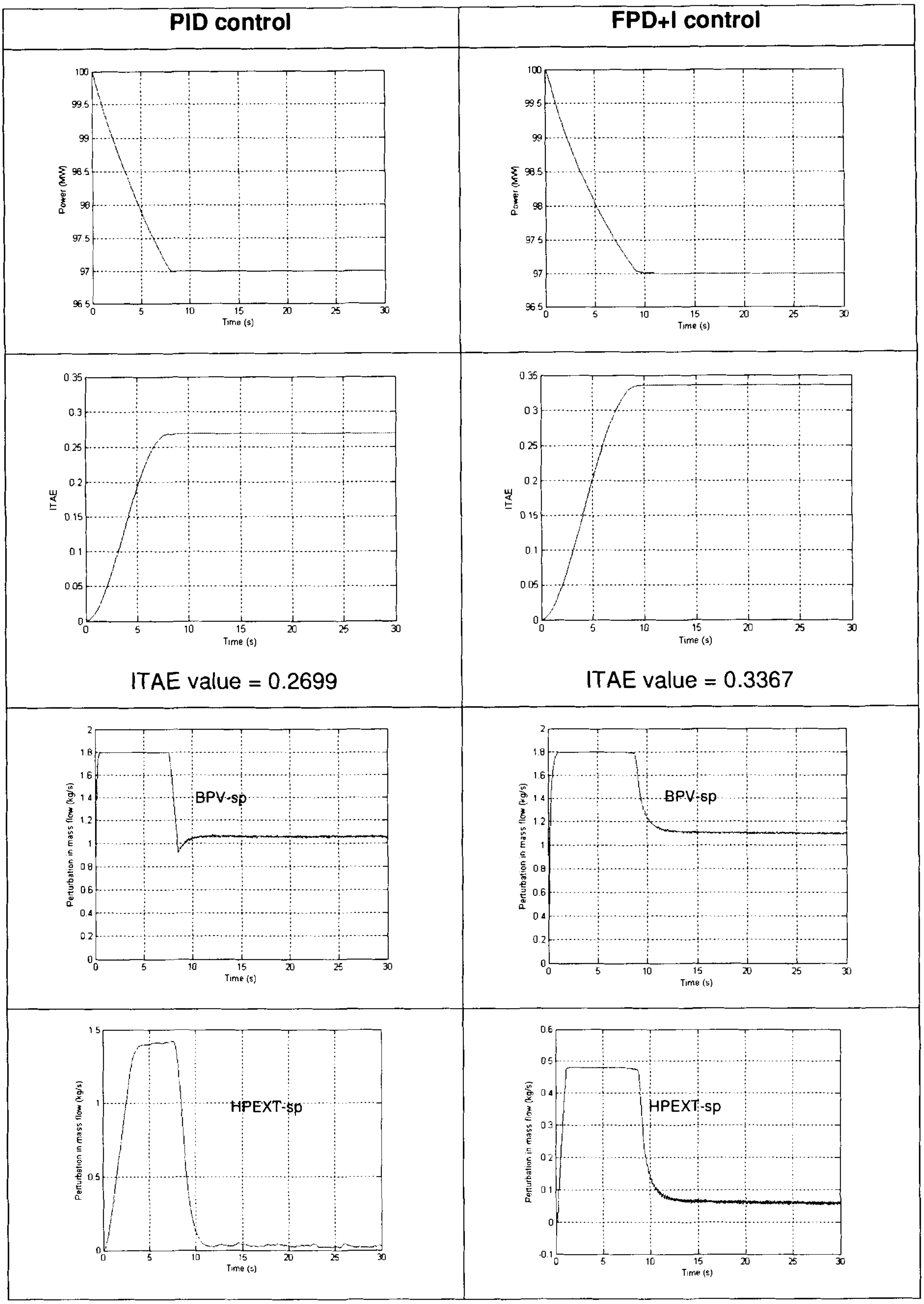


Figure 4-19 Results for a power decrease

In the second scenario the power level is increased from 100 MW to106 MW (See Figure 4-18). This large increase in the power results in the utilisation of the booster tank to inject a limited amount of helium at the high-pressure side of the system. Boosting allows an immediate increase in the power output. The boosting set point shows positive mass flow perturbations which simulates helium injection at the high-pressure side.

The power level is decreased in the third scenario (Figure 4-19) from 100 MW to 97 MW. For the power to decrease the mass flow through the bypass valve needs to increase; therefore the set point of the bypass valve shows positive mass flow perturbations. Positive perturbations of helium mass flow for high-pressure extraction results in helium being extracted at the high-pressure side of the system, also contributing to the decrease in power.

4.6 Conclusion

In this chapter two control strategies were tested on a linear model of the original PBMR. The PID control strategy performed well according to the ITAE performance index. The Fuzzy control strategy compared extremely well with the PID control strategy. It proved to show the same or even better responses (See Figure 4-17).

In the next chapter the Fuzzy control strategy will be optimised by means of a random search technique. A Genetic Algorithm will be used to adapt the gains of the four FPD+I controllers according to a cost function.

Chapter 5

Fuzzy control optimisation

5.1 Introduction

The performance of the Fuzzy control strategy can be improved by adapting the gain values of each FPD+I controller. A Genetic Algorithm (GA) is an effective parameter optimisation technique. The GA will adapt all four gains of each subsystem simultaneously (that is, 16 gains will be adapted at the same time) according to the value of a cost function, also called an objective function (See Figure 5-1). In this case the objective function will use the power error e_p to generate an objective value.

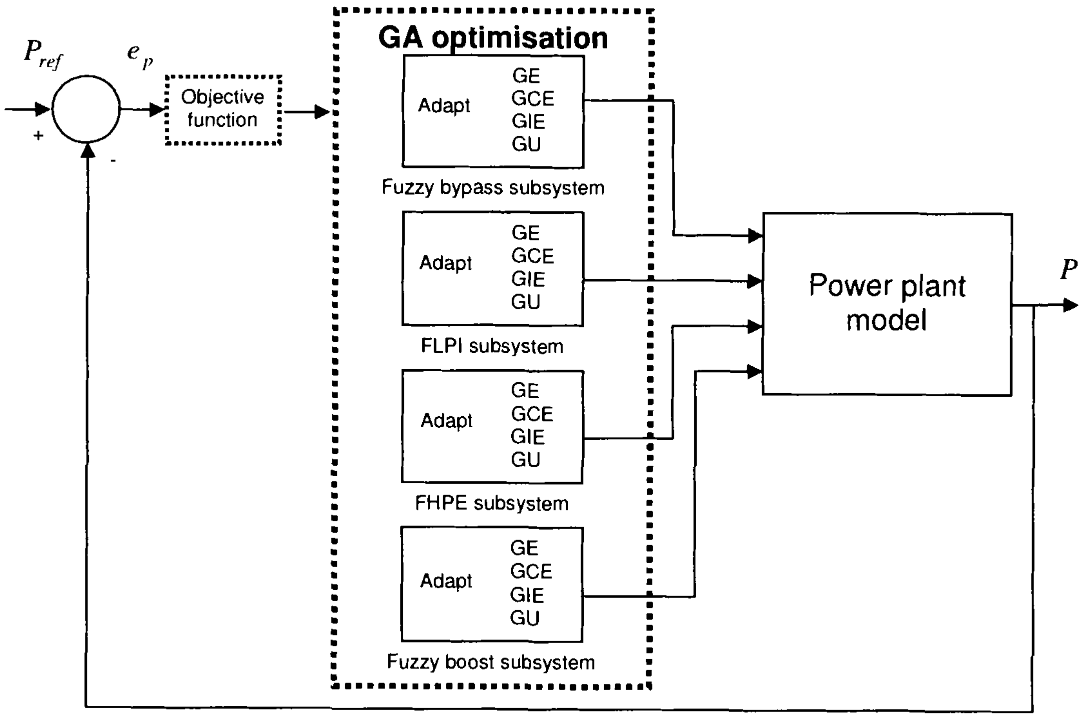


Figure 5-1 GA optimisation of the four Fuzzy subsystems

In the following paragraphs more detail on the specific GA used for the gain optimisation will be given.

