Title: Modelling and Predictive Control of a Drum-Type Boiler

Candidate: Barbara Molloy, B.E.

Ph.D. Thesis

University Name: Dublin City University

Supervisor: Dr. J.V. Ringwood

School: School of Electronic Engineering

Month and Year of Submission: February, 1997

I hereby certify that the material, which I now submit for assessment on the program of study, leading to the award of Ph.D is entirely my own work and has not been taken from the work of others save and to the extent that such work has been acknowledged with the text of my work

Signed: Barbara Molloy Id No. 92700179

Date: 5th Feb. 1997

Acknowledgements

I would like to express my sincere gratitude to the following people,

My supervisor, Dr. John Ringwood for his invaluable help and guidance over the past four years.

Dr. Jacques Richalet of Adersa, Paris for generously educating and advising me on the subject of predictive control.

Guinness, Ireland and in particular Tony O'Sullivan and Tony McFadden for providing me with plant data.

The members of the Control Systems Group for all kinds of help and support.

School of Electronic Engineering for funding this work.

Finally, I would like to thank my family and especially my mother, Eileen, for their much appreciated support and encouragement.

Contents

MODELLING AND PREDICTIVE CONTROL OF A DRUM-TYPE BOILER

BARBARA MOLLOY

ABSTRACT

Boilers generate steam continuously and on a large scale. Controlling the boiler process is extremely difficult - it is a highly nonlinear process, its dynamics vary with load and it is strongly multivariable. It is also inherently unstable due to the integrator effect of the drum. In addition, boilers are commonly used in situations where the load can change suddenly and without prior warning.

Traditionally, boilers have been controlled by Single-Input, Single-Output (SISO) Proportional plus Integral (PI) controllers. This strategy does not take into account the interaction of the controlled variables or the effect of load on boiler dynamics.

This work investigates whether boiler control can be improved by applying multivariable or nonlinear predictive control strategies. Predictive control is a model-based control strategy which is chosen for its ability to handle nonlinear, constrained and multivariable systems.

Two nonlinear controllers are developed - a fuzzified linear predictive controller which is based upon several linearised models of the plant and and a nonlinear predictive controller, based upon a single nonlinear plant model. These controllers are compared both with each other and with the conventional PI control strategy.

A detailed first-principles model of the boiler is developed for this work. This is used to simulate a boiler plant for controller testing. It is also used to derive a linear state-space model for the linear predictive controller. The nonlinear predictive controllers uses a neural network model.

1. Introduction

1.1 Background to Problem

Large-scale steam generation is a complex and expensive process. Boilers consume large amounts of fuel and produce considerable amounts of carbon dioxide and other environmentally damaging gases. Improving boiler control pays large dividends, in terms of reduced fuel costs, reduced pollution, improved safety and an extended plant life-time.

Traditionally boilers have been controlled by PI control. The typical PI controller configuration is well understood, however, given the complexity of the boiler control problem, it is likely that superior control could be achieved using a more sophisticated control methodology. The boiler process is subject to a number of control problems:

- The boiler process is highly nonlinear - its dynamics vary with operating point. This was not a significant problem in the past as it was common practice to operate boilers at their maximum capacity only. However, in order to improve efficiency many boilers are now operated in cycling mode. For example, a boiler is operated at maximum capacity during the day when the demand for steam is high and at half capacity during the night when the demand has decreased. In consequence, the control system must be capable of providing good control over a range of operating points.
- The boiler process is strongly multivariable. A control system based upon single input, single output (SISO) controllers does not account for interactions among the controlled variables with the result that the individual control loops must compete with each other in an attempt to achieve their individual objectives.
- The boiler process is inherently unstable due to the integrating effect of drum level. Instability in drum level, for instance, has very serious physical consequences.
- It is a stiff system - considerable differences exist between the speed of response of the controlled variables.
- Non-minimum phase effects exist in both drum and superheater.
- A significant time delay is inherent in the transportation of combustion gases through the furnace.
- Boilers are commonly used in a situation where the load can change suddenly and without prior warning.
- The process must operate within tight constraints.

Various linear controllers have been developed which can address one or more of these problems. For example, optimal control can provide improved boiler control performance by virtue of being a model-based strategy. A model-based controller has a significant advantage over a non-model-based controller in the case of a non-minimum phase system or a system with an integrator. Further improvements have been achieved by using multivariable controllers which take into account the interactions among the controlled variables. The issue of boiler nonlinearity has been addressed by various adaptive and gain-scheduling control techniques which take into account that the model dynamics vary with operating point. However, adaptive techniques can present additional problems with regard to stability and convergence issues.

1.2 Research Objectives

The first objective of this work is to address all of the boiler control problems in a coherent and systematic manner through the use of a nonlinear, multivariable predictive control strategy. The second objective is to assess the benefits of this strategy in terms of controller performance.

The benefits of nonlinear control are investigated by developing two types of nonlinear predictive controllers and comparing these to the industry standard PI control strategy. The first nonlinear controller is based upon several linear controllers and is referred to as a gain-scheduled or fuzzified linear controller. This is the traditional method of implementing a nonlinear controller. The second nonlinear controller is based upon a single nonlinear model of the boiler and is referred to in this work as the nonlinear controller. Many of the advanced, model-based control strategies are compared to a PI control strategy only. If the PI control strategy and the advanced control strategy are not of similar complexity, this comparison may not provide a useful assessment of the performance or potential of the advanced control strategy. In this instance, the nonlinear and gain-scheduled linear controller are both based upon a predictive control methodology.. Comparison of the nonlinear controller to the gain scheduled linear controller highlights the advantages and disadvantages of each of the two nonlinear control strategies.

All the controllers are tested using a detailed, first principles model of a boiler plant. The first-principles model is a comprehensive model which includes all the significant plant dynamics and has been validated against plant data.

1.3 Contributions of this Thesis

The objective of this work is to make a comprehensive assessment of the performance of a nonlinear controller for boiler control. Fundamental to this work is the development of a detailed

boiler model which can accurately simulate the performance of a boiler plant. A first-principles boiler model has been developed which uses the minimum number of simplifying assumptions. In particular, the model incorporates the following features:

- Detailed model of combustion process and gas composition, including measurement dynamics
- Improved modelling technique for *'Shrink and Swell'* phenomenon
- Improved implementation of water and steam tables
- Non-minimum phase effect in superheater is modelled
- Steam in drum is not assumed to be saturated

The performance of the nonlinear controller is dependent to a large extent on the accuracy of the internal controller model and on the optimisation technique employed. Features of controller development are:

- Feasibility of using a neural network model for nonlinear predictive control is demonstrated.
- Feasibility of using either gradient search or genetic algorithms to optimise the control signal is demonstrated.
- Specification of actuator constraints in the nonlinear controller design.

The success of nonlinear boiler control is measured by comparison with an adaptive or gain-scheduled controller of equal complexity. The validity of the comparison has been improved by:

- Development of nonlinear controller and gain scheduled linear controller using the same design methodology in order to achieve a meaningful assessment of the nonlinear control strategy.
- Improved gain scheduling technique is applied to the linear control strategy. Fuzzy logic is used to ensure "bumpless" transfer between linear controllers.

1.4 Layout of Thesis

This layout reflects the logical progression of controller development. - commencing with boiler model development, followed by a description of controller development and finishing with an assessment of controller performance.

In this instance, three types of boiler model are developed. Firstly, a detailed first-principles boiler model which can provide insight into the steam generation process and act as a plant simulator is developed. Secondly, a computationally efficient linear boiler models is developed for use by a linear model-based predictive control strategy. Thirdly, a computationally efficient

nonlinear model is developed for use by a nonlinear model-based predictive control strategy.

Controller development includes the development of both a gain-scheduled linear predictive controller and a nonlinear predictive controller. The gain-scheduled linear predictive controller provides a valuable point of comparison for the nonlinear predictive controller in addition to the usual benchmark - a PI control strategy.

The thesis layout is presented on a more detailed chapter-by-chapter basis to illustrate this progression:

In Chapter 1, the background to the boiler control problem and the motivation for improved control is presented. The objectives of this work are stated and the contributions of this work to boiler control research are summarised.

The process of steam generation in a typical drum-type boiler is described in Chapter 2. This chapter offers a simple understanding of a process which is described in more abstract terms in the following chapters. It also introduces the reader to much of the terminology employed throughout this thesis.

In Chapter 3, previous approaches to boiler modelling and control are reviewed and the advantages and disadvantages of these approaches are detailed. The typical motivations for improving boiler control and many common boiler control problems are also outlined. In the light of this review, areas of possible improvement in both boiler modelling and control are suggested. This chapter places the contributions of this thesis into the context of current approaches to boiler control.

In Chapter 4, the development of a detailed first principles model of the boiler plant is described. Development of this model is the first task to be undertaken, for two reasons. Firstly, the model provides physical insight into the steam generation process which is valuable during controller development. Secondly, the boiler is required to act as a plant simulator.

Chapter 5 describes the first stage in the development of the gain-scheduled linear predictive controller i.e. development of its internal controller model. The gain-scheduled controller is based upon a number of linear boiler models, which are obtained by linearising the first-principles boiler model. The linearisation procedure is described in detail.

The nonlinear predictive controller is based upon a nonlinear model of the boiler plant. A neural network model is used for this task as neural networks are both computationally efficient and can be used to model any type of nonlinear plant. Development of the neural network model is

described in Chapter 6.

Chapter 7 introduces the concepts and theory underlying predictive control. Predictive control theory offers the control engineer a variety of controller design and tuning options. Common controller design and tuning options are described and their effect on controller performance.

The development of the gain-scheduled linear predictive controller for a boiler is described in Chapter 8. In particular, the mechanism used to ensure a smooth transfer of control between the individual linear controllers is outlined.

In Chapter 9, the development of a nonlinear predictive controller for a boiler is described. The advantage and disadvantages of using either a gradient search optimisation or genetic algorithm optimisation are considered.

In Chapter 10, the linear predictive controller, nonlinear predictive controller and an industry standard PI boiler controller strategy are compared on the basis of controller performance, design methodology and implementation requirements. Controller performance is assessed for both disturbance rejection and setpoint changes. An extensive set of simulation results is provided.

Conclusions on this work are presented in Chapter 11.

2. Physical Description of a Drum-Type Boiler

2.1. Introduction

Boilers produce steam for a wide range of industrial purposes - electricity generation, chemical processes, heating. The process is continuous and large scale. A typical medium sized boiler generates 30,000 kg of steam per hour, at a temperature of 420^{o}C and a pressure of 4.5 MPa. A very large scale electric utility boiler may generate more than 4,000,000 kg of steam per hour.

There are two fundamental requirements for generating steam - water and heat. These two aspects of steam generation are commonly referred to as the water-steam side and the combustion side respectively. Fig. 2.1 is a schematic representation of a typical drum-type boiler.

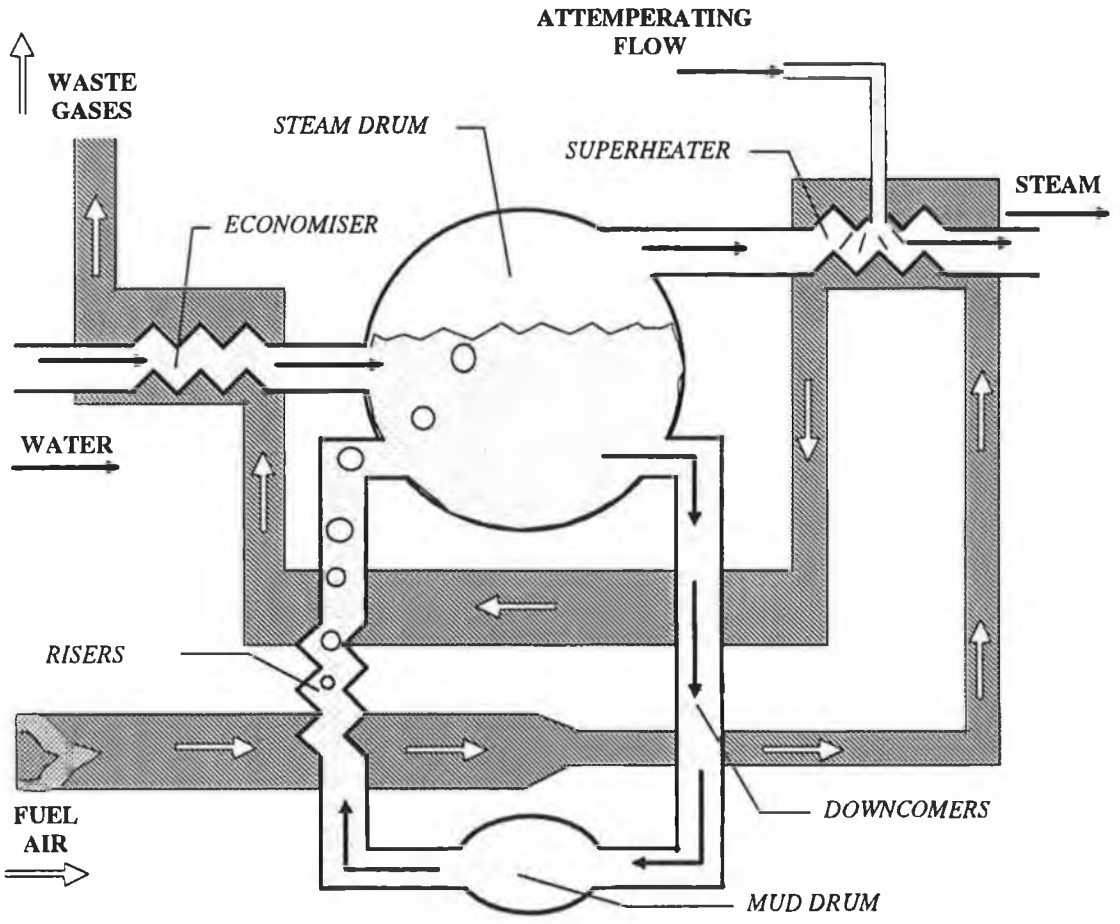


Fig. 2.1 Schematic of Boiler

The water-steam side of the boiler process involves converting water into high-temperature steam. The temperature at which water evaporates is known as the saturation temperature. Water below this temperature is referred to as subcooled. Subcooled water enters the boiler, is heated to saturation temperature and starts to evaporate into steam. Both the water and steam remain at

saturation temperature until all the water has been evaporated. Throughout evaporation, the water and steam are referred to as saturated. After all the water has been evaporated, the steam can be heated to a higher temperature. Steam at a higher temperature than saturation temperature is called superheated steam.

The combustion-side involves burning fuel to generate the heat necessary for steam generation. Fuel must be mixed with the correct volume of air in order to ensure complete combustion. The resulting gases are then directed around the furnace via a system of baffles losing heat to various sections of the boiler along the way. Safety and environmental considerations are of primary importance on the combustion side.

Much of the material in this chapter is based on Dukelow (1986) and Lindsley (1991).

2.2. Water-Steam Side

Water Treatment

The first phase in the steam generation process is water purification. Before water can enter the boiler proper, debris and other suspended matter must be removed by coagulation and filtration. The water must also be dimineralised. If this is not done, these impurities will be deposited on the inside of the tubes within the boiler and will quickly reduce boiler efficiency.

Aeration

Following chemical treatment the water is fed to aerator tanks, where dissolved gases are removed from the water. Even small traces of oxygen can cause corrosion of the boiler surfaces at the high temperatures in the boiler. Carbon dioxide would pass into the steam and turn into corrosive carbonic acid in the steam heat exchangers. To prevent this the gases are removed by boiling and agitating the water and by venting the gases to the atmosphere. The water is boiled by mixing it directly with steam. Scavenging chemicals are also used to remove any remaining gases.

Feedwater Pumps

Following water purification and aeration, the water, referred to as feedwater, is pumped into the economiser. The feedwater pump must supply at least enough energy to overcome the pressure head in the drum. A large boiler which is providing 1,100,000 kg of steam per hour (or 660 MW of power) will require two pumps each consuming approximately 10 MW of power.

Economiser

The economiser is a simple heat exchanger. Subcooled feedwater water enters the economiser and is heated to a higher temperature, but not to saturation temperature. This is a means of improving boiler efficiency, as waste heat in the combustion gases is used to preheat the water before it enters the evaporation system.