5.2 Description and outline of the GA

As previously discussed the GA is a random search method that mimics natural biological evolution. The GA operates on a population of potential solutions of a problem and then applies the principle of the survival of the fittest to produce better solutions. At each generation a new set of solutions is generated according to the fitness of each individual (solution). These more fit solutions are used to breed the next generation of possible better solutions.

The GA is outlined in pseudo-code format in Figure 5-2. The population at time t is represented by the time-dependant variable P, with the initial population of random estimates being $P(0)$. The population is then evaluated by means of an objective function. Next the best individuals are selected and used to generate offspring. These offspring are inserted into the population and then the whole population is evaluated again. This process iterates for a number of generations.

```
Procedure GA
begin
        t=0;
        initialise P(t);
        evaluate P(t);
        while not finished do

        begin
                t = t + 1;
                select P(t) from P(t-1);
                reproduce pairs in P(t);
                evaluate P(t);
        end
end
```

Figure 5-2 GA outline [1]

5.3 Initialisation of the population

The GA simultaneously operates on a number of potential solutions, called a population. The population usually consists of some encoding of the parameter set. Typically, a population is composed of between 30 and 100 individuals [10]. The most commonly used representation of chromosomes in the GA is that of the single-level binary string. The GA used to optimise the FPD+I controller uses real-valued genes instead of a binary encoding. Using real-valued genes has a number of advantages. The efficiency of the GA is increased as there is no need to convert chromosomes to phenotypes before each function evaluation; less memory is required as efficient floating-point internal computer representations can be used directly; there is no loss in precision by discretisation to binary or other values; and there is a greater freedom to use different genetic operators [10].

Consider the four Fuzzy subsystems presented in Figure 5-1. As discussed all four gain values of each subsystem will simultaneously be optimised by the GA. The first step is to set up an initial population. Let N_{ind} be the number of individuals in the population and L_{ind} the number of parameters that needs to be optimised. For the case in Figure 5-1 the number of individuals used can vary but the number of parameters is fixed at sixteen. The initial population is therefore a $N_{ind} \times L_{ind}$ matrix shown in Table 5-1.

Individuals	Fuzzy subsystem 1				Fuzzy subsystem 2					Fuzzy subsystem 4			
	GE	GCE	GIE	GU	GE	GCE	GIE	GU		GE	GCE	GIE	GU
1	40.23	17.7	28.95	15.38	20.45	33.61	10.2	15.56		20.45	33.61	10.2	15.56
	82.06	13.26	13.35	0.09	70.21	8.01	11.61	0.02	----→	70.21	8.01	11.61	0.02
	52.43	25.64	15.20	2.54	30.12	11.3	14.71	0.62		30.12	11.3	14.71	0.62
	47.5	49.10	9.09	10.65	80.33	22.27	1.56	4.32		80.33	22.27	1.56	4.32
	90.50	13.46	25.63	0.89	9.13	2.98	20.88	17.77		9.13	2.98	20.88	17.77
	47.21	25.29	7.89	10.48	43.2	15.21	8.63	18.23		43.2	15.21	8.63	18.23
$N_{ind}-1$	90.50	13.46	25.63	0.89	9.13	2.98	20.88	17.77	----→	20.45	33.61	10.2	15.56
N_{ind}	52.43	25.64	15.20	2.54	30.12	11.3	14.71	0.62		43.2	15.21	8.63	18.23

Parameters 1 ←--→ L_{ind}

Table 5-1 Initial population

The parameter values for the initial population are randomly generated. Each parameter has an upper and lower boundary value. The random values are normally distributed between these boundary values.

5.4 Objective and fitness functions

The objective function is used to provide a measure of how well individuals have performed in the problem domain. In the case of a minimisation problem, the fittest individuals will have the lowest numerical value of the associated objective function. This raw measure of fitness is usually only used as an intermediate stage in determining the relative performance of individuals in a GA. Another function, the fitness function, is normally used to transform the objective function value into a measure of relative fitness.

For this specific application it was decided to use the ITAE performance index as an objective function. The ITAE index is defined in (4-1). Many fitness functions allow some individuals to generate an excessive number of offspring which result in a premature convergence. A way to solve the problem is to assign fitness to individuals according to their rank in the population rather than their raw performance [10].

A variable, MAX, is used to determine the bias, or selective pressure, towards the fittest individuals and the fitness of the others is determined by the following rules:

- $MIN = 2.0 - MAX$ **(5-1)**
- $INC = 2.0 \times (MAX - 1.0) / N_{ind}$ **(5-2)**
- $LOW = INC / 2.0$ **(5-3)**

where MIN is the lower bound, INC is the difference between the fitness of adjacent individual and LOW is the expected number of trials (number of times selected) of the least fit individual. MAX is typically chosen in the interval $[1.1, 2.0]$. Hence, for a population size of $N_{ind} = 40$ and $MAX = 1.1$, we obtain $MIN = 0.9$, $INC = 0.05$ and $LOW = 0.025$. The fitness of individuals in the population may also be calculated directly as in (5-4)

$$F(x_i) = 2 - MAX + 2(MAX - 1)\frac{x_i - 1}{N_{ind} - 1} \qquad \textbf{(5-4)}$$

where x_i is the position in the ordered population of individuals i [10].

To illustrate the ranking fitness function, consider the following population with 10 individuals. The current objective values are:

$$ObjV = [1;\ 2;\ 3;\ 4;\ 5;\ 10;\ 9;\ 8;\ 7;\ 6] \tag{5-5}$$

The ranking fitness function is applied to the objective values given in (5-5) and the selective pressure is equal to 2. The resulting values of the fitness function are as follows:

$$FitnV = [2.00;\ 1.77;\ 1.55;\ 1.33;\ 1.11;\ 0;\ 0.22;\ 0.44; 0.66; 0.88] \tag{5-6}$$

5.5 Selection

Selection is the process of determining the number of times, or trials, a particular individual is chosen for reproduction; thus the number of offspring that an individual will produce.