Evaporation System

The evaporation system of a drum-type boiler is comprised of a steam drum, downcomers, water drum and risers. The steam drum is a large cylindrical vessel which is approximately half filled with water. The mud drum is a smaller cylinder situated below the drum containing water only. The steam drum is much larger than the mud drum and is commonly referred to as the drum. The downcomers and risers are banks of tubes connecting the drum and mud drum.

Subcooled water from the economiser flows into the water in the steam drum. It mixes with subcooled water in the steam drum and saturated water from the risers. The subcooled drum water flows into the risers via the mud drum and downcomers. In the risers, the subcooled fluid is heated to saturation temperature and a percentage of the saturated fluid evaporates. The saturated water-steam mixture in the risers is at a lower density than the water in the steam drum and flows back into the steam drum. This establishes a natural circulation system in the drum, downcomers and risers. Water circulates continuously around this system, passing from the steam drum, through the downcomers into the mud drum and returning to the steam drum via the risers. It is possible to increase the recirculation rate by incorporating a recirculation pump at the mud drum. This type of recirculation is referred to as forced recirculation.

Steam Drum

The steam drum acts as a separator for the water and steam in the evaporation system - steam is drawn off the top of the steam drum and water is fed to the downcomers which are situated at the bottom of the steam drum. The saturated steam in the steam drum contains moisture droplets. These droplets could cause thermal shock if they fell on the high temperature tubes in the superheater. They are removed from the saturated steam by mechanical separation devices called "scrubbers" or "separators" which return the moisture droplets to the steam drum. The "scrubbers" also prevent the carryover of boiler water chemicals into the steam.

The level of water in the steam drum must be maintained very carefully. If the drum water level falls too low, there is a risk of dry-out. If the drum water level rises too high, the drum water surface area will be reduced, which in turn reduces the amount of steam that can be generated by

the boiler. In order to provide accurate drum level control it is important to have accurate and reliable drum level measurement. If the drum level is measured directly above the risers, the rising steam bubbles may cause frequent variations in the measured drum water level. The drum water level should be measured above the downcomers where the water level is relatively stable. Problems with drum level measurement can arise if a downcomer receives too much heat and acts as a riser i.e. generates steam bubbles. This is avoided by careful boiler design. Flue gas baffles are installed to shield the downcomers from the combustion flame and also to direct gas flow. The combustion gases should be allowed to flow over the downcomers only after they have lost most of their heat.

Drum level control is further complicated by the phenomenon known as *Shrink and Swell.* This phenomenon occurs during transient changes in the steam flow from the boiler. An increase in the steam flow from the drum causes a reduction in steam pressure in the drum. The reduction in drum pressure causes the vapour bubbles in the evaporation system to expand and the density of the water in the evaporation system to decrease. Both effects reduce the density of the water-steam mixture and increase the overall volume of the water-steam mixture in the drum, causing the drum water level to rise - *Swell* for some time. The drum level falls when the increased evaporation rate takes affect.

The converse occurs during a decrease in steam demand. Steam pressure increases causing a reduction in bubble size and an increase in water density throughout the evaporation system. The overall density of the water-steam mixture increases and drum level suddenly reduces - *Shrink.* After some time, the drum level starts to increase due to the decreased evaporation rate.

Blowdown

The chemicals contained in the feedwater accumulate in the boiler water. In order to maintain a proper chemical balance of the boiler water, they must be removed. This removal is called blowdown and is carried out in two ways. Firstly, water containing dissolved solids is continuously removed from the water in the steam drum. Secondly, sludge is removed periodically by opening a valve in the lowest part of the mud drum.

In order to maintain a chemical concentration of the boiler water 10 times that of feedwater, blowdown must be 10 percent of feedwater. The greater the percentage blowdown the higher the associated heat losses. At higher pressures, the blowdown water has greater energy and the blowdown heat loss is greater. The blowdown flow is generally controlled automatically using water conductivity measurements as a measure of the chemical concentration of the water.

Superheater

The saturated steam in the drum must be superheated to a higher temperature before it can do mechanical work in a turbine. In the turbine the steam will lose heat and fall in temperature. If the steam entering the turbine is saturated, it will condense back to water. If condensation occurs in the turbine severe thermal shock will ensue. To prevent this occurring the steam must be heated to a sufficiently high temperature, so that it will still be superheated as it leaves the turbine.

The steam can be superheated by radiation or convection. The steam temperature in a radiant superheater decreases as steam flow increases. The steam temperature in a convective superheater increases as steam flow increases. By combining radiant and convection superheaters in series, the steam temperature variations caused by changing load can be minimised.

The temperature of the superheated steam can also be affected by varying the amount of heat received by the superheater. This can be done crudely by changing the firing rate or in a more sophisticated way by tilting the burners either towards or away from the superheaters.

Steam temperature also depends on a lot of other factors which cannot be varied for control purposes - steam pressure, state of heat transfer surfaces and others.

Attemperation

It is not feasible to accurately control the temperature of the steam leaving the superheater by any direct means such as varying the firing rate. This can at best provide only a very crude steam temperature control due to the slow dynamics and delays in the system. The usual means of providing accurate steam temperature control is attemperation; steam is heated to a higher temperature than required in the superheater and then cooled to the desired temperature.

Attemperation typically involves spraying the steam with cooling water. This may be carried out in a single stage or there may be a stage of attemperation for each superheating stage. Another and less common type of attemperation involves passing a fraction of the superheated steam through tubes which are suspended in the drum water. This is a more heat efficient form of attemperation as the heat is transferred from the superheated steam to the drum water.

Load

A boiler supplies steam to one or more steam consuming processes such as turbines or heating systems. The steam flow rate from the drum is partially determined by the pressure in the steam

drum and partially determined by the demands of the steam consuming process. For example, the steam consuming process can usually demand more or less steam by opening or shutting valves on the steam flow pipeline. The steam flow from the drum is referred to as the load. For example, if the steam consuming process requires the maximum steam flow, the boiler is operating at full load. If the steam consuming process requires half of the maximum steam flow, the boiler is operating at half load. The boiler and its load cannot be considered in isolation from each other. Any change in load demand causes important changes within the boiler. A load increase, for example, reduces boiler steam pressure. Lowering the steam pressure, reduces the saturation temperature and produces a short term increase in the steam generated by the boiler (compensating temporarily for the increased load). Changes in steam pressure also cause the phenomenon known as *Shrink and Swell*, which results in a temporary change in drum level.

2.3. Combustion Side

Fuels

Gas, oil and coal are the most common boiler fuels. Gas and oil are both fluids and consequently are easier to manage than solid fuels such as coal. Simple gas or oil burners are used for these.

There are several methods for burning coal. It can be pulverised and burnt with conventional burners. Alternatively, solid coal may be burned on a chain grate or a fluidised bed. A chain grate moves slowly and continuously under the furnace. Coal is loaded onto the grate, carried under the furnace for burning and the remaining ashes drop from the grate into a hopper.

In a fluidised bed boiler, coal and crushed limestone are fed onto a bed which is continuously agitated by compressed air from below. The coal burns at a lower temperature so that the emission of NO_x gases is reduced. The crushed limestone also traps the SO_2 produced by combustion.

Other solid fuel, such as wood waste, sugar cane, coffee grounds or municipal refuse can also be used by boilers. These can be burnt on a chain grate or in a fluidised bed boiler.

Air

The relative proportions of fuel and air are important for a number of reasons. If there is insufficient air for combustion, unburned fuel can collect downstream. This unburned fuel may explode when it comes into contact with air. Black smoke and poisonous carbon monoxide are another consequence of incomplete combustion. If there is an excess of air, however, boiler efficiency is reduced as this excess air also absorbs some of the heat from combustion.

The air entering the furnace may be preheated by the waste combustion gases. For solid fuel burners, this has the advantage of removing surface moisture from the fuel and improving combustion. It is important not to extract too much heat from the water combustion gases however as it is possible to cool them to the extent that acids from the combustion gases condense on metal tubes in the boiler.

Forced draft fans may be used to blow the air for combustion into the boiler. In a balanced draft plant an induced draft fan is used to pull the hot flue gases towards the stack and maintain a constant gas flow through the furnace. A slight negative pressure is generally maintained in the furnace by the fans. This prevents fuel leakage around the burning equipment and prevents hot combustion gases escaping through the chimney walls. It also maintains an inward pressure on inspection portholes and other weak points in the combustion chamber.

Combustion Chamber and Stack

Fuel and air are mixed and ignited in the furnace. There may be several burners in the furnace. By varying the tilt of these burners, or the proportion of fuel received by a particular burner, it is possible to vary the amount of heat transferred to the risers or to the superheater.

Flue Gas Treatment

The flue gases produced by burning coal contain small particles of incombustible solids and contaminating gases - notably sulphur and nitrogen oxides. For environmental reasons, these contaminants must be removed before the flue gases enter the atmosphere. The solids can be removed from the gas using filters or precipitators. The contaminating gases are removed chemically.

2.4 Conclusions

It is clear from the above that steam generation is a complex and potentially dangerous process. The description highlights some of the more important process control problems:

- The combustion side and steam-water side are two very different processes. However, the interaction of these processes must be carefully regulated. Poor regulation will have serious consequences. If the evaporation rate, which is largely dictated by the fuel flow rate is greater than the feedwater flow rate, drum water level will drop. If no corrective action is taken, this will ultimately cause the drum to dry out. Likewise, if the evaporation rate is less than the feedwater flow rate, drum level will continue to rise. Ultimately, the drum water will

overflow into the superheater tubes, causing massive thermal shock. Drum level control is further complicated by the phenomenon known as *Shrink and Swell.*

- Superheated steam temperature must be maintained within tight limits to prevent thermal shock occurring.

- Air flow rate must be sufficient to ensure complete stoichiometric combustion. However, an excess of air reduces boiler efficiency.

- Variations in the steam demand (or load) affect pressures and temperatures throughout the boiler.

It is not possible to gauge the true complexities of these problems without a better understanding of the process dynamics. For any process, a rigorous understanding is best achieved by a first-principles model. In Chapter 4, a detailed first-principles boiler model is described which provides far greater insight into the steam generation process.

3 Review of Current Boiler Modelling and Control Developments

3.1 Introduction

The question of applying advanced control methodologies and technologies to boiler control first arose in the 1960's. Since then, there has been a consistent interest in the potential for improving boiler control. This interest can be attributed in part to some new developments in the economic and industrial environment:

- Rising fuel costs have prompted plant engineers to find new ways of improving boiler efficiency. One approach has been the construction of larger and more complex power plants which can be operated at higher pressures and temperatures and thus more efficiently (Nicholson (1964)) and (Unbehauen and Kocaarslan (1990)). Such plants require very sophisticated control schemes.
- Improvements in efficiency have also been achieved by operating boiler plants in a cycling mode i.e. operating at a high load during the day when demand is large and operating at a low load at night when demand is low. The ability of a plant to respond quickly and efficiently to load changes is the dominant motivation cited in the literature for improving boiler controls (McDonald and Kwatny (1973)), (Kitami *et al* (1978)), (Nakamura and Akaike (1981)), (Sato *et al* (1984)), (El Sayed *et al* (1989)), (Nakamura and Uchida (1989)), (Nomura and Sato (1989)), (Gibbs and Weber (1992)) and (Ding and Hogg (1991)). The speed at which the load may safely be changed is limited to a large extent by controller performance. The transition between two operating points causes variations in metal temperatures which must be minimised to reduce metal fatigue and extend plant life-time (Unbehauen (1969)), (McDonald and Kwatny (1973), (Rossiter *et al* (1991)) and (Matsumura *et al* (1993)). It also causes pressure and drum water level fluctuations. A good control scheme must be capable of maintaining such variations within tight limits.
- Stricter environmental controls have been introduced with respect to combustion gas emissions. More sophisticated control systems are required to ensure that environmental restrictions can be met (El-Rabaie and Hogg (1991)), (Ding and Hogg (1991)), (Wu *et al* (1992)) and (Gibbs and Weber (1992)).

Higher efficiency and reduced emissions are not the only motivations for developing improved boiler control strategies. In many instances, the motivation is simply that the existing boiler controls are inadequate to cope with the complexities of the boiler process. The boiler process

has many characteristics which can present particular control difficulties:

- Significant interaction exists between the controlled variables. The relatively standard practice of using a set of SISO controllers (Rosemount, Baileys, Westinghouse) to control the boiler process may not deliver a good closed loop system response (Nicholson (1964)), (El-Sayed *et al* (1989)), (Sun Demin *et al* (1989)), (Unbehauen and Kocaarslan (1990)), (Ding and Hogg (1991)), (Unbehauen *et al* (1991)), (Rovnak and Corlis (1991)), (Gibbs and Weber (1992)) and (Wu *et al* (1992)).
- The process parameters are strongly dependent on load. A single linear controller which is tuned to control a boiler at full load may be inadequate at half load (Unbehauen (1969)), (Fernandez-del-Busto (1985)), (Sato *et al* (1984)), (Uchida *et al* (1988)), (El-Rabaie and Hogg (1989a)), (El-Sayed *et al* (1989)), (Unbehauen and Kocaarslan (1990)), (Ding and Hogg (1991)), (Unbehauen *et al* (1991)), (Gibbs and Weber (1992)), (Wu *et al* (1992)), (Matsumura *et al* (1994)).
- The process parameters drift over time for a number of reasons such as the build-up of soot on heating surfaces, actuator wear, variations in fuel quality. It may be necessary to continuously update the controller parameters (Unbehauen (1969)), (Nakamura and Akaike (1981)), (Uchida *et al* (1988)), (Nakamura and Uchida (1989)), (Sun Demin *et al* (1989)), (Unbehauen and Kocaarslan (1990)), (Pellegrinetti *et al* (1991)), (Ding and Hogg (1991)), (Unbehauen *et al* (1991)), (Wu *et al* (1992)) and (Matsumura *et al* (1994)).
- The process includes varying time delays (Pellegrinetti *et al* (1991)), (Pellegrinetti and Bentsman (1992) and (Wu *et al* (1992)).
- Both drum water level and main steam temperature are subject to non-minimum phase behaviour (Gyun Na and Cheon No (1990)), (Pellegrinetti *et al* (1991)) and (Elshafei *et al* (1995)).
- Drum water level is an integrator. This is a potential source of system instability. (Pellegrinetti *et al* (1991)).
- The boiler process is a stiff system (Rossiter (1991)). For example, the dynamics of main steam temperature are significantly faster than those of main steam pressure.
- The boiler may be subject to unpredictable load disturbances ((Wu *et al* (1992)) and (Broderick *et al* (1995))

These characteristics must be taken into account if optimum plant performance is to be achieved in terms of increasing plant efficiency and reducing environmental impact. To date, no "off-the-shelf" control solution is capable of optimising the performance of such a complex process. Nevertheless, many of the above problems can be successfully tackled by the application of

appropriate control strategies. The reviewed literature investigates a variety of mechanisms for improving boiler performance.

3.2 Boiler Modelling

3.2.1 Overview

The development of a successful boiler control strategy is largely dependent on the availability of a good boiler model. This can be seen from a review of the boiler control literature, a good part of which is dedicated to a discussion of boiler modelling techniques. A model is necessary both for controller optimisation and as a simulator for testing new control strategies. A reliable simulator is particularly important for boiler control as it is not feasible to perform preliminary tests on an actual boiler plant for safety reasons. The models described in the reviewed literature fall into two classes:

1. First principles models which are derived using the laws of physics and chemistry.

   A first principles model provides insight into the physical processes occurring within the boiler. It uses physical parameters such as mass and volume which may be obtained directly from the boiler plant. Consequently, it can be considered as a generic boiler model which may be customised to represent different boiler plants. First principles models may be used to test new control strategies or to train plant operators. First-principles models require a considerable development effort and may involve a large computational overhead.
2. Data-based models of the boiler which are generated from plant data using system identification techniques.

   Data-based models have the advantage that they may be generated far more quickly than a first-principles model and may even be generated online. However, they are not generic and do not provide physical insight into the process. The computational overhead of identifying and computing data-based model can be reduced by limiting the scope of these models. For example, a linear data-based models is capable of predicting plant behaviour at one particular operating point only. Data-based models are generally used for controller optimisation.

In some instances a mixture of first-principles and data-based equations have been used to represent the boiler plant. The data-based equations can be used to model a complex physical process such as evaporation in a more parsimonious manner than a rigorous physical description.