For this particular case a stochastic universal sampling (SUS) algorithm [10] is used. Instead of the single selection pointer employed in roulette wheel selection methods, SUS uses N equally spaced pointers, where N is the number of selections required. The population is shuffled randomly and a single random number in the range $[0 \quad Sum/N]$ is generated called ptr. Sum is the cumulative sum of the fitness value vector $FitnV$. The N individuals are then chosen by generating the N pointers spaced by 1, $[ptr, ptr+1, \ldots, ptr+N-1]$, and selecting the individuals whose fitness span the positions of the pointers.

5.6 Recombination

The basic operator for producing new chromosomes in the GA is that of crossover. Like its counterpart in nature, crossover produces new individuals that have some parts of both parents' genetic material.

Real-valued chromosomes are used in this application. Therefore a special crossover technique must be used. This crossover technique is also called discrete recombination [10] and is explained by means of an example.

Consider the following population with five real-valued individuals:

$$Oldpop = \begin{array}{c} \begin{bmatrix} 40.23 & 17.17 & 28.95 & 15.38 \\ 82.06 & 13.26 & 13.35 & 9.09 \\ 52.43 & 25.64 & 15.20 & 2.54 \\ 47.50 & 49.10 & 9.09 & 10.65 \\ 90.50 & 13.46 & 25.63 & 0.89 \end{bmatrix} \end{array} \begin{array}{l} \text{parent 1} \\ \text{parent 2} \\ \text{parent 3} \\ \text{parent 4} \\ \text{parent 5} \end{array} \tag{5-7}$$

An internal mask determining which parents contribute which variables to the offspring is generated and is shown in (5-8).

$$Mask = \begin{bmatrix} 1 & 2 & 1 & 2 \\ 2 & 2 & 1 & 1 \\ 2 & 1 & 2 & 1 \\ 1 & 1 & 2 & 2 \end{bmatrix} \begin{array}{l} \text{for producing offspring 1} \\ \text{for producing offspring 2} \\ \text{for producing offspring 3} \\ \text{for producing offspring 4} \end{array} \tag{5-8}$$

After recombination the new chromosome would become:

$$Newpop = \begin{bmatrix} 40.23 & 13.26 & 28.95 & 9.09 \\ 82.06 & 13.26 & 28.95 & 15.38 \\ 47.50 & 25.64 & 9.09 & 2.54 \\ 52.43 & 25.64 & 9.09 & 10.65 \end{bmatrix} \begin{array}{l} \text{Mask(1,:) parent 1 \& 2} \\ \text{Mask(2,:) parent 1 \& 2} \\ \text{Mask(3,:) parent 3 \& 4} \\ \text{Mask(4,:) parent 3 \& 4} \end{array} \tag{5-9}$$

Consider the first two rows of (5-7) and (5-8). A value of one in the first row of (5-7) means the first parent retains its parameter value, however when there is a value two in the first row of (5-7), it means the parameter value of the second parent is copied over that of the first parent. The same process is followed with the second pair of parents.

As the number of individuals in the parent population, *Oldpop* was odd, the last individual is appended without recombination to *Newpop* and the offspring returned is as follows:

$$NewChrom = \begin{bmatrix} 40.23 & 13.26 & 28.95 & 9.09 \\ 82.06 & 13.26 & 28.95 & 15.38 \\ 47.50 & 25.64 & 9.09 & 2.54 \\ 52.43 & 25.64 & 9.09 & 10.65 \\ 90.50 & 13.46 & 25.63 & 0.89 \end{bmatrix} \qquad \textbf{(5-10)}$$

5.7 Mutation

In natural evolution, mutation is a random process where one allele of a gene is replaced by another to produce a new genetic structure. In GAs, mutation is randomly applied with low probability, typically in the range 0.001 and 0.01, and modifies elements in the chromosomes. Usually considered as a background operator, the role of mutation is often seen as providing a guarantee that the probability of searching any given string will never be zero and acting as a safety net to recover good genetic material that may be lost through the action of selection and crossover.

Mutation for the real-valued chromosomes is achieved by either perturbing the gene values or randomly selecting new values within the allowed range. The advantage of real-value mutation is that higher mutation rates can be used than in binary codings, increasing the level of possible exploration of the search space without adversely affecting the convergence characteristics [10].

5.8 Re-insertion

Once a new population has been produced by selection and recombination of individuals from the old population, the fitness of the individuals in the new population may be determined. To maintain the size of the original population, the new individuals have to be re-inserted into the old population. Similarly, if not all the new individuals are to be used at each generation or if more offspring are generated than in the size of the old population then a reinsertion scheme must be used to determine which individuals are to exist in the new population. When selecting which members of the old population should be replaced the most apparent strategy is to replace the least fit members deterministically.

5.9 Termination of the GA

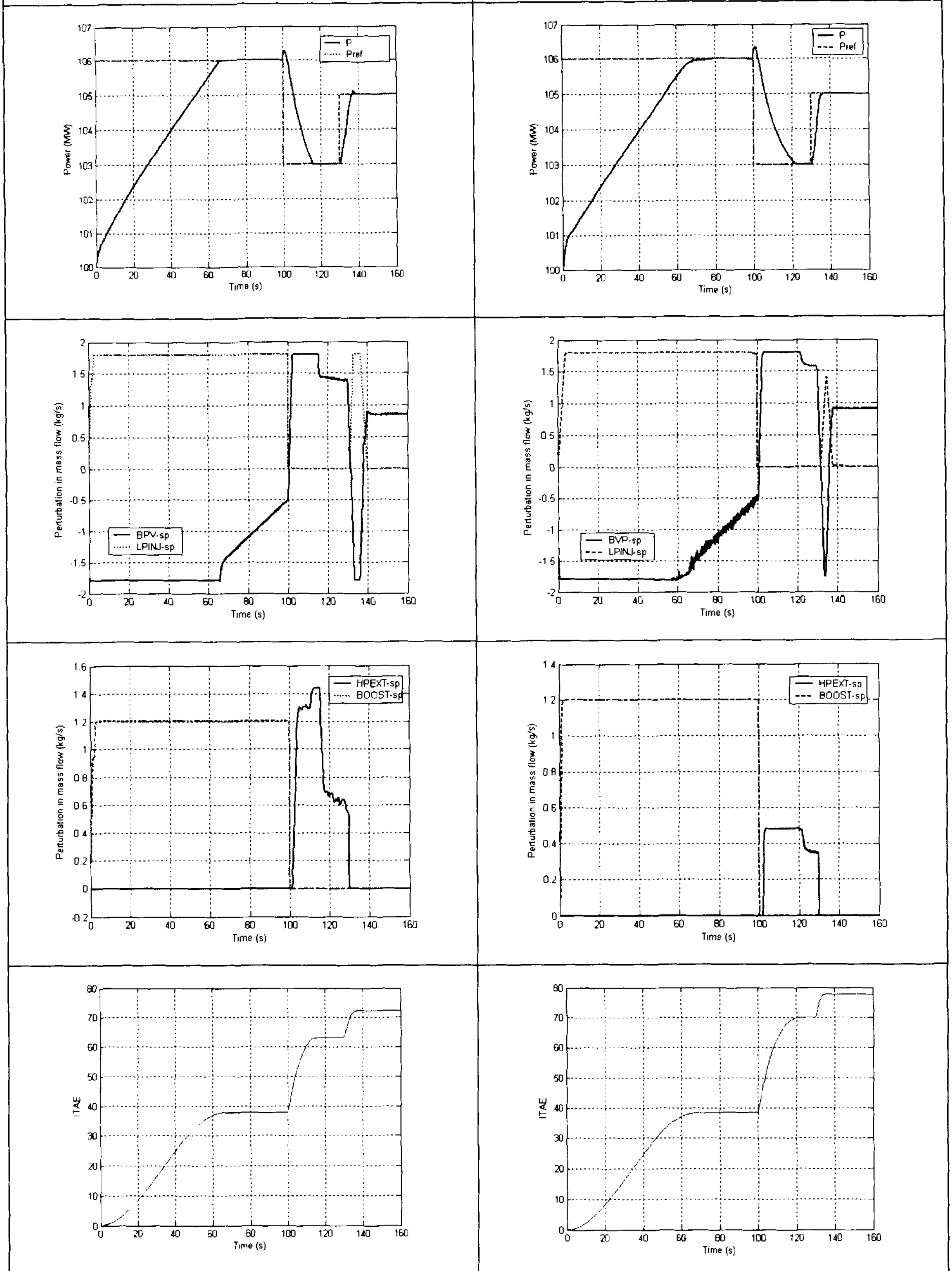
Because the GA is a stochastic search method, it is difficult to formally specify convergence criteria. As the fitness of a population may remain static for a number of generations before a superior individual is found, the application of conventional termination criteria becomes problematic. A common practice is to terminate the GA after a pre-specified number of generations and then test the quality of the best members of the population against the problem definition. If no acceptable solutions are found, the GA may be restarted or a fresh search initiated.

5.10 Fuzzy controller optimisation

The sixteen parameters of the four Fuzzy subsystems are optimised for a power set point sequence. This power reference sequence is used firstly to simulate both the PID and Fuzzy control strategies defined in chapter 4. The results are then compared with the optimised Fuzzy subsystem.

Figure 5-3 shows the results of the PID and Fuzzy control strategies. The first graphs represent the power responses, followed by the set point values of the four helium inventory actuators. Lastly the performance of both strategies is given by the ITAE performance index.

PID control	FPD+I control



ITAE value = 72.44	ITAE value =77.7

Figure 5-3 PID and Fuzzy system results

Before moving to the optimisation process of the Fuzzy control strategy certain parameters of the GA has to be defined first. The population size, generations and boundary values for the GA used are summarised in 5-2.

The boundaries are chosen in such a way that input values to the Fuzzy controller (e and ce) do not exceed their boundary values. By simulation it was determined that the control gains (GU) are in the range of 0.1 and 0.9. From chapter 4 it was determined that the integral gain (GIE) should be close to zero or zero and therefore the boundaries for this gain was chosen to be 0 and 0.01.

Fuzzy subsystem 1				
	GE	GCE	GIE	GU
Lower Boundary	0	0	0	0.1
Upper Boundary	100	100	0.01	0.9
Fuzzy subsystem 2				
	GE	GCE	GIE	GU
Lower Boundary	0	0	0	0.1
Upper Boundary	100	100	0.01	0.9
Fuzzy subsystem 3				
	GE	GCE	GIE	GU
Lower Boundary	0	0	0	0.1
Upper Boundary	100	100	0.01	0.9
Fuzzy subsystem 4				
	GE	GCE	GIE	GU
Lower Boundary	0	0	0	0.1
Upper Boundary	90	100	0.01	0.9
Populations size and number of Generations				
N_{ind}	40			
L_{ind}	16			
Number of generations	100			

Table 5-2 GA parameters

It was decided to choose the number of individuals equal to 40 to make sure there is a large amount of different individuals at the beginning of the GA. A hundred generations was chosen to allow the GA enough time to generate optimal values according to the objective function. The optimal gain values for the Fuzzy control strategy after 100 generations are summarised in 5-3.

FPD+I controller no	GE	GCE	GIE	GU
1 (BVP-sp)	14.5071	83.9731	0.00011757	0.305015
2 (LPINJ-sp)	0	0	0.00049162	0.17023
3 (HPEXT-sp)	0	61.0114	0.0072707	0.79424
4 (BOOST-sp)	19.346	62.3392	0.0052254	0.69497

Table 5-3 Optimal gain values for Fuzzy control strategy

It can be seen that the GA also chose the integral gains close to zero. This shows that proportional and derivative control is sufficient. The GA penalises low-pressure injection by giving the proportional and derivative gains a value of 0. This means that according to the objective function low-pressure injection leads to undesirable responses. Figure 5-4 shows the plot of the objective value of the fittest individual in each generation.

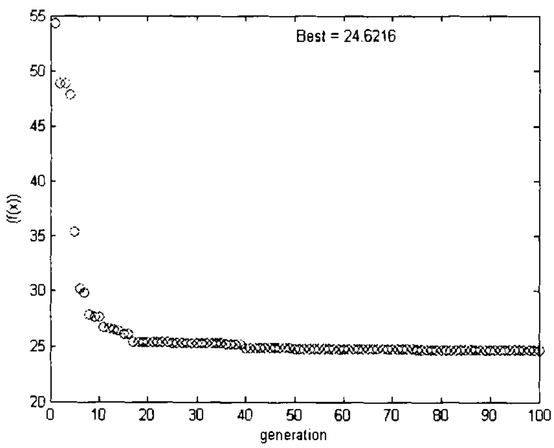


Figure 5-4 Objective value of the fittest individual in each generation

The optimised Fuzzy control power output for the power reference sequence is presented in Figure 5-5 along with the responses of the actuators and the ITAE performance.