3.2.2 First Principles Boiler Models

One of the earliest first-principles boiler models was developed by Chien *et al* (1958). This was

an extensive model of a natural circulation boiler plant. The steam and water in the drum are not assumed to be continuously in saturated equilibrium and are modelled separately. Unlike the majority of the later models, the rate of evaporation and condensation at the drum water surface is taken into account using a heuristic equation which relates evaporation and condensation to the difference between the liquid temperature and the saturation temperature. This may not substantially improve model accuracy, however, as evaporation in the drum only accounts for a small percentage of the total evaporation, all other evaporation is assumed to take place in the risers. Steam quality is assumed to be constant throughout the risers. The model includes a superheater but does not model desuperheating (attemperation). A load model is not included, instead, a load change is equated to a change in steam flow (a model input). On the gas side, a number of simplifying assumptions have been employed. For example, the dynamics of the gas are neglected, the air to fuel ratio is assumed to be constant and the temperature of the gas entering the superheater is assumed proportional to the fuel rate. The authors make no claims as to the accuracy of the model, stating that the results indicate that it is feasible to obtain a useful mathematical model of a complicated dynamical system.

In 1970, Kwan and Anderson presented a more extensive model of a boiler plus turbine. This model includes an economiser, two superheaters, superheater attemperators, a turbine and reheaters. The gas path is divided into several sections, but similar simplifying assumptions are adopted to Chien's model. - the dynamics of the gas are neglected and the air to fuel ratio is assumed constant. The model is not compared to actual plant data, but open loop responses are given which, using physical arguments, are deemed by the authors to be correct.

The first model to be compared with actual plant data was developed by McDonald *et al* (1971). This is still one of the most extensive models of a boiler-turbine unit available to-date. It comprises 16 subsections which include boiler feed pumps, coal mills, a superheater furnace and reheater furnace in addition to the basic model components. It further differs from previous models by modelling the effect of burner tilt. Unlike the models proposed by Chien *et al* (1958) and Kwan and Anderson (1970), the vapour and liquid phases in the drum are not modelled separately. This assumes that the entire contents of the drum are in saturated equilibrium. This assumption is quite acceptable for the vapour in the drum, but seems unrealistic for the drum liquid, as subcooled liquid from the economiser is continuously mixing with the drum water. McDonald *et al* circumvent this difficulty by assuming that all feedwater flows directly into the downcomers. The energy storage capacity of the metal tubes throughout the boiler is taken into account by lumping it with the mass of the fluid in the tubes. This method is parsimonious but it does assume that the metal temperature is proportional to the fluid temperature.

Åström and Eklund (1972) adopt a combined heuristic and physical approach to modelling a drum boiler-turbine unit. Their examination of experimental data led them to use a first order nonlinear model to represent the plant behaviour in the range of half power to full power. It is assumed that the distribution of stored energy in the boiler does not change greatly over the operating range and consequently that the stored energy can be described as a function of one variable - namely drum pressure. Drum pressure is chosen as this variable because it changes significantly over the plant operating range and is also a measured variable. The model has five parameters which are estimated from data obtained using experiments on the actual plant. The results of a comparison between plant data and simulated model output are provided which demonstrate the validity of the model.

Tysso (1981) aims to develop a boiler model which has a minimum number of characteristic parameters and state variables and thus is suitable for online identification. The resulting model is parsimonious but includes important boiler dynamics. The mass flow rate in the risers is calculated using a heuristic equation which incorporates the *'Shrink and Swell'* effect into the model. In contrast to earlier models, the water in the drum is not assumed to be saturated. Comparison with closed loop plant data reveals model structural errors which are thought to be caused by neglecting the riser metal dynamics. Despite this the model is assumed to be sufficiently accurate for the purposes of controller optimisation.

The usual approach to modelling heat transfer in the evaporation systems uses the physical sections of the evaporation system as lumped systems. For example, the evaporation system may be modelled as three lumped systems - drum, downcomers and risers or alternatively as a single lumped system which includes the entire evaporation system. De Mello (1991) utilises a different approach in that he uses lumped systems which reflect the thermodynamic state of the fluid in the evaporation system. The following lumped systems are used - saturated liquid in risers and drum, subcooled liquid in risers, saturated steam in risers and drum. This approach reduces the total number of lumped systems required to model the evaporation system.

A relatively simple first-principles model of the boiler process is developed by Alatiqi and Meziou (1992). The overall performance of the model is improved however by using an empirical equation to model *'Shrink and Swell'*. The equation includes a term which may be used to correctly scale the magnitude of modelled *'Shrink and Swell'* effects.

A simple, nonlinear model suitable for the design of control strategies is the stated aim of Åström and Bell (1993). This model is based upon an earlier model by Åström and Bell (1988) and utilises the conclusions of Åström and Eklund (1972) that the essential dynamics of a boiler can

be captured in a simple model. In this instance, however, the model is derived from physical principles in contrast to the Åström and Eklund (1972) model which was developed heuristically. The resulting model has only three states - drum water volume, drum pressure and steam quality and requires just six physical parameters which can be obtained from construction data. However, it does not predict steam flow. In fact, it requires steam flow as an input to the model. This drastically limits the usefulness of this model for predicting plant behaviour. Drum level is calculated using the water volume in the drum and the density of the fluid in the risers, and thus incorporates the effect of *'Shrink and Swell'*. Validation results indicate however that the model exaggerates the effect of *'Shrink and Swell'* after changes in fuel flow.

This third order model is used as the basis for a fourth order model (Åström and Bell (1996)) which uses a different method to predict drum level. The modified model uses the volume of water in the drum and the volume of steam below the water surface in the drum to calculate drum level. The fourth state is used to model the volume of steam below the water in the drum. This model does improve the accuracy of the *'Shrink and Swell'* prediction but it is unclear why the volume of steam in the risers is no longer taken into account in the calculation of drum level.

The Åström and Bell (1993) model provides a foundation for the development of an improved model by Pellegrineti and Bentsman (1996). In particular a load disturbance is modelled as the position of an imaginary valve at the output of the boiler. This imaginary valve position effectively replaces steam flow as a model input and greatly enhances the validity of the model. In addition, a nonlinear combustion equation with a first-order lag is added to model the excess oxygen in the stack gases.

3.2.3 Data-Based Boiler Models

Data-based boiler models are used extensively for controller optimisation. Generic, linear models are by far the most common type of data-based model. The most simple model used in the reviewed literature for this task is the ARX (Auto-Regressive model with Exogenous inputs) model (Nakamura and Akaike (1981)), (Nakamura and Uchida (1989)). Nakamura's decision to use an ARX model was based on the fact that no significant improvement could be achieved using the ARMAX (Auto-Regressive Moving Average model with Exogenous inputs) model, which also required more extensive computing. The ARMAX representation (Nomura and Sato (1989)), Rovnak and Corlis (1991), Matsumura *et al* (1994) Matsumura *et al* (1993) is also used. However the most widely used data-based model of the reviewed literature is the CARIMA model (Controlled Auto-Regressive and Integrated Moving-Average Model) (Hogg and El-Rabaie (1990)), (El-Rabaie and Hogg (1991)), (Ding and Hogg (1991)), (Hogg and El-Rabaie (1991)),

(Wu *et al* (1992)), (Yang and Hogg (1992)), Zengqiang *et al* (1993)), (Manayathara *et al* (1994)), (Broderick *et al* (1995)).

Data-based models have several attractive features:

1. They do not require a substantial development effort. Model development merely involves selecting the structure of the model (e.g. ARMAX, ARIMAX), and the order of the model. Model development does not require extensive knowledge about the boiler process or the boiler construction.
2. Data-based models can be identified either offline or online using an optimisation method such as least squares. As a result, these models are suitable for adaptive controllers.
3. In general, they have few parameters and can be calculated quickly.
4. Data-based models can produce very accurate results provided that the data used to generate the models is representative of plant operating conditions.
5. Linear modelling techniques can be used to provide insight into the correct order and structure of a plant model. This information may be useful in the development of other types of data-based models such as neural network models.

Elshafei *et al* (1995) suggest the use of Laguerre filters as an alternative to the structured models, such as the ARIMAX representation. A Laguerre-filters based model consists of a first-order low pass filter followed by a large number of all-pass filters. The advantages of a Laguerre filters based model is said to be improved robustness in the presence of unmodelled dynamics and a reduction in the need for *a-priori* information. The problem with this approach however is that Laguerre filters are best suited to representing stable, well damped systems. Given that water drum level is an integrator, this approach is only suitable under controlled conditions.

There is growing interest in the use of neural network modelling - a nonlinear, data-based modelling technique. Reinschmidt *et al* (1994) develop a set of 9 SISO neural network models to model all possible input-output relationships in a boiler-turbine system with three inputs and three outputs. Brown *et al* (1994) utilise the inherent multivariable structure of neural networks by developing a single neural network model to model a boiler-turbine system with three inputs and three outputs. The ability of neural networks to model nonlinear systems constitutes a significant advantage over the structured, linear models described above.

3.3 Boiler Control

The boiler control system is defined by two design decisions:

1. Configuration of the Control System:

   This is specified by the choice of manipulated and controlled variables. If the control system comprises SISO controllers rather than a single MIMO control, the manipulated and controlled variable for each SISO controller must also be specified. This is also referred to as the boiler control strategy.

2. Choice of Control Methodology:

   The reviewed literature focused almost entirely on a relatively small number of "proven" control methodologies. This conservative approach is doubtless due to the potential volatility and inherent instability of the boiler process. The following methodologies have been investigated in the reviewed literature.

   - PI Control
   - Optimal Control
   - Predictive Control
   - Model Reference Adaptive Control
   - Several controllers which encompass more than one methodology such as a robust predictive controller or an optimal PI controller.

3.3.1 Configuration of Control System

Two common configurations of the boiler control system are described below.

1. 'Boiler-Following'

   This is an abbreviation of the phrase 'Boiler-Following-Turbine'. Where the boiler is not providing steam for a turbine, a more appropriate term would be 'Boiler-Following-Load'. In this configuration, the steam consuming process may demand more steam by opening or shutting a throttle valve at the boiler output. This causes fluctuations in boiler pressure which are removed by increasing fuel flow. In effect, the steam consuming process is permitted to draw on the boilers energy reserves at any time. The boiler control system must then act to ensure that the load demand is met and that boiler pressure is restored to the setpoint. The configuration requires that the boiler and its control system can 'follow' the demands of the steam consuming process. The system can provide good load control but it places heavy demands on the boiler. The boiler is subject to sudden pressure and temperature variations due to large load changes. In addition the performance of the system is dependent on the amount of stored energy available for meeting changes in the load demand. If the demand for steam cannot be met by the boiler, the quality of the steam deteriorates.

2. 'Load-Following'

   This configuration uses the throttle valves as a means of controlling boiler pressure (rather than steam flow). The boilers inputs - fuel, air and feedwater flow rate - are set at a value which will generate the required amount of steam. If more steam is required the boiler inputs are increased as appropriate. When the boiler pressure starts to increase (as a result of the increased fuel flow), the throttle valves are opened more fully and the extra steam is provided to the steam consumer. This method is preferable with respect to the boiler because it eliminates pressure and temperature fluctuations due to sudden load changes. However, spurious variations in steam flow are passed to the steam consuming process. In the case of a turbine, for example, this causes grid frequency disturbances.

The difference between these two configurations can be summarised in terms of the controlled variables: In the 'Boiler-Following' configuration the boiler control system controls steam pressure and the steam consuming process controls steam flow. In the 'Turbine-Following' configuration the steam consuming process controls steam pressure and the boiler control system controls steam flow.

In addition to steam pressure or steam flow, a boiler control strategy must also control the following variables:

- Drum Level
- Main Steam Temperature
- Percentage of O_2 in the stack gases
- Furnace Pressure
- Reheat Steam Temperature (if a reheater is present)
- Reheat Steam Pressure/Reheat Steam Flow (if a reheater is present)

3.3.2 PI Control

A conventional boiler control scheme is described fully in a Rosemount publication, which describes control of drum water level, steam pressure, percentage O_2 and furnace pressure. Each variable is controlled using a SISO controller, with the inclusion of feedforward control signals where appropriate.

Drum water level can be controlled by a SISO controller which uses feedwater flow as the manipulating variable. This simple strategy, referred to as single element control, is shown in Fig. 3.1.

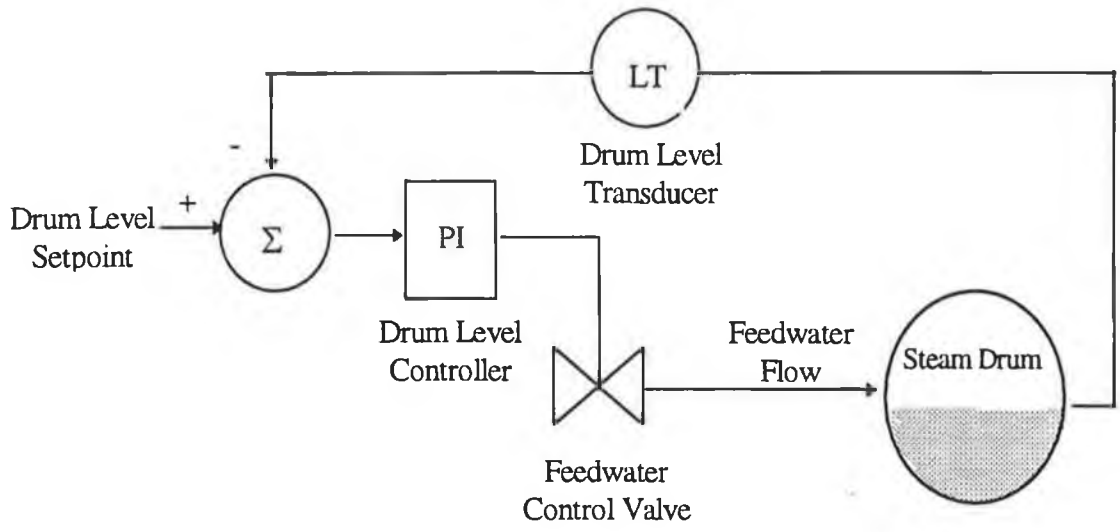


Fig. 3.1 Single Element Drum Level Control

This scheme is considered to be inadequate by Rosemount however because drum level is a slow-responding variable which integrates the effect of variations in drum pressure, feedwater flow or steam flow. Westinghouse (1980) also state that because of the *'Shrink and Swell'* phenomenon, this method is unsatisfactory if the boiler to be controlled is subject to sudden changes in load. In the event of a sudden increase in load for example, drum pressure decreases causing the density of the fluid in the drum to decrease and drum water level to increase (swell) for a short time. If single element control drum level control is employed, the drum level controller will reduce or shut-off feedwater flow at this point. When pressure equilibrium is re-established in the boiler, the density of the fluid in the drum decreases and drum water level falls (shrink). As water level has been reduced or even cut-off, a danger exists that drum water level may drop to a dangerously low level. An enhancement to the single element control scheme is to incorporate steam flow as a feedforward element. This ensures that when steam flow is increased that water flow is also increased and effectively allows the controller to react more quickly to changes in steam flow. The enhanced strategy, referred to as two element control, is shown in Fig. 3.2.

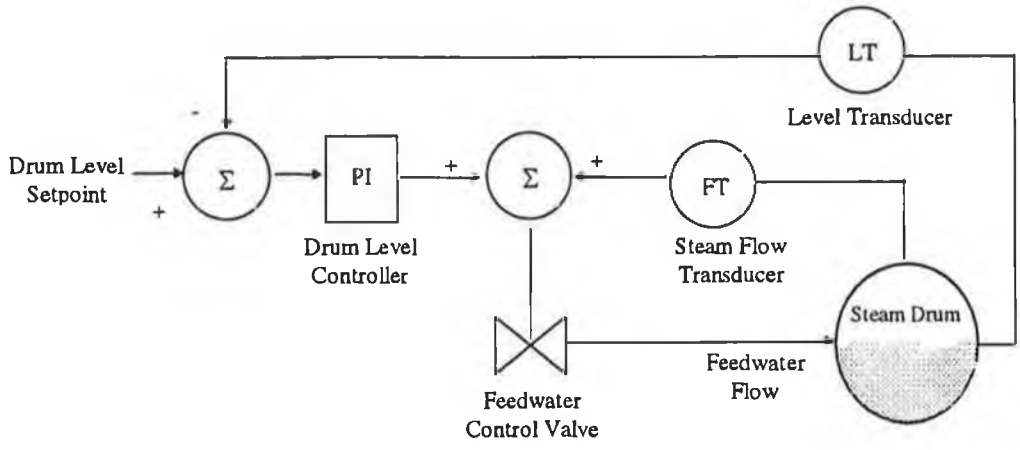


Fig. 3.2 Drum Level Control with Feedforward Steam Flow Signal

Inclusion of steam flow as a feedforward signal demands that the steam flow transducer is carefully calibrated with respect to the feedwater control valve. This strategy also has the disadvantage of exaggerating the effect of *'Shrink and Swell'* in the drum. If load demand increases, this causes an increase in steam flow which results in a reduction in drum pressure. This reduction in drum pressure allows the water bubbles in the boiler to expand, and consequently causes the drum water level to rise. However, the increase in steam flow increases the feedwater flow *via* the steam flow feedforward signal, thus raising the drum water level even higher.