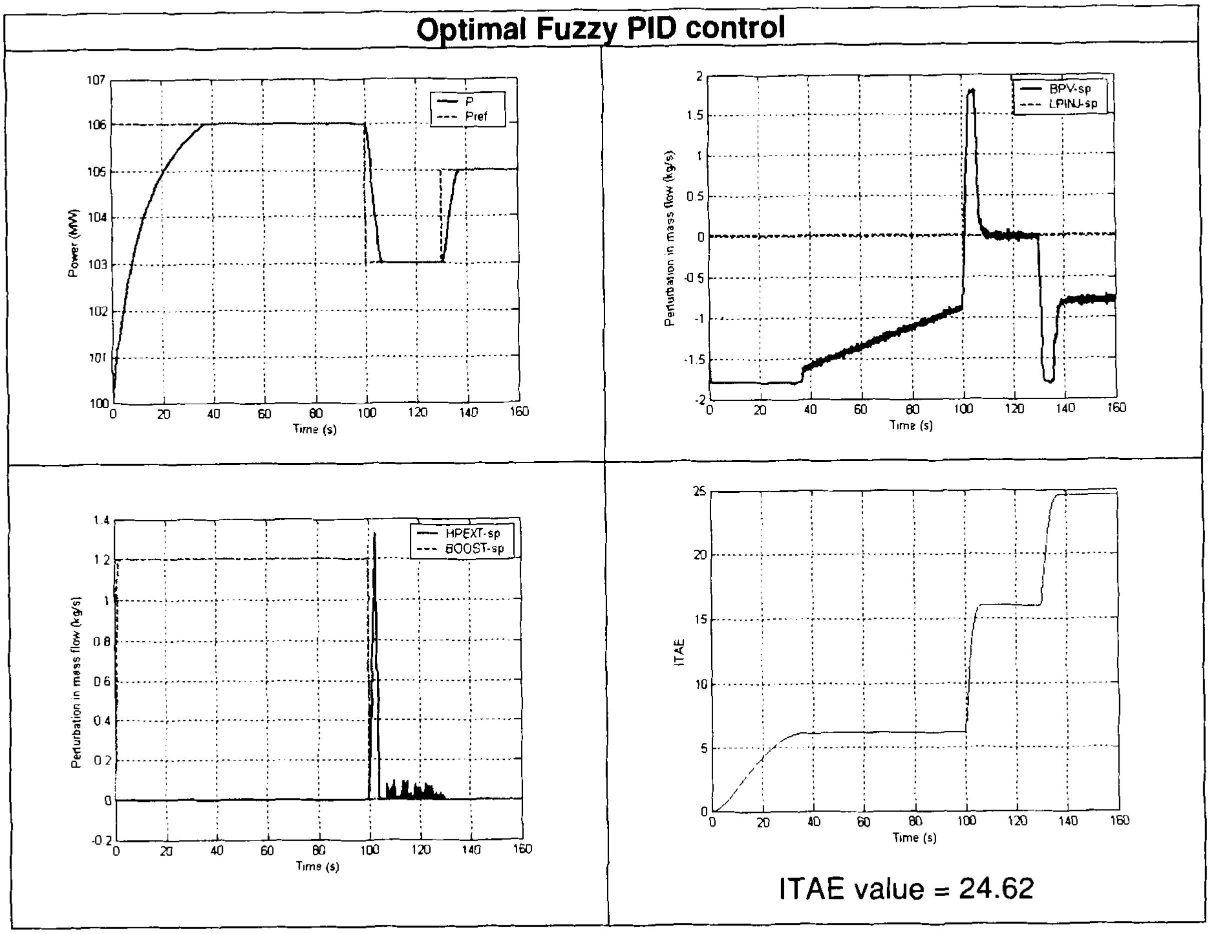


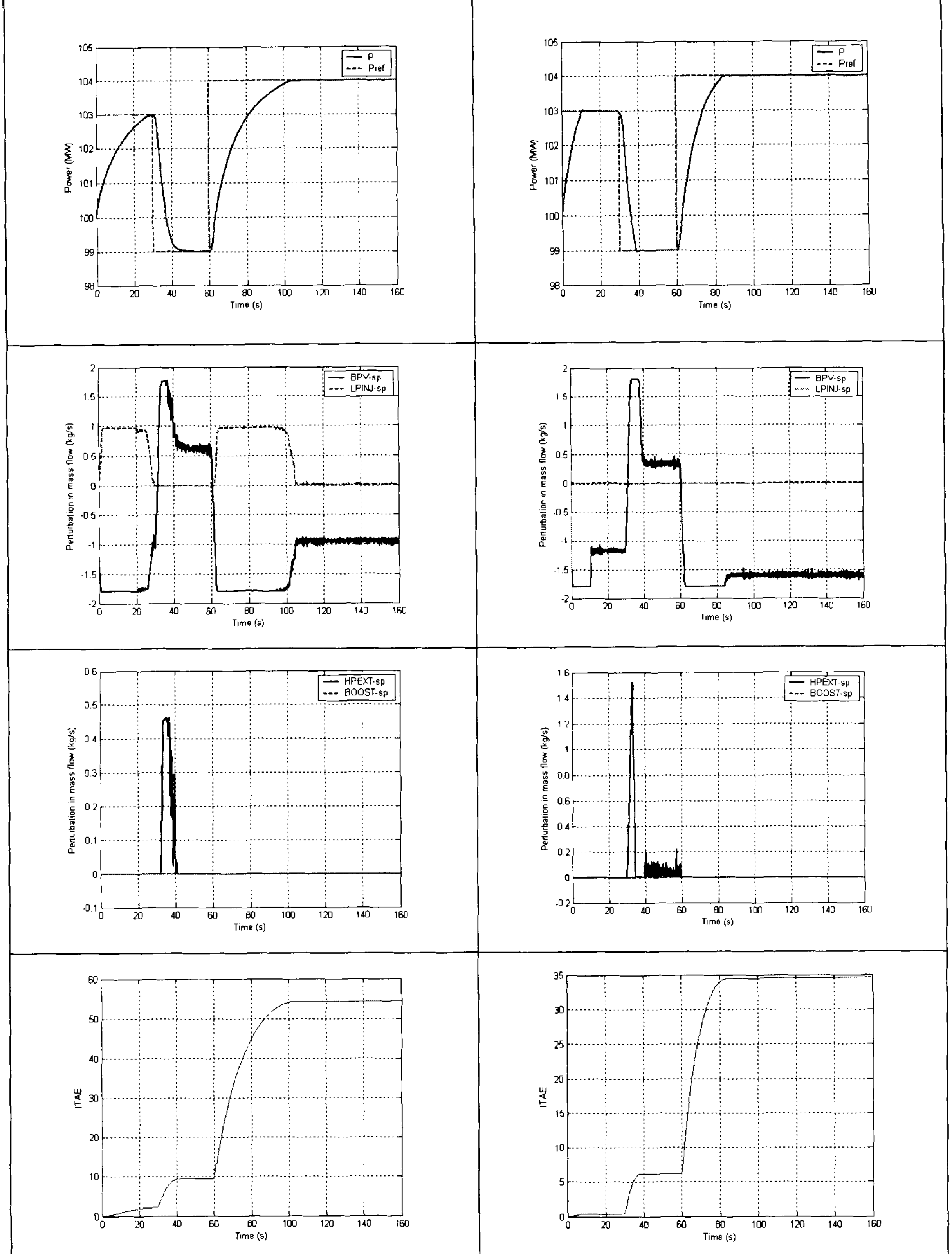
Figure 5-5 GA optimised Fuzzy control system results

The power response compared to Figure 5-3 shows no overshoot. This is a major improvement in this case. It can be seen that the value of the LPINJ-sp is zero as a result of the penalising effect of the GA. The BOOST-sp stays the same compared to Figure 5-3 and the time allowed for high-pressure extraction is shortened.

The optimal Fuzzy control strategy is also tested for a different power reference control signal and compared with the original Fuzzy control strategy. This will demonstrate if the control gains generated by the GA will work well for different scenarios.

Figure 5-6 shows the results of the optimal Fuzzy control strategy and the original Fuzzy control strategy given in chapter 4. The first graphs represent the power responses, followed by the responses of the four helium inventory actuators and ITAE performance.

Original Fuzzy control	**Optimal Fuzzy control**



ITAE value = 54.42 | ITAE = 34.66

Figure 5-6 Optimised Fuzzy control compared to original Fuzzy control

By inspecting the performance index, it is clear that the GA improved the power response of the Fuzzy control strategy. Again the low-pressure injection set point of the optimised Fuzzy control strategy is zero compared to the original. The peak value for the high-pressure extraction set point is a bit higher than that of the original resulting in a faster settling time when the power decreases.

5.11 Conclusion

The Fuzzy control strategy consisting of four Fuzzy subsystems was optimised by means of a GA. The GA operated on sixteen parameters simultaneously and after 100 generations produced parameter values that reduced the objective value of 77.7 to 24.62. This shows that the GA is an effective parameter optimisation technique. The optimised Fuzzy control strategy also improved responses for different power set point values as shown in Figure 5-6.

In chapter 4 the gains for both the PID and Fuzzy control strategies were chosen by a process of trial and error. During that process best results were achieved by choosing the integral constants to be zero. In the optimisation process of the Fuzzy control strategy the GA also generated values very close to zero for the integral gains. It is therefore concluded that the integral part can be discarded for both control strategies.

Chapter 6

Conclusions and recommendations

6.1 Introduction

In this dissertation the problem of optimally controlling the power output of a Brayton-cycle based power plant under normal load following conditions was addressed. This problem was basically broken up into three sub-problems. First an evaluation platform in the form of a Simulink® model needed to be constructed to be able to evaluate different control strategies. Secondly a control strategy needed to be derived that would also answer the question of what level of sophistication is needed for this control problem. Lastly an optimisation method was needed to optimise the specific control strategy. This chapter concludes how these problems were investigated and solved and possible future work that can be done is identified.

6.2 Concluding remarks

It was decided to investigate two control strategies namely PID control and Fuzzy control. A linear model of the power plant was created in the Simulink® simulation environment. This linear model was then used as a test platform for the two control strategies.

Each control strategy needed to generate four set point values for the four power manipulating actuators in the linear model. It was therefore decided to create control subsystems responsible for generating the respective set point values. In the PID control strategy each subsystem contained a PID controller, where in the Fuzzy controller strategy each subsystem contained a Fuzzy PD and I (FPD + I) controller.

The gains for the controllers were chosen by a method of trial and error due to the complexity of the system. Fairly good responses were shown by both control strategies. The Fuzzy control strategy proved to be just as good or better than the PID controller according to the performance index.

The Fuzzy control strategy was chosen to be optimised by an optimisation technique. Genetic algorithms (GAs) is a very attractive optimisation technique to use in this particular case because of the high complexity of the system. It can easily optimise a large amount of parameters simultaneously using a random search technique.

A specific GA was devised for optimising the gain values of the Fuzzy control strategy. The four subsystems of the Fuzzy control strategy have four gains each that needed to be optimised. This resulted in sixteen parameter values that have to be adapted by the GA. The GA was coded using real valued chromosomes. It was decided to use forty individuals in each population and to iterate the GA for 100 generations.

According to the ITAE performance index the GA greatly improved the performance of the Fuzzy control strategy. The optimised Fuzzy control strategy was tested for a different power set point sequence. A great improvement compared to the original Fuzzy control strategy was obtained.

It is also possible to apply the GA on the PID controller and optimise the gain constants. However the PID and FPD+I relations described in chapter 2 to derive an optimal PID controller from the optimal FPD+I controller can also be used. To demonstrate this optimal parameters from the bypass subsystem of the Fuzzy control strategy were used to derive an optimal PID bypass subsystem.

$$G_p = FG_u \cdot FG_p = 0.35015 \cdot 14.5071 = 5.07966$$

$$G_d = \frac{FG_d \cdot G_p}{FG_p} = \frac{83.9731 \cdot 5.07966}{14.5071} = 29.4032$$

$$G_i = \frac{FG_i}{FG_p} = \frac{0.00011757 \cdot 5.07966}{14.5071} = 0.000041$$

These constants were substituted in the original PID bypass subsystem and this already showed a large improvement on the power response of the PID control strategy. The result of the power responses and the performance is given in Figure 6-1.

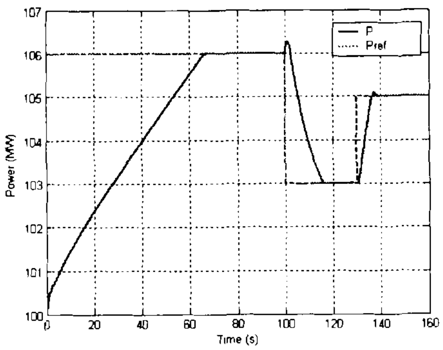
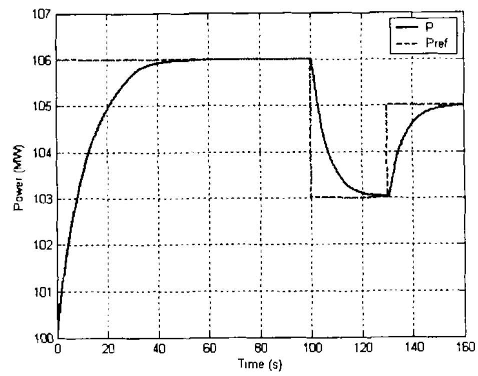
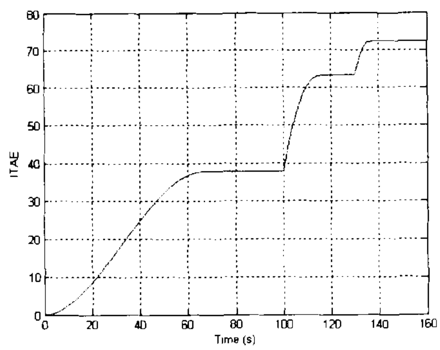
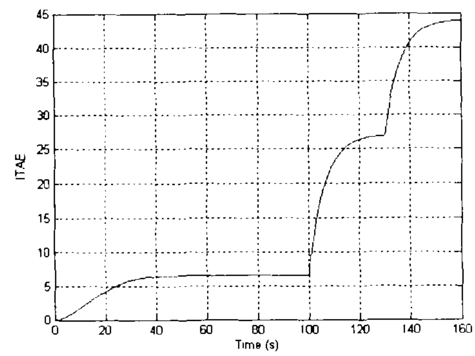
Original PID control	**Original PID control with optimised bypass subsystem**
	
 ITAE value = 72.44	 ITAE value = 43.9

Figure 6-1 Results for PID control strategy with optimised bypass subsystem

6.3 Future work

GAs is still a relatively new technique. There is still a lot of research to be done on GAs and specifically their interaction with Fuzzy systems. Different parts of the Fuzzy system can be optimised by means of a GA. The effect of optimisation of the rule base and the membership function parameter values are valuable future work to be done.

The cost function used in this study was not very complex. Further work on the cost function is needed to take other variables into account such as the reserve capacity of the system. Minimisation of stresses on the system is also an important factor.

In this case linear Fuzzy controllers were used that simulated PID controllers. These linear Fuzzy PID controllers can be converted to non-linear Fuzzy PID controllers by using Gaussian membership functions. It will be interesting to test these controllers and see if better results can be achieved.

The linear model of the power plant used to test the different control strategies can be improved by modelling more effects making it more realistic, or a non-linear model can be derived.