According to Rosemount, the problem of *'Shrink and Swell'* can be overcome by using the output of the drum water level controller (including the steam flow feedforward signal) to act as the setpoint to a feedwater flow controller. This configuration, referred to as three element control is shown in Fig. 3.3. It is not clear, however, why this modification should have a significant effect on *'Shrink and Swell'*. For this strategy, the steam flow transducer must be calibrated with respect to the feedforward flow transducer.

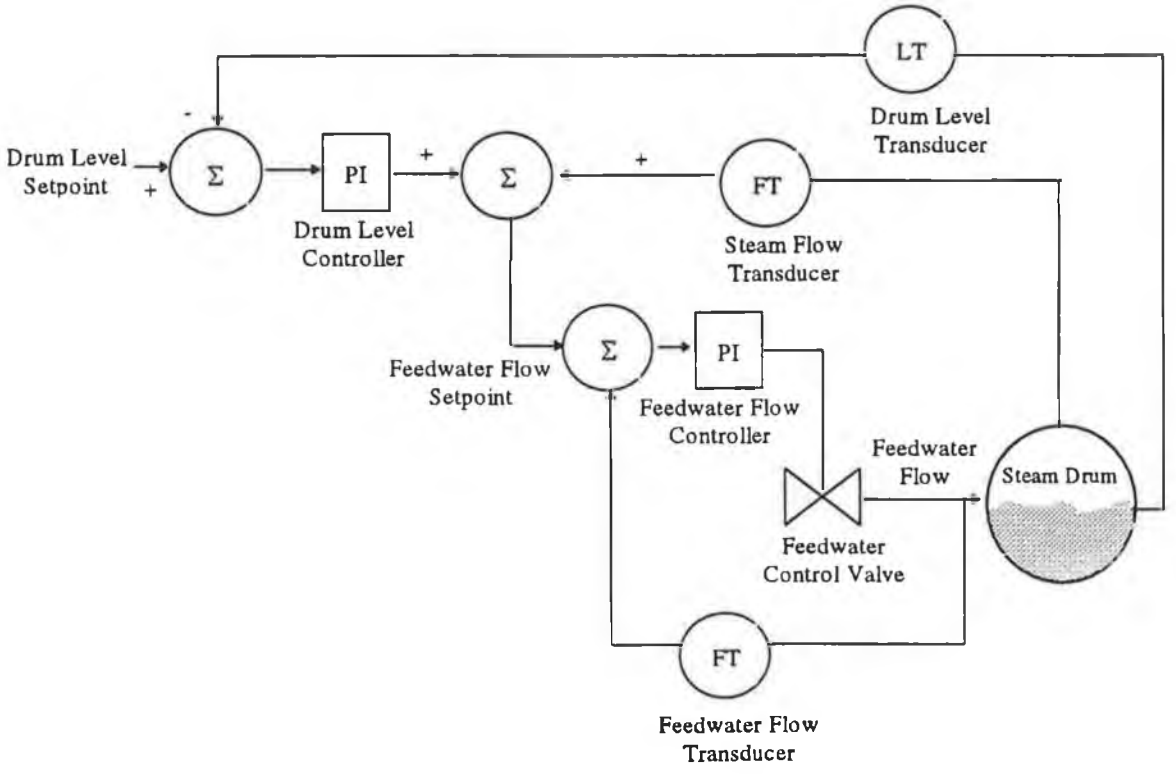


Fig. 3.3 Drum Level Control including Feedforward Steam Flow Signal and Feedwater Flow Controller

Rosemount suggests that steam pressure can be controlled using firing rate with steam flow as a feedforward signal, as shown in Fig. 3.4

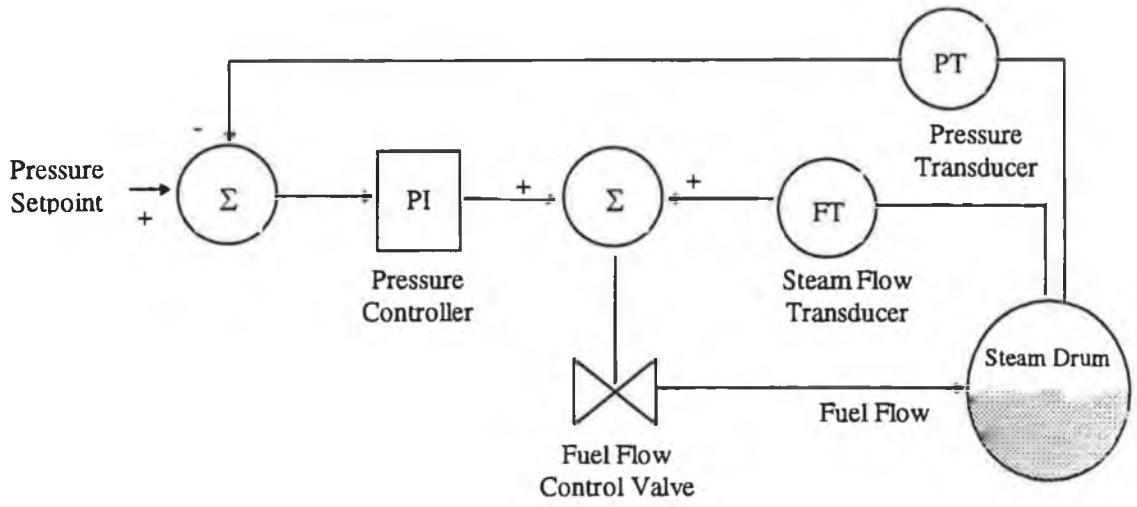


Fig. 3.4 Steam Pressure Control

Air flow is primarily dictated using the fuel flow signal and a pre-specified air/fuel ratio curve. Air flow control can be refined by adding a trim air controller which attempts to maintain the percentage of measured O_2 in the stack gases at its setpoint. Air flow through the furnace is regulated by varying the position of the forced draft fan damper. The suggested control configuration is shown in Fig, 3.5.

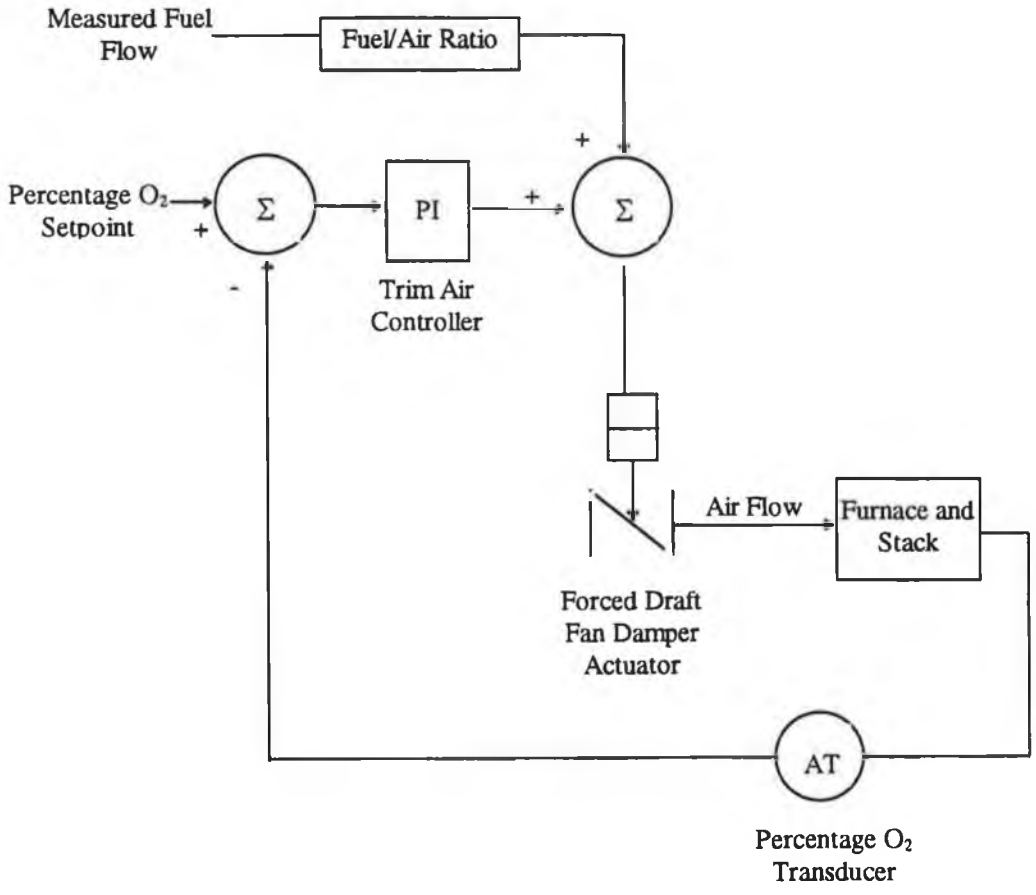


Fig. 3.5 Percentage O_2 Control

Furnace pressure is regulated using the induced draft fan. A feedforward signal from the forced draft fan damper position may be added to increase the speed of response of the furnace pressure control system to changes in air flow. This control configuration is shown in Fig 3.6.

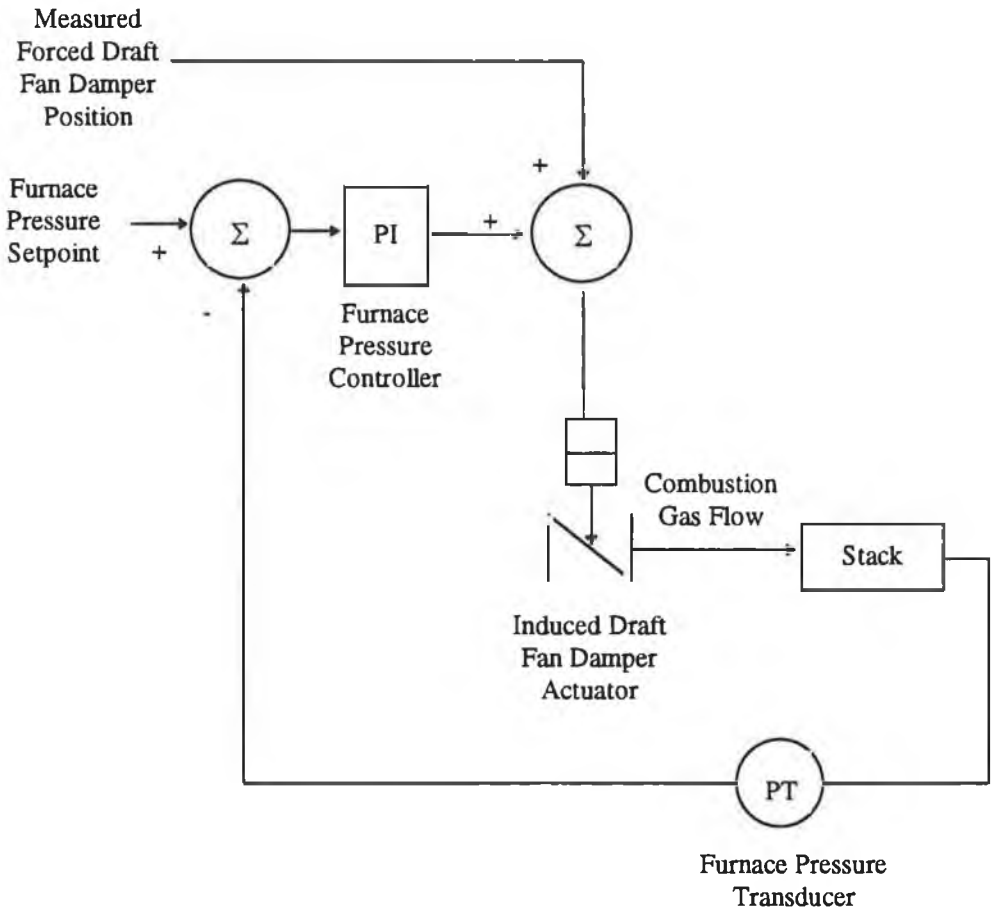


Fig. 3.6 Furnace Pressure Control

One of the main problems associated with PI control is the difficulty of choosing suitable controller parameters. Unbehauen (1969) estimates optimal controller parameters for a steam temperature controller by minimising a quadratic cost function which penalises tracking error. He concludes that both the range of stable controller settings and the optimal controller settings are dependent on load. He further concludes that steam temperature control is strongly affected by the other control loops and should not be regarded as a single, isolated loop. These conclusions highlight two shortcomings of standard boiler PI control schemes, similar to the one described above. Firstly, such schemes do not take into consideration the variations in process dynamics with load. The controllers may be well tuned at one particular operating point, but controller tuning at other operating points may be very poor and even lead to instability. Secondly, the interaction between the loops is largely ignored (with the exception of several feedforward control signals).

One means of overcoming the first problem is to employ an adaptive PI control scheme. One such scheme, proposed by Sato *et al* (1984) for steam temperature control identifies the process dynamics online assuming a linear physical model of the superheater. The system characteristic equation is derived and the proportional and integral gains are adapted so as to yield specified closed loop stable poles. The states of the process are estimated using a Kalman filter and control is further improved by using the multi-step prediction of the plant output as an input to the controller instead of the actual plant output. The scheme is implemented on both a once-through and drum-type boiler and reduces temperature deviations from the set-point by 50% compared with a non-adaptive control scheme.

An alternative method of adapting the parameters of a PI controller of main steam pressure is suggested by Sun Demin *et al* (1989). The process is represented by an ARMAX model, whose parameters are identified online using an extended least square algorithm with a forgetting factor. The controller parameters are determined by minimising a quadratic cost function which penalises tracking error and weighted control increments. A further supervisory level is included which checks parameter estimation convergence and stability of the process poles. The scheme is implemented on a 20,000 kg/hr industrial boiler and demonstrates that the main steam pressure remains constant in the presence of external disturbances.

The controllers developed by Unbehauen (1969), Sato *et al* (1984) and Sun Demin *et al* (1989) are optimal in the sense that the parameters of a PI control scheme have been optimised with respect to a specified cost function. These controllers are not optimal, according to the accepted definition of "optimal" however, in that the structure of these controllers is predefined and possibly sub-optimal.

3.3.3 Optimal Control

There has been considerable interest in the application of optimal control methodologies to boiler control. This interest is partially motivated by the multivariable nature of boiler control, which can be systematically handled using optimal control. It is also motivated by the potential of optimal control to provide a systematic method of controller tuning.

An early application of optimal control for the control of drum level, steam pressure and steam temperature is described by Nicholson (1964). A linearised model of a physical boiler model is reduced by inspecting the eigenvalues and eliminating the fast modes. The reduction has the undesirable consequence of eliminating the non-minimum phase effect of the drum water. A cost function which penalises displacement of the system states and controller action is defined. For comparative purposes, the cost function is minimised both sub-optimally and optimally. The sub-

optimal controller optimises the cost function with respect to each of the manipulated variable individually. Neither the optimal nor sub-optimal controller, however, is capable of controlling the non-minimum phase response of drum water. Stability may be ensured however by using a Lyapunov function in the criterion or by including additional integrating control. Both the sub-optimal and optimal controller produced better results than a conventional PI control scheme. The optimal controller did not produce superior results than the sub-optimal controller however, indicating that the extra computational effort involved may be unjustified.