6.4 Closing remarks

The goals in terms of optimally controlling the power output of a Brayton cycle based power plant were achieved and the foundation for further research in control optimisation was laid.

REFERENCES

[1] A.P. George, "Introduction to the Pebble Bed Modular reactor (PBMR)," PBMR, Pretoria, Tech. Rep. 2001.

[2] M.C. Nieuwoudt and L.P. Cilliers, "Demonstration Plant Operations Summary," PBMR, Pretoria, Tech. Rep. Revision: 4, 2002.

[3] L. Magdalena and F. Monasterio-Huelin (1997), "A Fuzzy Logic Controller with learning through the evolution of its knowledge base," *International Journal of Approximate Reasoning.* No.5, pp. 335-35.

[4] M.R. Delgado, F.V. Zuben and F Gomide (2001), "Hierarchical genetic fuzzy systems," *Information Sciences,* No. 136, pp 29-52. Available: http:\\www.elsevier.com/locate/ins [Date of access: May 2004].

[5] W. Liebert, "Performance evaluation of a fast digital Fuzzy Controller for an active magnetic bearing," Laboratory of Electrical Engineering Design. ETH-Zürich, Switserland.

[6] Matlab® Version 6.5 Release 11, "Fuzzy logic Toolbox 2.1.2 Release notes".

[7] V. Georgescu, "Fuzzy Control Applied to Economic Stabilization Policies," Faculty of Economic Sciences, University of Craiovo, Romania. Available: http:\\www.ici.ro/ici/revista/sic2001/art5.htm [Date of access: June 2005]

[8] J. Jantzen, "Tuning of Fuzzy PID Controllers," Department of Automation, Technical University of Denmark, Denmark, Tech. Rep. 1998.

[9] K.F. Man, K.S. Tang and S. Kwong, "Genetic Algorithms," Springer-Verlag London Limited, Great Britain, 1999.

[10] A. Chipperfield, P. Fleming, H. Pohlheim and C. Fonseca, "Genetic Algorithm Toolbox for Matlab®," Department of automatic control and systems engineering, University of Sheffield, Users guide, Ver. 1.2.

[11] F. Herrera and L. Magdalena, "Genetic Fuzzy Systems: A Tutorial," Department of Computer Science and A.I., University of Granada, Spain.

[12] J.F. Pritchard, "Development of a linear model of the Brayton cycle," PBMR, Pretoria, Tech. Rep. 2002.

APPENDIX

A.1 Data CD

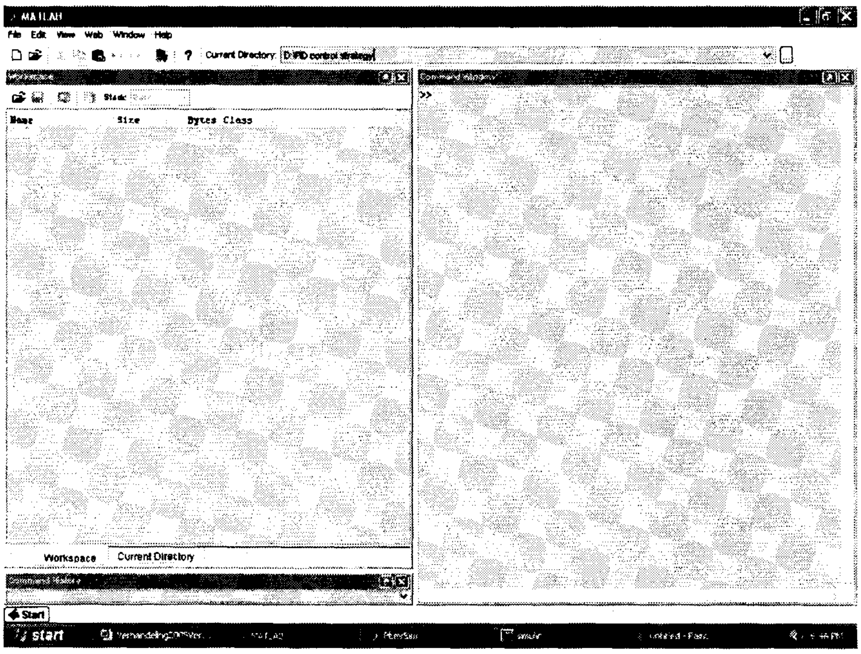
This dissertation contains a data CD at the back. This CD contains the Simulink® and Matlab® files used in this study. These files are stored in the following directories:

- ✓ PID control strategy
- ✓ Fuzzy control strategy
- ✓ Optimal Fuzzy control strategy

The procedure for running these files will be discussed shortly.

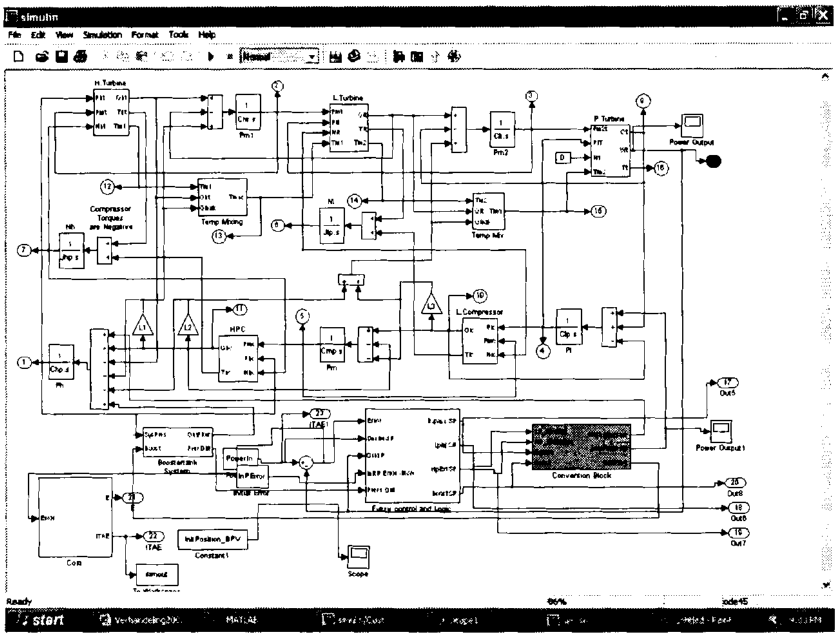
A.2 PID control strategy

- ✓ Load the folder "PID control strategy" into the current directory of Matlab®.