Nicholson's positive results were contradicted to some extent by results obtained by Anderson (1969), which indicated that integrated optimal control schemes did not significantly improve the performance of the unit under consideration.

McDonald and Kwatny (1973) felt that neither Nicholson's nor Anderson's plant models may have adequately represented a typical boiler plant and as such both results were open to criticism. They developed an optimal controller to control power, pressure, main and reheater temperature and steam flow minus feedwater flow plus a coefficient of drum level error. The latter controlled variable is an interesting way of including steam flow as a feedforward signal, but it does require that both steam flow and feedwater flow are both accurately calibrated. The optimal controller is based upon a linear model obtained at a single operating point, which includes a noise vector. The plant states are estimated using the actual control inputs which is an important feature if the system controls saturate. The cost function which penalises tracking error and controller action is solved offline using a Ricatti equation. The resulting state variable feedback control law is tested on an extensive nonlinear plant model (14 states) and produces a significant improvement over conventional PI control. This improvement is attributed to the co-ordination of the manipulated variables which resulted from the multivariable approach.

McDonald and Kwatny's positive experience of optimal control has since been repeated by many successful implementations of optimal controllers on boiler plants. One of the first implementations was carried out in 1978 on a 500 MW plant in Japan, where there has been a consistently strong interest in optimal control of boilers. Nakamura and Uchida (1989) describe the control system configuration and details of an optimal regulator which is employed by five plants. The controller is based upon a multivariable ARX model which is identified offline and converted into state space format. The cost function which penalises state variable error and controller action is minimised using dynamic programming. The controller differs from McDonald and Kwatny's optimal controller in that it attempts to overcome the restrictions of applying a linear control methodology to a nonlinear plant in two different ways. Firstly, it adapts the control parameters according to load by obtaining linearised models of the plant at two

or three different loads. Secondly, it generates feedforward signals which reduce steam temperature deviations after a load change. The appropriate feedforward signals are calculated using step response models, identified from plant data. Selection of the cost function weighting matrices was deemed to be the most elaborate part of the regulator design. Plant operation using the optimal regulator shows a considerable control improvement in control performance over conventional PI control. At the time of publication, the optimal regulator had been in operation for over ten years, without any readjustment of the controller parameters. The authors stress however, that the plant is oil-fired and as such changes to the process dynamics due to ageing of the boiler or fouling or tubes may be less than that of a coal fired boiler.

The problem of process parameters varying either with time or load variations may also be dealt with using an adaptive approach. Fessl (1986) developed a self-tuning linear quadratic regulator to control steam temperature in two linked superheaters and to control steam pressure. The steam temperature in the two linked superheaters was controlled using either a single MIMO controller or two SISO controllers. The MIMO controller achieved better results. The model parameters are identified on-line using a recursive square root version of the least squares method. This scheme was successfully verified experimentally. An adaptive approach, albeit quite restrictive, has also been implemented on a boiler by Nomura and Sato (1989). Nomura and Sato do not permit parameter identification during large changes of load. Parameters identification may only occur during periods of either high or low load when load fluctuations are small. Model parameters and control gains at intermediate load levels are obtained by interpolating between the values determined at high and low load levels. Results demonstrate a significant improvement over conventional PI control.

It is clear from the literature that there has been a concerted effort to take advantage of reliability of conventional analog control where possible. For example Nomura and Sato (1989) treat the plant and its conventional control object as a single controlled object, allowing the conventional control scheme to continue operating if the optimal controller fails. Another scheme (Nakamura and Uchida (1989)) simply sums the output of the optimal control scheme with that of the conventional PI control scheme.

In a similar manner, some schemes try to include as much as possible of tried and tested conventional control wisdom into the new and untried optimal controllers. Cori and Maffessoni (1984) developed a "practical-optimal" controller which incorporates two of the features of conventional control - proportional output feedback and a feedforward steam flow signal which partially decouples drum level control from load variations. The controller is tested on a boiler and produces reasonable results. Anakwa and Swamy (1985) also develop an output feedback

optimal controller which is tested on a boiler plant model (six states) and compares favourably with state- and observer- based controllers.

Fundamental to the successful implementation of a model-based controller is the determination of an adequate plant model. Fessl (1986) compares the performance of an optimal controller which uses parameter estimates to that of an optimal controller which uses parameter estimates and their uncertainties. The latter (cautious) control proved to be the more successful scheme. More recently Pellegrinetti *et al* (1991) incorporated model uncertainty by implementing a H_∞ optimal controller. A H_∞ controller aims to find a feedback gain matrix which ensures that the infinity norm of the closed loop transfer function is below some prespecified bound for a nominal plant plus specified uncertainties. One of the cited advantages of this control method is that it permits frequency based control and output weights unlike standard state space LQ control which uses constant spectrum weightings. However, frequency based weightings in this instance did not successfully reduce high frequency control action. The problem of high frequency controller action was solved by filtering the plant measurements before sending them to the controller. This solution could not be applied to the oxygen level signal however, as the oxygen level dynamics are in the same frequency band as the noise. Simulation results using a three state plant model are promising. This work has also been extended to take into account the effect of delays. (Pellegrinetti and Bentsman (1992)).

In summary, there have been several successful applications of optimal control strategies to the boiler process, both in simulation and on actual plants. Much of the motivation for the application of optimal control strategies appears to lie in its multivariable approach. Indeed, the improvement in control performance is generally attributed to the multivariable approach. The performance of the controller seems fairly dependent on the choice of the weighting matrices in the cost function. Little information is provided on how weights are obtained, which suggests that the process of weight selection is largely experimental.

3.3.4 Predictive Control

There has been a recent surge of interest in the application of predictive methodologies to boiler control. There does not seem to be any one dominant factor which motivates this interest, rather, the suitability of predictive control for many types of control problems. For example, the basic predictive methodology can be applied to multivariable control problems, self-tuning or adaptive control schemes, nonlinear control schemes and constrained control schemes.

The performance of a predictive controller is quite dependent on the ability of the internal controller model to adequately represent the boiler plant. The most popular internal controller

representation is the CARIMA model (Hogg and El-Rabaie (1990)), (El-Rabaie and Hogg (1991)), (Ding and Hogg (1991)), (Hogg and El-Rabaie (1991)), (Wu *et al* (1992)), Zengqiang *et al* (1993)), (Manayathara *et al* (1994)) and Broderick *et al* (1995)). This model is used to derive controllers with inherent integral action, which are capable of automatically rejecting step disturbances. The model structure is described in Section 7.2.1.

There is also considerable uniformity in the cost function definition of all the predictive controllers. The cost function is generally a quadratic function of set-point tracking error. In some papers, the setpoint is filtered to reduce the effect of large initial deviations from the set-point (Wu *et al* (1992)) and (Zengqiang *et al* (1993)). An output weighting matrix may also be included to smooth the output response (Zengqiang *et al* (1993)) and (Manayathara *et al* (1994))

The more usual method of smoothing plant response is to penalise increments in control output in the cost function (Ding and Hogg (1991)), (Wu *et al* (1992)) and (Zengqiang *et al* (1993)) and (Manayathara *et al* (1994)), or simply controller action (Fujiwara and Miyakawa (1990)). Inclusion of such terms implies that a suitable controller action weighting matrix must be determined. In all the reviewed publications this has been obtained on a trial and error basis.

A stabilising predictive control which ensures stability of the closed loop system for arbitrary cost functions is derived by including the terminal state in the cost function (Manayathara *et al* (1994). The terminal state is equal to the value of the state at the end of the prediction horizon. Simulation results exhibit good tracking performance regardless of the length of the prediction horizon.

A robust predictive controller is developed Broderick *et al* (1995) which uses a mixed H_2/H_∞ cost function. This cost function penalises the peak of the finite sum of the disturbance spectrum to achieve good stability and robustness (as for H_∞ control) while minimising the reference spectrum to obtain good tracking properties (as for H_2 control). A predictive control strategy was chosen in preference to the standard state-space H_∞ methods, because it is hoped to incorporate constraints into the design at a later stage. This is more easily facilitated using a receding horizon strategy like predictive control. Simulation results validate the robustness of this controller - the closed loop system under robust control remains stable at all setpoints whereas it displays instability at lower setpoints when controlled by a Generalised Predictive Controller.

Many of the designs exploit the ability of predictive control to handle multivariable systems. Rovnak (1991) designs a multivariable controller of steam pressure, steam temperature and power which is based upon a matrix of SISO step response models. The models are identified offline, each model representing the relationship between a single controlled and manipulated

variable. The controller is tested on a model of a once-through boiler and performs well. This multivariable predictive control approach, which is based upon SISO step response models, is referred to as Dynamic Matrix Control (Cuttler and Ramaker (1980).

Rossiter (1991) takes a different approach to the design of a multivariable generalised predictive controller for the control of power and pressure on a boiler-turbine unit. A linearised model of the plant is decomposed into its eigenvalues/eigenvectors. This allows the definition of separate output and control horizons for each characteristic subsystem. It also greatly reduces the optimisation computational overhead by converting a vector optimisation problem into a scalar one. Rossiter also tackles the problem of controlling a stiff system with a multivariable controller by proposing that the future control increments for the slow system and fast system are permitted to change at different frequencies. Simulation results on an 80 state physical model demonstrate an improvement over PI control.

Predictive control is also very suitable for application in adaptive control strategies. Hogg and El-Rabaie (1990) describe an adaptive generalised predictive controller to control steam pressure in a drum boiler. A multi-loop adaptive control scheme of superheater steam pressure, superheater steam temperature and reheater steam temperature controller has also been developed (Hogg and El-Rabaie (1991)). An adaptive predictive steam temperature controller is developed by Fujiwara and Miyakawa (1990).

This adaptive work has been extended to several multivariable control schemes - a multivariable power and pressure controller (El-Rabaie and Hogg (1991)) and a multivariable power, drum steam pressure and drum water level controller (Ding and Hogg (1991)). In each case, the internal controller model is a linear autoregressive model, whose parameters are identified using a recursive least squares algorithm, a constant forgetting factor and conditional interruption of identification in the case of very large or very small prediction errors. In all cases, simulation results demonstrate a big improvement over conventional PI control.

Wu *et al* (1992) develop an interesting variant of adaptive generalised predictive control, based upon the requirement that each control increment must be proportional to the previous control increment. This controller is denoted "stair-like generalised predictive control" on the basis that if the constant of proportionality is set equal to one, each control increment will equal the previous, resulting in a stair-like projected control sequence. The advantage of this adaptation is that the control law does not require matrix inversion, which is computationally intensive and not guaranteed to exist.

The robustness of adaptive controller in the presence of the noise and the need for *a-priori*

information have been identified by research as two weak points of adaptive controllers, according to Elshafei *et al* (1995). Research has further identified that the use of structured models such as the ARMAX model may be the source of these problems. Elshafei *et al* suggest using a truncated Laguerre series as the internal controller model i.e. a first-order low pass filter followed by a number of all-pass filters. The orthonormality of the filters ensures that identification yields an unbiased estimate of the plant in the presence of unmodelled dynamics and coloured noise. The filter parameters are identified using a recursive least squares algorithm. Good simulation results are shown but a description of the plant model is not provided.

Nonlinear control is an alternative means of coping with the variation in plant dyamics with load. Gibbs and Weber (1992) propose a nonlinear model predictive controller for the control of seven plant variables based upon a reduced order physical model. The states of the model are estimated in real-time using Kalman filtering and the cost function is optimised. It is hoped that the use of a model based upon first principles mass and energy balances should allow on-line calibration of sensor biases and internal furnace parameters. The controller is tested successfully on a model of a supercritical boiler plant.

A continuous-time generalised predictive controller for the control of steam pressure has been developed by Ping Yang and Hogg (1992). Continuous-time control has a number of advantages - control variables are updated continuously (sampling periods of 30s are commonly used in boiler control) and continuous control will eliminate the problem of sample rate selection in multivariable controllers of stiff systems, such as boilers.

The predictive control methodology incorporates the essential advantages of optimal control, such as the ability to handle multivariable systems and an "optimal" method of tuning with respect to a given cost function. It has also been applied successfully to the boiler control process, yielding results which are superior to those of conventional PI control. It is clear from the literature review however that flexibility is one of the primary strengths of predictive control. A large variety of predictive controllers have been reviewed, including adaptive, multivariable, robust, continuous and nonlinear controller. For example, a predictive control framework is used to implement a robust control strategy on the basis that constraints could be more easily included into the predictive control design than into the design of a state-space, finite horizon controller.

3.3.5 Model Reference Adaptive Control

There has been little interest in the application of model reference adaptive controllers to boiler control. Man Gyun Na and Hee Cheon No (1990) propose a Model Reference Adaptive System (MRAS) for water level control, Fernandez-del-Busto *et al* (1985) for main steam pressure,

Uchida *et al* (1988) for main steam pressure, Åkesson (1987) for steam temperature control and Matsumura *et al* (1993), (1994) for steam temperature control.

3.4 Conclusions

- Most research into advanced boiler controls is based upon a relatively small number of first-principles boiler models. Most of these models are too simple to be of real benefit in assessing a control strategy. For example, it is common to neglect combustion chamber dynamics. It is also frequently assumed that all the feedwater is transferred directly to the downcomers. These simplifying assumptions may have been motivated by a need to reduce simulation time. It is now realistic to develop more complex and accurate models however.
- Neural network models have been demonstrated to be a viable alternative to linear data-based models.
- Boiler control presents problems which cannot be adequately dealt with using a linear, SISO control strategy. In particular, these include interactions between the controlled variables and varying process dynamics.
- The issue of interaction between the controlled variables can be addressed using a multivariable approach which takes into account interactions between the controlled variables. An adaptive or self-tuning approach may also be implemented which takes into account variations in process parameters. The adaptive approach involves online system identification of a linear model. This is only possible in the case of small perturbations around a steady state operating point. It may be necessary to implement a higher level of control which supervises model identification. An alternative approach is to use a nonlinear model of the boiler process.
- Optimal and predictive control are the most popular boiler control strategies for a number of reasons. Both approaches are model based, can be applied to multivariable control, can be implemented adaptively, and provide a mechanism for systematic controller tuning.
- Currently, there is greater interest in the implementation of predictive boiler control strategies than optimal boiler control strategies. This seems to be motivated by the flexibility of the predictive control methodology. Predictive controllers have the capacity to react pre-emptively to known, future changes in setpoint. It is easier to incorporate system constraints into a predictive control strategy than an optimal control strategy.

4. Development of an Analytical Boiler Model

4.1 Introduction

The function of a boiler is to generate steam, continuously and in large quantities. Steam generation is not a single physical process. It involves a large number of diverse sub-processes, such as combustion, fluid flow, heat transfer and evaporation. In order to model the operation of a boiler, each of these sub-processes must be modelled by applying the appropriate analytical equations. The resulting set of interacting, analytical, equations constitutes a mathematical boiler model.

This exercise has obvious benefits. Firstly, the task of relating the physical processes to the appropriate equation provides enormous insight into the boiler behaviour. Secondly, the resulting model can be used to investigate the behaviour of the boiler process under any condition. The model may also be used for the investigation of new controllers, boiler designs or the possibility of design for control.

The value of these investigations is highly dependent on the accuracy of the model. If the model does not accurately represent the boiler process, the value of the investigations is reduced. The accuracy of the model is largely determined by the type of simplifying assumptions used in the model. The simplifying assumptions adopted should be dictated to a great extent by the intended function of the model. The model described here is to be used for controller design. Consequently, it was felt that all the dominant effects should be included and that the minimum of simplifying assumptions should be used. All such assumptions have been detailed in full.

The final model is comprised of twenty first order equations, and a very large number of fluid flow, heat transfer, thermodynamic state and chemical equations. It was a significant task to test the software implementation of a model of this size. A systematic testing procedure was devised, which ensured that the software implementation was error free.

Ultimately, the validity of the model can only be measured by comparison with an operating boiler. Validation results, which demonstrate the validity of the model, are provided at the end of this chapter.

4.1.1 Model Structure

Steam generation requires two simultaneous processes - combustion and evaporation. These two

processes are modelled using very different methods. This is reflected in the description of model development, which is divided into two sections describing:

1. Development of combustion-side model
2. Development of fluid-side model

The description of the fluid-side and of the combustion-side of the model are, in turn, partitioned into several smaller sections. The sections used are illustrated in Fig 4.1.