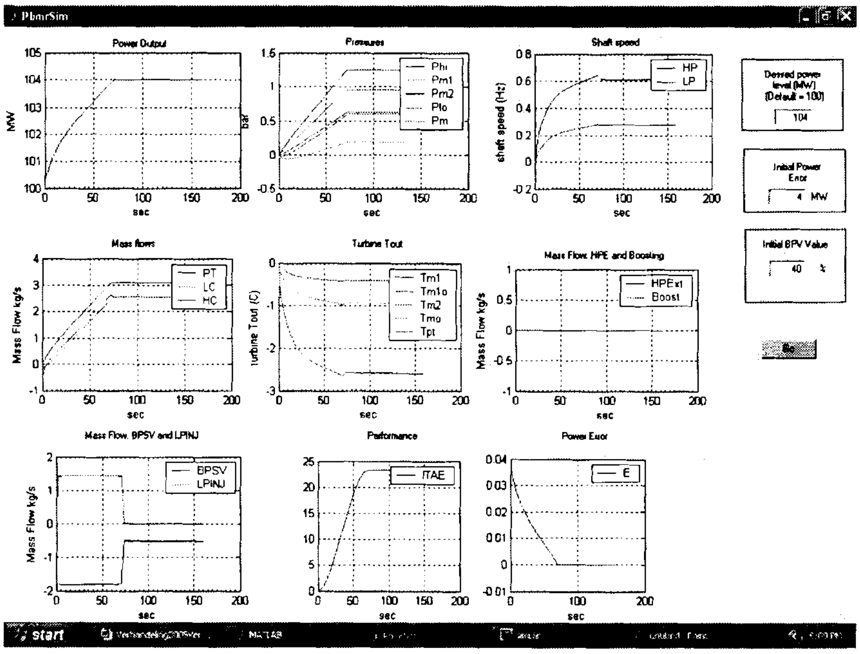
A-1 Matlab® command window

✓ Type "simulin" at the command line to open the Simulink® linear model.



A-2 Simulink® linear model

✓ To open the graphical user interface (GUI) developed for simulation purposes, type "PbmrSim" at the command line.



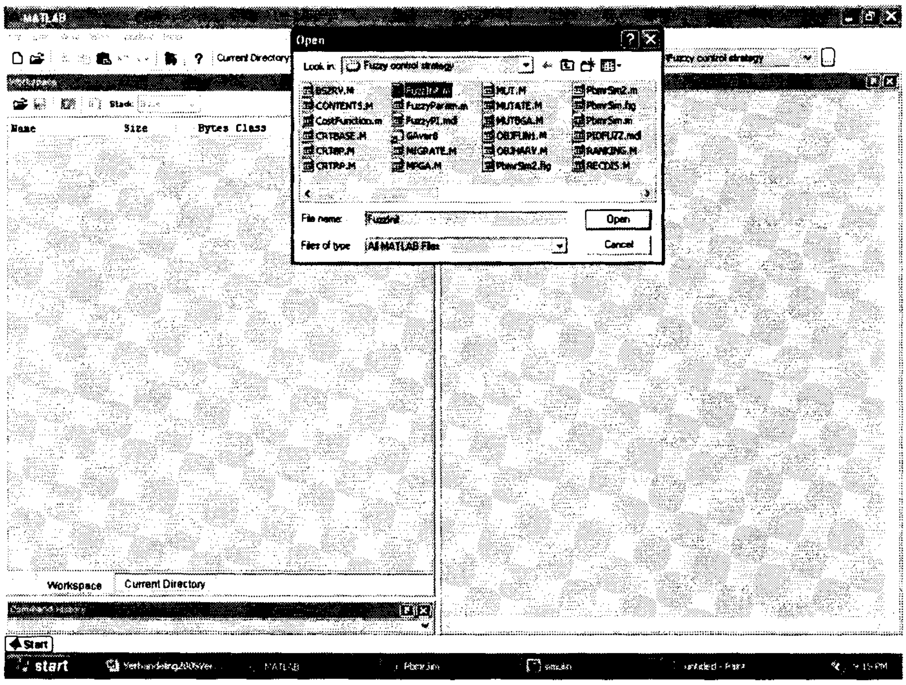
A- 3 GUI for system simulation

- ✓ The desired power level can then be typed in the "Desired power level" block. To simulate the system for a particular power level set point, click the "Go" button.

A.3 Fuzzy control strategy

- ✓ Load the folder "Fuzzy control strategy" into the current directory of Matlab®.

- ✓ Open and run the m-file "FuzzyInit" to initialise the Four Fuzzy controllers.



A-4 Initialisation of the Fuzzy controllers

- ✓ Type "PbmrSim" to open the GUI and simulate the system for a desired power level.

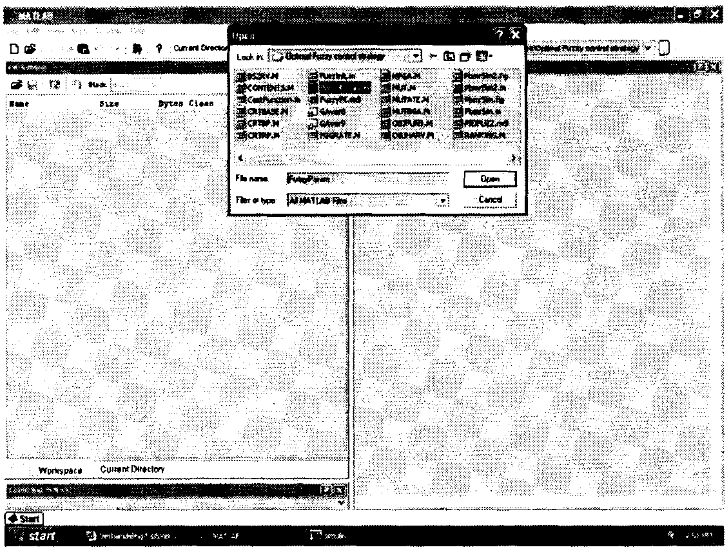
A.3 Optimal Fuzzy control strategy

- ✓ Load the folder "Optimal Fuzzy control strategy" into the current directory of Matlab®.

- ✓ Open and run the m-file "FuzzyInit" to initialise the Four Fuzzy controllers.

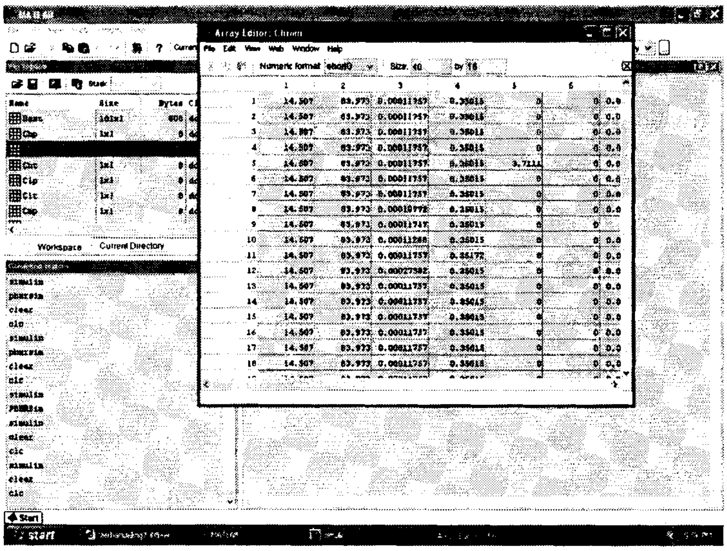
- ✓ Run the "PbmrSim" GUI for a specific power reference

- ✓ To run the Genetic Algorithm that optimise the Fuzzy control system, open and run the m-file "FuzzyParam"



A-5 Genetic Algorithm optimisation

- ✓ The chromosome values are stored in the matrix "chrom"



- ✓ The objective function values are stored in the vector "ObjV"

- ✓ The fitness function values are stored in the vector "FitnV"