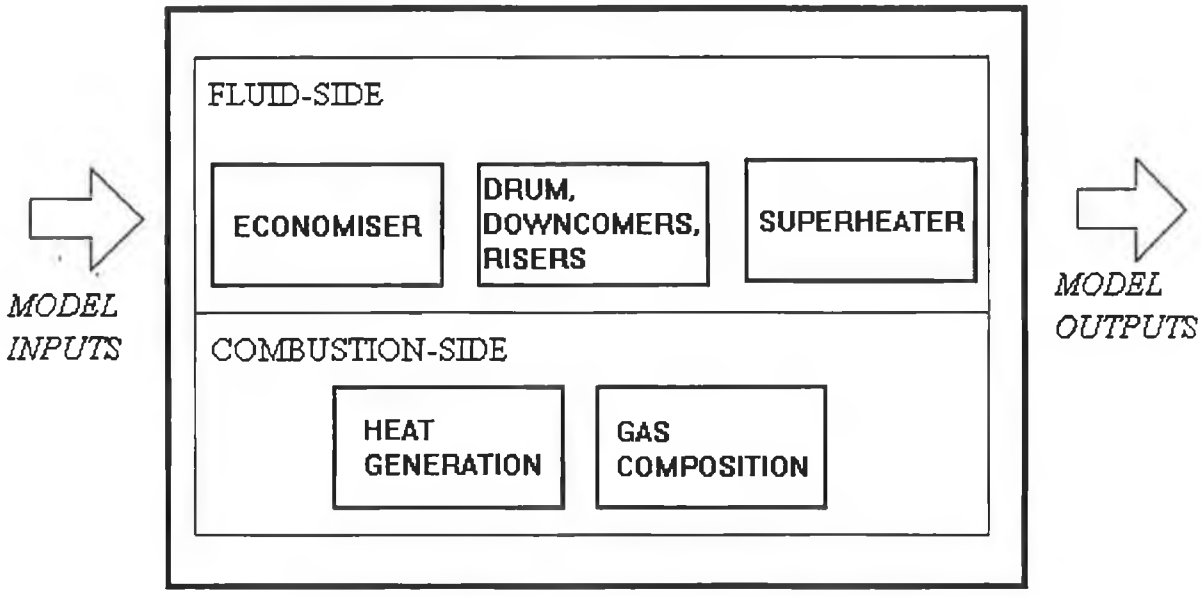


Fig. 4.1 Boiler Model Components

Each section of the model may be comprised of a number of subsystems, called lumped systems, which are the fundamental model components. Each lumped system models a region of the boiler, within which it is assumed that all the properties of that region are uniform. For example, the economiser metal tubes can be treated as a lumped system if it is assumed that the temperature of the tubes is uniform throughout the economiser. A distributed parameter model would not require such an assumption but it would be extremely complex. It is unlikely that the possible improvement in model accuracy would compensate for the extra development and implementation effort. A schematic representation of all the lumped systems and their interfaces is given in Fig. 4.2.

This chapter describes each section of the model in considerable detail. A complete list of the model inputs, state variables and outputs is provided in Fig. 4.3. Fig 4.1, 4.2 and 4.3 provide an overview of the full model.

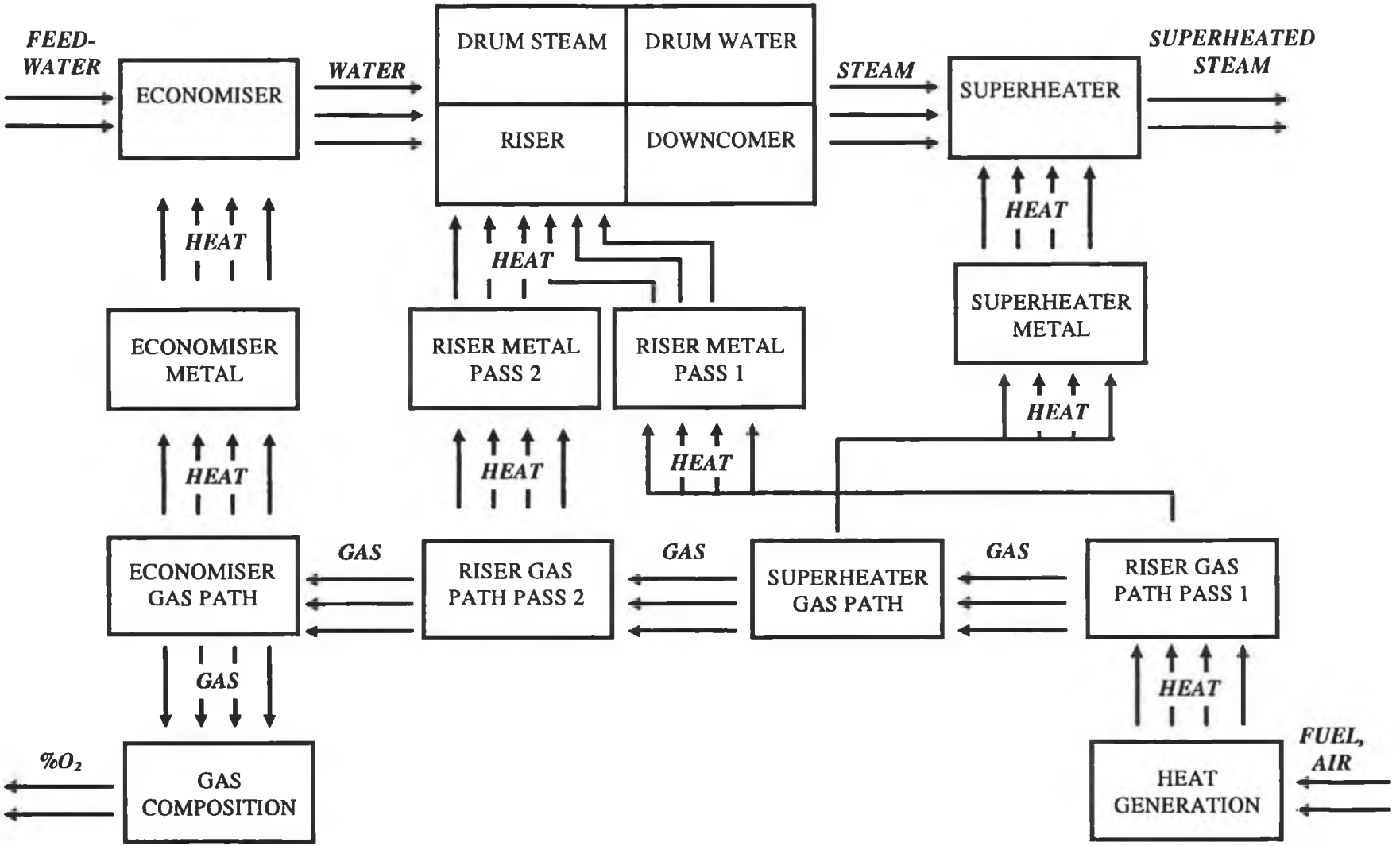


Fig. 4.2 Schematic of Lumped Systems

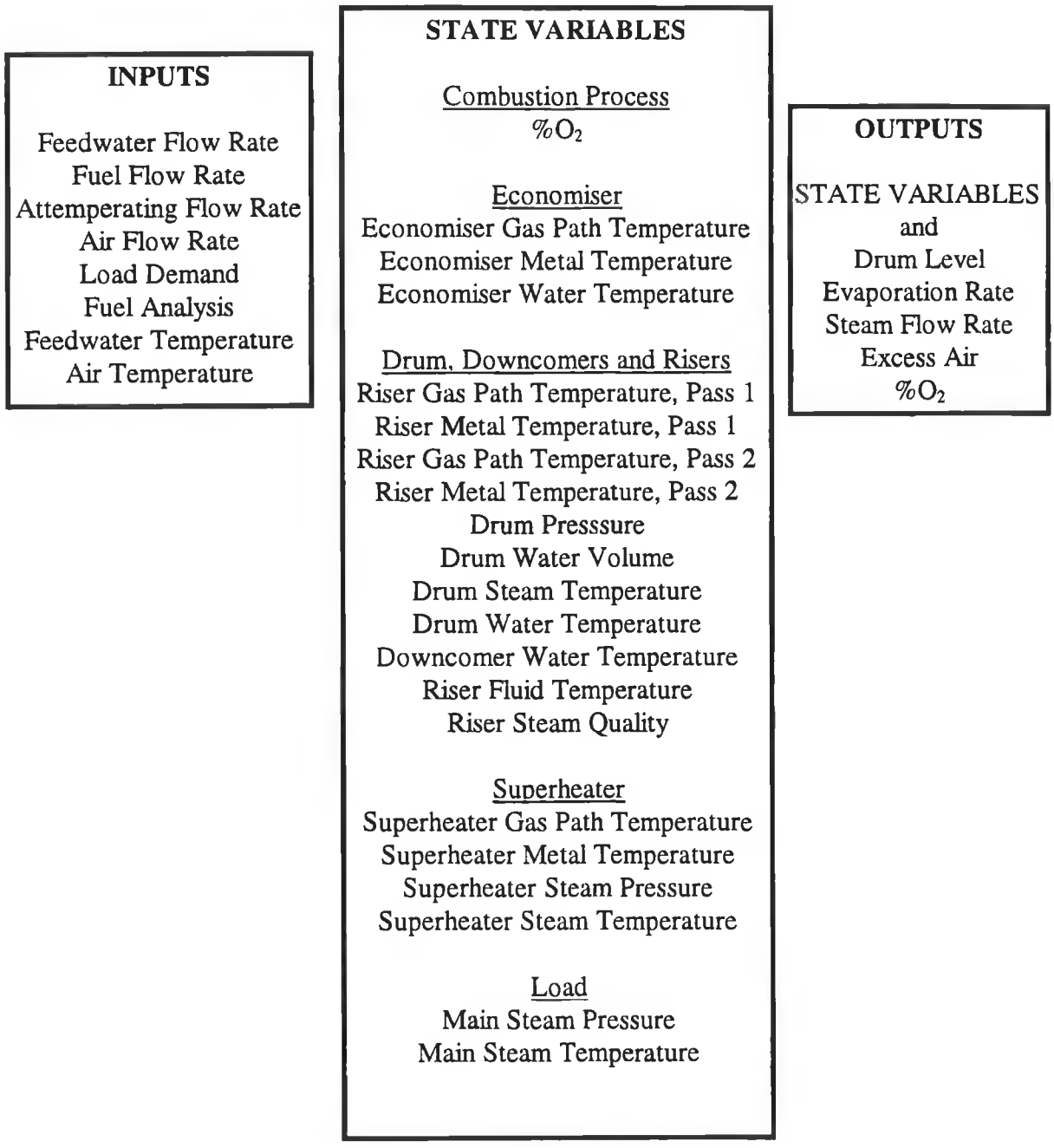


Fig. 4.3 Model Inputs, Outputs and State Variables

4.2 Development of Combustion-Side Model

4.2.1 Overview

There are two distinct aspects to the combustion-side model - heat generation and combustion gas composition.

- Heat Generation

  The heat generation model is required to calculate how much energy is available to the boiler for steam generation. The motivation for modelling heat generation is obvious - the amount of energy available for steam generation is dependent on the amount of energy released by combustion.

- Combustion Gas Composition

  The combustion gas composition model calculates the contents of the combustion gases as they leave the furnace. It is subdivided into a description of the composition of the products of gaseous fuel combustion and a description of the composition of the products of solid and liquid fuel combustion. Each of these sections is further subdivided into a description of stoichiometric and sub-stoichiometric combustion modelling. The section also describes how the gas composition measurement dynamics are modelled.

  There are two principal incentives for modelling the composition of the combustion gases. Firstly, the model can be used to investigate combustion efficiency and means of improving it. Combustion efficiency is decreased if an excess of air is available for combustion as this excess air must be heated up to combustion chamber temperatures. In this case the percentage of oxygen in the combustion gases will be greater than zero. If the amount of air available for combustion is less than or equal to the amount of air which is required for complete (stoichiometric) combustion of the fuel, the percentage of oxygen in the combustion gases will equal zero.

  Secondly, the model can be used to investigate the amount of toxic and environmentally damaging gases which are generated by a boiler. Carbon monoxide, which is poisonous, is produced if the amount of oxygen available for combustion is insufficient to burn all the fuel (sub-stoichiometric combustion). For this reason, it is important to keep the percentage of oxygen in the combustion gases above zero. Carbon dioxide, oxides of sulphur and oxides of nitrogen which are damaging to the environment are also produced by combustion.

4.2.2 Heat Generation

The heat which is generated in the combustion chamber has three sources:

1. Combustion of fuel
2. Heat contained in the incoming air
3. Heat contained in the incoming fuel

Assuming stoichiometric combustion, heat generation can be modelled, on a macroscopic level, using the following equation:

$$Q_{cc} = C_{fu} m_{fu} + m_{fu} c_{fu} T_{fu} + m_{air} c_{air} T_{air}$$

where

c_{air} = specific heat of air
c_{fu} = specific heat of fuel
C_{fu} = calorific value of fuel
m_{air} = mass flow rate of air
m_{fu} = mass flow rate of fuel
Q_{cc} = heat generated by combustion
T_{air} = temperature of air
T_{fu} = temperature of fuel

(4.1)

The dynamics of the combustion process are extremely fast in comparison to the dynamics of evaporation. For the purposes of modelling, the dynamics of the combustion process have been neglected.

Most sections of the boiler - economiser, downcomers and superheaters, are heated by convection i.e. by hot flue gases flowing over them. The mass flow rate of the combustion gases is needed to calculate how much heat is transferred to these sections of the boiler.

For gaseous fuels, the mass flow rate of gas through the furnace is found by summing the mass flow rate of the incoming fuel and air.

$$m_g = m_{air} + m_{fu}$$

where

m_{air} = mass flow rate of air (kg / s)
m_{fu} = mass flow rate of gas (kg / s)

(4.2)

For solid and liquid fuels, the mass flow rate of gas can be assumed equal to the mass flow rate

of the incoming air only.

$$m_g = m_{air} \tag{4.3}$$

4.2.3 Combustion Gas Generation

Simple chemical equations are used to model the composition of the combustion gases. The combustion equations of a fuel are used to calculate both the oxygen requirements for stoichiometric combustion, and the type and quantity of gases which are produced by combustion. The constituents of the fuel must be provided as inputs to the model.

The method of modelling is slightly different for gaseous and non-gaseous fuels. The procedure for gaseous fuels will be described first.

The primary reference for this section is Hanby (1994).

4.2.3.1 Gaseous Fuels

Gaseous fuels, such as natural gas may be composed of several different types of gas. For example the two largest components in a typical analysis of Kinsale Natural Gas are:

Methane (CH_4):	99.15%
Ethane (C_2H_6):	0.30%

Firstly, the composition of the combustion gas produced by each type of gas in the fuel is calculated. The composition of the fuel's combustion gas is then found on a proportional basis. For example, given that the following chemical reaction equations are available for CH_4 and C_2H_6:

$$1 \text{ vol. } CH_4 + x1 \text{ vol. air} \rightarrow y1 \text{ vol. } CO_2 + z1 \text{ vol. } H_2O$$
$$1 \text{ vol. } C_2H_6 + x2 \text{ vol. air} \rightarrow y2 \text{ vol. } CO_2 + z2 \text{ vol. } H_2O$$

Using the typical gas analysis:

1 vol. fuel requires:	0.9915x1 + 0.003x2 vol. air
1 vol. fuel produces:	0.9915y1 + 0.003y2 vol. CO_2

Stoichiometric Combustion

Stoichiometric combustion occurs if the volume of air available for combustion is equal to or

exceeds the volume of air required for combustion.. This means that all of the available fuel can be fully burned. For each component of the fuel, the same procedure is used to calculate the volume of air required for stoichiometric combustion and to calculate the composition of the combustion gases. Consequently, the procedure shall be explained for methane only. The stoichiometric combustion reaction for 1 volume of methane is:

$$\begin{aligned} CH_4 \; + \; 2O_2 \; &\rightarrow \; CO_2 \; + \; 2H_2O \\ 1 \text{ vol.} \; + \; 2 \text{ vol.} \; &\rightarrow \; 1 \text{ vol.} \; + \; 2 \text{ vol.} \end{aligned} \qquad (4.4)$$

2 volumes of oxygen are required to burn 1 volume of methane. The volume of air necessary to provide 2 volumes of oxygen can be deduced from the composition of air by volume:

Oxygen (O_2) : 21%

Nitrogen (N_2): 79%

This shows that one volume of oxygen in air corresponds to 79/21 (3.76) volumes of nitrogen. The amount of air required for stoichiometric combustion of 1 volume of methane is thus.

Stoichiometric Air Requirement: 2 vol. O_2 + 7.52 vol. N_2 = 9.52 vol. air

If the available air is greater than the stoichiometric air requirement, the excess air is simply added to the combustion equation e.g.

$$\begin{aligned} CH_4 \; + \; 2O_2 \; + \; 7.52N_2 \; + \; (M-2)(O_2 \; + \; 3.56\,N_2) \\ \rightarrow CO_2 \; + \; 2H_2O \; + \; 7.52N_2 \; + \; (M-2)(O_2 \; + \; 3.56N_2) \end{aligned} \qquad (4.5)$$

where M is the number of volumes of oxygen available for combustion. Excess air is usually expressed as a percentage increase over the stoichiometric air requirement. The equation for excess air by volume is:

$$\text{Excess Air} = \frac{\text{Volume Actual Air - Volume Stoichiometric Air}}{\text{Volume Stoichiometric Air}} \text{ x } 100\% \qquad (4.6)$$

Sub-stoichiometric Combustion

If the air available for combustion is less than the stoichiometric air requirement, sub-stoichiometric combustion will occur. Sub-stoichiometric combustion is quite different to stoichiometric combustion - it involves complex chemical equilibria and reaction kinetics. However the following simplified mechanism of combustion can be assumed:

Stage 1. The available oxygen first burns the hydrogen in the fuel to water vapour

Stage 2. Any remaining oxygen then burns the carbon in the fuel to carbon monoxide

Stage 3. Any remaining oxygen then burns the carbon monoxide to carbon dioxide.

Depending on the amount of oxygen available, some or all of the 'stages' of combustion will carried out. The combustion reaction equations are presented below for each stage of sub-stoichiometric equation. The number of volumes of oxygen available per volume of methane is denoted by M.

Stage 1: ($M \leq 1$)

If there is less than 1 volume of oxygen available per volume of methane, only some of the hydrogen is burned is water vapour.

$$CH_4 + (M)(O_2 + 3.56N_2) \rightarrow (M)(2)H_2O + (M)(3.56)N_2 \tag{4.7}$$

If 1 volume of oxygen is available all the hydrogen in 1 volume of methane can be burned to water vapour

$$CH_4 + O_2 + 3.56N_2 \rightarrow 2H_2O + 3.56N_2 \tag{4.8}$$

Stage 2: ($1 < M \leq 1.5$)

If there is between 1 and 1.5 volumes of oxygen available per volume of methane, all of the hydrogen is burned to water vapour. Some of the carbon is burned to carbon monoxide.

$$CH_4 + (M)(O_2 + 3.56N_2) \rightarrow 2H_2O + (2)(M-1)CO + (M)(3.56)N_2 \tag{4.9}$$

Stage 3: ($1.5 < M < 2$)

If there is between 1.5 and 2 volumes of oxygen available per volume of methane, all of the hydrogen is burned to water vapour, all of the carbon is burned to carbon monoxide and some of the carbon monoxide is burned to carbon dioxide. The combustion reactions for this stage occurs in two sub-stages.

Firstly, all the hydrogen is burned to water vapour and all the carbon is burned to carbon monoxide:

$$CH_4 + (M)(O_2 + 3.56N_2) \rightarrow 2H_2O + CO + (M-1.5)O_2 + (M)(3.56)N_2 \tag{4.10}$$

Secondly some or all of the carbon monoxide is burned to produce carbon dioxide. For each

volume of oxygen available, two volumes of carbon dioxide can be produced.

$$CO + (M - 1.5)O_2 \rightarrow (2)(M - 1.5)CO_2 + (1 - (2)(M - 1.5))CO \tag{4.11}$$

The reaction equations for these two sub-stages can be rewritten as a single equation:

$$CH_4 + (M)(O_2 + 3.56N_2) \rightarrow 2H_2O + (2)(M - 1.5)CO_2 + (1 - (2)(M - 1.5)CO + (M)(3.56)N_2 \tag{4.12}$$

4.2.3.2 Solid and Liquid Fuels

The composition of solid and liquid fuels is provided in terms of the mass of each of the elements present in the fuel. A typical analysis of #2 distillate oil is as follows:

Carbon,	C:	87.3%
Hydrogen,	H_2:	12.5%
Sulphur,	S:	0.20%

Fuels may also contain other constituents, such as oxygen, nitrogen, ash or moisture.

Stoichiometric Combustion

The stoichiometric air requirement for a solid or liquid fuel is found by considering separately the combustion equation for each constituent of the fuel. For example, the mass of oxygen required to fully combust 12 kg of carbon can be obtained using molecular weights and the combustion equation for carbon:

$$\begin{aligned} C \; &+ \; O_2 \; \rightarrow \; CO_2 \\ 12\text{ kg} &+ 32\text{kg} \rightarrow 44\text{ kg} \end{aligned} \tag{4.13}$$

The mass of oxygen required to burn all the carbon in 1 kg of #2 distillate oil (0.873 kg from the analysis) is in direct proportion to the amount of oxygen used to burn 12 kg of carbon.

$$\begin{aligned} C \quad &+ \quad O_2 \quad \rightarrow \quad CO_2 \\ 0.873\text{ kg} \quad &+ \; 0.873 \times \frac{32}{12}\text{ kg} \; \rightarrow 0.873 \times \frac{44}{12}\text{ kg} \\ \Rightarrow 0.873\text{ kg} \; &+ \; 2.328\text{ kg} \quad \rightarrow \quad 3.201 \quad \text{kg} \end{aligned}$$

The combustion equation for hydrogen is:

$$\begin{aligned} H_2 \; &+ \; \frac{1}{2}O_2 \; \rightarrow \; H_2O \\ 2\text{ kg} \; &+ \; 16\text{kg} \; \rightarrow \; 18\text{ kg} \end{aligned} \tag{4.14}$$

For the hydrogen in 1 kg of #2 distillate oil:

$$H_2 + \frac{1}{2}O_2 \rightarrow H_2O$$

$$0.125\ \text{kg} + 0.125 \times \frac{16}{2}\ \text{kg} \rightarrow 0.125 \times \frac{18}{2}\ \text{kg}$$

$$\Rightarrow 0.125\ \text{kg} + 1\ \text{kg} \rightarrow 1.25\ \text{kg}$$

The combustion equation for sulphur is:

$$S + O_2 \rightarrow SO_2$$
$$32\ \text{kg} + 32\text{kg} \rightarrow 64\ \text{kg} \qquad (4.15)$$

For the sulphur in 1 kg of #2 distillate oil:

$$S + O_2 \rightarrow SO_2$$

$$0.0021\ \text{kg} + 0.0021 \times \frac{32}{32}\ \text{kg} \rightarrow 0.0021 \times \frac{64}{32}\ \text{kg}$$

$$\Rightarrow 0.0021\ \text{kg} + 0.0021\ \text{kg} \rightarrow 0.0042\ \text{kg}$$

The amount of oxygen required to burn each component of the fuel in 1 kg of #2 distillate is summarised in Table 4.1.

Component	Oxygen Combustion Requirements (kg/kg)
Carbon	2.3280
Hydrogen	1.0000
Sulphur	0.0021

Table 4.1 Combustion Requirements for Components of #2 Oil

From the above, it can be seen that the total amount of oxygen required to burn 1 kg of #2 distillate oil, is equal to (2.328 + 1 + 0.0021) or 3.3 kg.

The mass of air required to supply this mass of oxygen can be found from the composition of air by mass:

Oxygen (O_2):	23%
Nitrogen (N_2)	76%

This shows that 1 kg of oxygen corresponds to 76/23=3.3 kg of nitrogen, in air. The mass of air required for stoichiometric combustion of 1 kg of #2 distillate oil is.

Stoichiometric Air Requirement: 3.3 kg O_2 + (3.3)(3.3) kg N_2 = 14.2 kg air

As for gases, any excess air is mixed with the other combustion gases. The equation for calculating the mass of excess air is:

$$\text{Excess Air} = \frac{\text{Mass Actual Air - Mass Stoichiometric Air}}{\text{Mass Stoichiometric Air}} \times 100\% \qquad (4.16)$$

Sub-stoichiometric Combustion

Sub-stoichiometric combustion will occur, if the air available for combustion is less than the stoichiometric air combustion. As for gaseous fuels, it is possible to assume a simplified mechanism for sub-stoichiometric combustion:

1. The available oxygen first burns the sulphur in the fuel to sulphur dioxide (This does not occur first in practice. However, it may be important to have available worst-case estimates for sulphur production. In any case, the volume of sulphur in most fuels is so small, that this is an unimportant simplification).
2. Any remaining oxygen then burns the hydrogen in the fuel to water vapour
3. Any remaining oxygen then burns the carbon in the fuel to carbon monoxide
4. Any remaining oxygen then burns the carbon monoxide to carbon dioxide.

The model first calculates how much oxygen is required to complete each stage of the combustion process. The mass of oxygen available for combustion per kg of fuel is denoted by M.

Stage 1: ($M \leq 0.0021$ kg)

The amount of oxygen required to burn all the sulphur in 1 kg of fuel to sulphur dioxide has previously been calculated and is equal to 0.0021 kg.

If the mass of oxygen available for combustion is less than or equal to 0.0021 kg, this stage will occur. Some or all of the sulphur in the fuel will be burned to sulphur dioxide. The amount of sulphur dioxide produced can be calculated as follows:

$$\begin{array}{ccccc} S & + & O_2 & \rightarrow & SO_2 \\ M \times \dfrac{0.0021}{0.0021} & + & M & \rightarrow & M \times \dfrac{0.0042}{0.0021} \quad \text{(kg)} \end{array} \qquad (4.15)$$

Stage 2: (0.0021 kg $< M \leq$ 1.002 kg)

The amount of oxygen required to burn all the sulphur to sulphur dioxide and all the hydrogen to water vapour in 1 kg of fuel is equal to (0.0021+1) or 1.002 kg of oxygen.

If there is between 0.0021 kg and 1.002 kg of oxygen available per kg of fuel, this stage will occur. All the sulphur in the fuel will be burned to sulphur dioxide. Some or all of the hydrogen in the fuel will be burned to water vapour. The amount of sulphur dioxide and water vapour produced can be calculated as follows:

$$\begin{aligned} S \;+\; O_2 &\rightarrow SO_2 \\ 0.0021 \;+\; 0.0021 &\rightarrow 0.0042 \;\; (kg) \end{aligned} \tag{4.15}$$

$$\begin{aligned} H_2 \;+\; \frac{1}{2}O_2 &\rightarrow H_2O \\ (M - 0.0021)(0.125) + (M - 0.0021) &\rightarrow (M - 0.0021)(1.25) \;\; (kg) \end{aligned} \tag{4.17}$$

Stage 3 (1.002 kg<M ≤2.166 kg):

If there is between 1.002 kg and 2.166 kg of oxygen available per kg of fuel, this stage will occur. All the sulphur and hydrogen in the fuel will first be burned. The amount of sulphur dioxide and water vapour produced is calculated as follows.

$$\begin{aligned} S \;+\; O_2 &\rightarrow SO_2 \\ 0.0021 \;+\; 0.0021 &\rightarrow 0.0042 \;\; (kg) \end{aligned} \tag{4.15}$$

$$\begin{aligned} H_2 \;+\; \frac{1}{2}O_2 &\rightarrow H_2O \\ 0.125 \;+\; 1 &\rightarrow 1.25 \;\; (kg) \end{aligned} \tag{4.18}$$

The remaining oxygen is used to burn some or all of the carbon in the fuel to carbon monoxide. The amount of oxygen required to burn all the carbon in the fuel to carbon monoxide must be determined.

$$\begin{aligned} C \;+\; \frac{1}{2}O_2 &\rightarrow CO \\ 12\,kg + 16kg &\rightarrow 28\,kg \end{aligned} \tag{4.19}$$

For the carbon in 1 kg of #2 distillate oil:

$$\begin{aligned} 0.873\,kg \;+\; 0.873\times\frac{16}{12}\,kg &\rightarrow 0.873\times\frac{28}{12}\,kg \\ \Rightarrow 0.873\,kg \;+\; 1.164\,kg &\rightarrow 2.037\,kg \end{aligned}$$

The total amount of oxygen required to complete stage 3 is equal to (1.002+1.164) or 2.166 kg.

The amount of carbon monoxide produced can be calculated using the combustion equation for this process:

$$\begin{aligned} &C + \frac{1}{2}O_2 \rightarrow CO \\ &(M-1.002)\left(\frac{0.873}{1.164}\right) + (M-1.002) \rightarrow (M-1.002)\left(\frac{2.037}{1.164}\right) \quad \text{(kg)} \end{aligned} \tag{4.20}$$

Stage 4: (2.166 kg<M ≤3.3 kg)

If there is between 2.166 and 3.3 kg of oxygen available per kg of fuel, this stage will occur. The combustion equations for this stage can be written in two steps. Firstly, the sulphur is burned to sulphur dioxide, the hydrogen is burned to water vapour and the carbon is burned to carbon monoxide:

$$\begin{aligned} &S + O_2 \rightarrow SO_2 \\ &0.0021 + 0.0021 \rightarrow 0.0042 \quad \text{(kg)} \end{aligned} \tag{4.15}$$

$$\begin{aligned} &H_2 + \frac{1}{2}O_2 \rightarrow H_2O \\ &0.125 + 1 \rightarrow 1.25 \quad \text{(kg)} \end{aligned} \tag{4.21}$$

$$\begin{aligned} &C + \frac{1}{2}O_2 \rightarrow CO \\ &0.873 + 1.164 \rightarrow 2.037 \quad \text{(kg)} \end{aligned} \tag{4.22}$$

There is now (M-2.166) kg of oxygen available for combustion of carbon monoxide to carbon dioxide. The combustion equation for this reaction is:

$$\begin{aligned} &CO + \frac{1}{2}O_2 \rightarrow CO_2 \\ &28 + 16 \rightarrow 44 \quad \text{(kg)} \\ &\Rightarrow (M-2.166)\left(\frac{28}{16}\right) + (M-2.166) \rightarrow (M-2.166)\left(\frac{44}{16}\right) \quad \text{(kg)} \\ &= (M-2.166)(1.75) + (M-2.166) \rightarrow (M-2.166)(2.75) \quad \text{(kg)} \end{aligned} \tag{4.23}$$

The combustion equation shows that burning carbon monoxide with (M-2.166) kg of oxygen can produce (M-2.166)(2.75) kg of carbon dioxide. Consequently, this reduces the mass of carbon monoxide in the combustion gases by (2.75)(M-2.166) kg.

4.2.3.3 Calculation of Gas Composition

The composition of the combustion gases is described in terms of the percentage of each constituent of the combustion gases by volume. For example, the percentage of oxygen in the combustion gases (generated by burning methane) is calculated as follows:

$$\text{Percentage } O_2 = \frac{\text{Volume } O_2}{\text{Volume } O_2 + \text{Volume } N_2 + \text{Volume } CO_2 + \text{Volume } H_2O} * 100 \quad (4.24)$$

If the volume of water vapour in the combustion gases is included in the total volume of the combustion gases (as in equation 4.24), the percentage is referred to as "wet". Otherwise, the percentage is referred to as "dry"

For liquid or solid fuels, the products of combustion have been calculated in units of mass rather than volume. In this case, the volume of each constituent of the combustion gases is calculated from its mass using molecular weights. For example, 1.25 kg of H_2O are produced during the stoichiometric combustion of #2 distillate oil. H_2O has a molecular weight of 18, so 1.25/18 or 0.0684 volumes of H_2O are produced for each kg of fuel.

4.2.4 Dynamics of Gas Composition Measurement

The gas composition cannot be measured in the combustion chamber. It is measured downstream where the flue gases are cooler. Consequently, there are significant lags and delays between the actual combustion process and the gas composition measurement. These comprise

1. Pure transportation delay as combustion gases travel to sensor
2. Lag due to mixing of combustion gases in furnace
3. Pure sensor measurement delay

$$\frac{d}{dt}[O_{2_{lag}}] = \frac{O_{2_{mod}} - O_{2_{lag}}}{\tau_{gc}}$$

$$O_{2_{meas}}(t + T_{cc} + T_{sensor}) = O_{2_{lag}}(t)$$

where

$O_{2_{mod}}$ = Value of O_2 calculated using static combustion equations
$O_{2_{lag}}$ = Value of $O_{2_{mod}}$ subject to a time lag
$O_{2_{meas}}$ = Modelled value of measured O_2
τ_{gc} = Lag time constant
T_{cc} = Combustion chamber time delay
T_{sensor} = Sensor time delay

(4.25)

The combustion chamber time delay depends on the velocity of the combustion gases in the furnace. The velocity of the combustion gases is calculated from their mass flow rate.

$$v_g = \frac{m_g}{\rho_g A_{furn}} \tag{4.26}$$

where

ρ_g = combustion gas density

A_{furn} = cross - sectional area of the combustion gas path

The time delay can be calculated using the length of the flue gas path and the combustion gas velocity.

$$T_{cc} = \frac{L_{furn}}{v_g} \tag{4.27}$$

where

L_{furn} = length of the combustion gas path through furnace

4.2.5 Summary of Variables

Inputs	
Fuel Mass Flow Rate	m_{fu}
Air Mass Flow Rate	m_{air}
Fuel Temperature	T_{fu}
Air Temperature	T_{air}

Table 4.2 Combustion Process Inputs

System Parameters	
Fuel Composition Analysis	-
Calorific Value of Fuel	C_{fu}
Time Lag in Combustion Chamber	τ_{gc}
Distance to O_2 sensor	L_{furn}
Cross-sectional Area of Furnace	A_{furn}
Sensor Measurement Delay	T_{sensor}

Table 4.3 Combustion Process System Parameters

Thermodynamic Properties	
Specific Heat of Air at Constant Pressure	c_{air}
Specific Heat of Fuel at Constant Pressure	c_{fu}
Density of Combustion Gas	ρ_g

Table 4.4 Combustion Process Thermodynamic Properties

State Variables	
Percentage O_2 in Flue Gases by Volume (Wet/Dry)	-

Table 4.5 Combustion Process State Variables

Outputs	
Heat Generated by Combustion	Q_{cc}
Combustion Gas Mass Flow Rate	m_g
Percentage O_2 in Flue Gases by Volume (Wet/Dry)	-

Table 4.6 Combustion Process Outputs

4.3 Development of Fluid-Side Model

Note: the nomenclature for the fluid side model is in Appendix A.

The description of the fluid-side of the boiler model is divided into three main sections, reflecting the physical construction of the boiler:

1. Economiser
2. Drum, Downcomers and Risers
3. Superheater

A fourth section, which describes a simple downstream process is also included. This process represents an arbitrary boiler load.

Each of these sections is subdivided into the following subsections, which are described in detail below:

- **Lumped Systems**

  There are three types of lumped systems employed in this model

  1. Gas
  2. Metal
  3. Fluid

  Both the economiser and superheater are simple heat exchangers and can be modelled by just three lumped systems - a gas lumped system, a metal lumped system and a fluid lumped system. The gas lumped system imparts heat to the metal lumped system which in turn imparts heat to the fluid lumped system.

The model of the drum, downcomers and risers is far more complex and requires a greater number of fluid lumped systems.

- **Assumptions**

The assumptions pertaining to each section are listed. The model is largely defined by the choice of lumped systems and of the assumptions.

- **Continuity and State Variable Equations**

Continuity of mass, energy and momentum equations form the backbone of the model. In this section the continuity equations used to model the various lumped systems are presented and explained.

The gas lumped systems are modelled by continuity of energy and mass equations. Each gas lumped system is the volume of gas responsible for imparting heat to a metal lumped system and is of uniform temperature throughout. For modelling purposes it is assumed that the gases are generated in the combustion chamber (which is lined by risers), flow past the superheater, then a second set of riser tubes and finally the economisers. The gas lumped systems are also referred to as gas-paths e.g. the economiser gas path. A continuity of energy equation is used to trace the energy stored in each gas path.

The pressure of the gases is assumed to be held constant by induced and forced draft fans. As a result the mass flow rate of gases is assumed to be equal at each point in the furnace and a steady state continuity of mass equation can be used for each gas path. A further consequence of steady state mass flow is that the momentum of the gas is constant, and the continuity of momentum equation is also steady state.

The metal lumped systems are modelled by continuity of energy equations only. Continuity of mass and momentum equations are not required for solids of fixed mass and zero velocity.

Continuity of mass, energy and momentum equations are used for the fluid lumped systems.

By rearranging the continuity equations, a set of equations is obtained for the rate of change of each state variable. These equations are highlighted by being italicised and placed inside a box.

- **Thermodynamic State Equations**

  The continuity equations employ a number of thermodynamic properties such as density, enthalpy and specific heat. These properties are not useful model outputs as they cannot be directly compared with measurable thermodynamic properties such as temperature and pressure. However, there is a known relationship between all the thermodynamic variables of a substance which is prescribed by the state of the substance e.g. solid, gas, subcooled water. In order to solve a set of continuity equations and obtain equations for the state variables, it is necessary to know the thermodynamic relationship between these variables.

  This section outlines the relationship between the thermodynamic properties used in the continuity equations and states whether any simplifying assumptions have been adopted. For example, it is reasonable to neglect the effect of pressure on gas density and enthalpy as the flue gas pressure is practically constant due to the action of forced and induced draft fans at the furnace inlet and outlet.

- **Equations for Dependent Variables**

  The continuity equations may also involve variables which are neither thermodynamic properties nor state variables. Such variables are calculated from state variables and model inputs and thus are termed dependent variables. A typical dependent variable is heat transfer rate.

- **Summary of Variables**

  The inputs, state variables and system parameters for this section of the model are summarised in tabular format.

4.3.1 Economiser

4.3.1.1 Lumped Systems (Economiser)

Subcooled fluid enters the economiser from the aerator. It is preheated to a higher temperature by waste flue gases before it enters the drum.

The economiser model is made up of three lumped systems:

1. Economiser gas path
2. Economiser metal
3. Economiser fluid

4.3.1.2 Assumptions (Economiser)

1. Economiser contains subcooled water
2. Mass flow rate of fluid is constant throughout economiser
3. No heat is lost through furnace walls.

4.3.1.3 Continuity and State Variable Equations (Economiser)

Economiser Gas Mass Balance

The mass flow rate of gas through the furnace is assumed to be steady-state and equal to the mass generation rate of combustion gases. The mass flow rate of gases through the economiser is likewise equal to the mass generation rate of combustion gases.

$$m_{eg_i} = m_{eg_o} = m_g \tag{4.28}$$

Economiser Gas Energy Balance

The combustion gases are assumed to flow from the second bank of riser tubes to the economiser. The energy remaining in the gases leaving the riser tubes is partially transferred to the fluid in the economiser. The remainder is stored in the combustion gases leaving the economiser gas path.

$$\frac{d}{dt}[\rho_{eg_o} V_{eg} m_g c_{eg} T_{eg_o}] = m_g (c_{rg2} T_{rg2_o} - c_{eg} T_{eg_o}) - Q_{egm} \tag{4.29}$$

Economiser Gas Temperature

Temperature was chosen as the state variable to describe economiser gas energy because it is easily related to gas energy and it is also a useful output. The equation for the rate of change of economiser gas temperature is obtained by rearranging the economiser gas energy balance:

$$\frac{d}{dt}[T_{eg_o}] = \frac{m_g (c_{rg2} T_{rg2_o} - c_{eg} T_{eg_o}) - Q_{egm}}{\rho_{eg_o} V_{eg} c_{eg}} \tag{4.30}$$

Economiser Metal Energy Balance

The energy of the metal economiser tubes is increased by energy received from the combustion gases and decreased by the energy transferred to the fluid in the economiser.

$$\frac{d}{dt}[\rho_{em} V_{em} c_{em} T_{em}] = Q_{egm} - Q_{emf} \tag{4.31}$$

Economiser Fluid Mass Balance

The mass flow rate of fluid through the economiser is assumed to be steady-state and equal to the feedwater flow rate into the boiler. This assumption is justifiable on the basis that the economiser contains subcooled water which is practically incompressible. As a result, any change in the feedwater flow rate will be transferred virtually instantaneously to the economiser inlet and outlet. This assumption implies that the pressure of the economiser fluid is constant as variations in pressure cannot occur without transient differences between the economiser inlet and outlet mass flow rate.

$$m_{e_i} = m_{e_o} = m_{fw} \tag{4.32}$$

Economiser Metal Temperature

Temperature was also chosen as the state variable for the economiser metal energy because it is easily related to metal energy and is a useful output. The equation for the rate of change of economiser metal temperature is:

$$\frac{d}{dt}[T_{em}] = \frac{Q_{egm} - Q_{emf}}{\rho_{em} V_{em} c_{em}} \tag{4.33}$$

Economiser Fluid Energy Balance

The continuity of energy equation for the economiser fluid tracks the energy of fluid in the economiser by comparing the input and output energy flow rate.

$$\frac{d}{dt}[V_e \rho_{e_o} h_{e_o}] = Q_{emf} + m_{fw} h_{fw} - m_{e_o} h_{e_o} \tag{4.34}$$

The economiser fluid has been modelled by one dynamic continuity of energy equation only. This equation shows that the energy of subcooled fluid is proportional to a product of fluid density and enthalpy. In general the density and enthalpy of a subcooled fluid are two independent thermodynamic properties and thus, it is not possible to solve this equation. However in this instance, it has been assumed that the pressure of fluid in the economiser is constant. Consequently variations in density or enthalpy are related to variations in temperature only. (In general variations in density and enthalpy of a subcooled fluid are related to variations in both pressure and temperature).

By rewriting density and enthalpy as functions of temperature, the continuity of energy equation can be rewritten as an equation with just one independent variable - temperature.

Firstly, the economiser energy equation must be expanded using the product rule:

$$V_e \rho_{e_o} \frac{d}{dt}[h_{e_o}] + V_e h_{e_o} \frac{d}{dt}[\rho_{e_o}] = Q_{emf} + m_{fw} h_{fw} - m_{e_o} h_{e_o} \qquad (4.35)$$

Secondly, the rate of change of density and enthalpy must be written in terms of the rate of change of temperature and pressure. The equations used below, are derived from the thermodynamic state equations.

$$\frac{d}{dt}[\rho_{e_o}] = \frac{\delta \rho_{e_o}}{\delta T_{e_o}} \frac{d}{dt}[T_{e_o}]$$
$$\frac{d}{dt}[h_{e_o}] = \frac{\delta h_{e_o}}{\delta T_{e_o}} \frac{d}{dt}[T_{e_o}] \qquad (4.36)$$

Thirdly, the expressions for the rate of change of density and enthalpy must be substituted into the continuity of energy equation. The continuity of energy equation can then be rewritten as an equation for the rate of change of economiser fluid temperature:

Economiser Fluid Temperature

$$\frac{d}{dt}[T_{e_o}] = \frac{Q_{emf} + m_{fw}(h_{fw} - h_{e_o})}{V_e(\rho_{e_o} \frac{\delta h_{e_o}}{\delta T_{e_o}} + h_{e_o} \frac{\delta \rho_{e_o}}{\delta T_{e_o}})} \qquad (4.37)$$

4.3.1.4 Thermodynamic State Equations (Economiser)

The economiser continuity equations employ several thermodynamic properties, namely density, enthalpy and specific heat. These properties are not useful model outputs as they cannot be directly compared with measurable thermodynamic properties such as temperature and pressure. However, it is possible to substitute between the different thermodynamic variables given that the relationship between the variables is known. In practice, all of the properties listed below are functions of both pressure and temperature. However in certain cases the effect of pressure or temperature may be negligible and can be discounted.

Table 4.7 gives the assumed relationship between the thermodynamic variables and highlights any implicit assumptions.

Thermodynamic Property	State	Thermodynamic Relations	Assumptions
Density of Economiser Water	Subcooled Fluid	$\rho_{e_o} = f(T_{e_o})$ (4.38)	Fluid pressure has negligible effect
Derivative of Density of Economiser Water	Subcooled Fluid	$\frac{d}{dt}[\rho_{e_o}] = \frac{\delta \rho_{e_o}}{\delta T_{e_o}} \frac{d}{dt}[T_{e_o}]$ (4.39)	Fluid pressure has negligible effect
Enthalpy of Economiser Water	Subcooled Fluid	$h_{e_o} = f(T_{e_o})$ (4.40)	Fluid pressure has negligible effect
Derivative of Enthalpy of Economiser Water	Subcooled Fluid	$\frac{d}{dt}[h_{e_o}] = \frac{\delta h_{e_o}}{\delta T_{e_o}} \frac{d}{dt}[T_{e_o}]$ (4.41)	Fluid pressure has negligible effect
Economiser Gas Path Specific Heat at Constant Pressure	Gas	$c_{eg_o} = f(T_{eg_o})$ (4.42)	Gas pressure is nearly constant, so has negligible effect
Economiser Gas Path Density	Gas	$\rho_{eg_o} = f(T_{eg_o})$ (4.43)	Gas pressure is nearly constant, so has negligible effect
Economiser Metal Specific Heat at Constant Pressure	Solid	$c_{em} = c_m$ (4.44)	Metal temperature and pressure have negligible effect
Economiser Metal Density	Solid	$\rho_{em} = \rho_m$ (4.45)	Metal temperature and pressure have negligible effect

Table 4.7 Modelled Thermodynamic Relations for Economiser Gas, Metal and Fluid

4.3.1.5 Equations for Dependent Variables (Economiser)

1. Q_{egm} - *Economiser Gas-Metal Heat Transfer Rate*

Heat transfer from the economiser gas path to the economiser metal is assumed to be convective, and is calculated using the following empirical equation (Kwan and Anderson (1970)):

$$Q_{egm} = k_{egm} m_g^{0.6} (T_{eg_o} - T_{em}) \tag{4.46}$$

2. Q_{emf} - *Economiser Metal-Fluid Heat Transfer Rate*

Heat transfer from the economiser metal to the economiser fluid is assumed to be convective and is calculated using the following empirical equation (Kwan and Anderson (1970)):

$$Q_{emf} = k_{emf} m_{e_o}^{0.8} (T_{em} - T_{e_o}) \tag{4.47}$$

4.3.1.6 Summary of Variables (Economiser)

Inputs	
Feedwater Mass Flow Rate	m_{fw}
Temperature of Feedwater	T_{e_i}

Table 4.8 Economiser Inputs

System Parameters	
Economiser Gas-Metal Heat Transfer Coefficient	k_{egm}
Economiser Metal-Fluid Heat Transfer Coefficient	k_{emf}
Volume of Economiser	V_e
Volume of Economiser Gas	V_{eg}
Volume of Economiser Metal	V_{em}

Table 4.9 Economiser System Parameters

State Variables	
Economiser Gas Temperature	T_{eg_o}
Economiser Metal Temperature	T_{em}
Economiser Fluid Temperature	T_{e_o}

Table 4.10 Economiser State Variables

4.3.2 Drum, Downcomers and Risers

4.3.2.1 Lumped Systems (Drum, Downcomers and Risers)

The drum, downcomers and risers are the most important and the most complex section of the boiler. For simple boiler models, it is sometimes considered sufficient to model this section using one lumped system only (Astrom and Bell (1988)). However, this is unlikely to result in an accurate model for several reasons:

- Considerable temperature differences occur throughout this section of the boiler. The highest temperatures are found in the riser gas path (*circa* 1200°C), which receives hot combustion gases directly from the combustion chamber. The lowest temperatures occur in the drum water (*circa* 270°C), which is mixed with subcooled liquid from the economiser.

- All three fluid states exist in the drum, downcomers and risers: the water in the drum and downcomers is subcooled; the fluid in the risers is saturated; the steam in the drum may be either saturated or superheated.

The choice of lumped systems for the drum, downcomers and risers is as follows:

Gas Lumped Systems:
1. Riser gas path, pass 1
2. Riser gas path, pass 2

Metal Lumped Systems:
3. Riser metal, pass 1
4. Riser metal, pass 2

Fluid Lumped Systems:
5. Water in drum
6. Steam in drum
7. Water in downcomers
8. Water-steam mixture in risers

There are no gas path or metal lumped systems for the drum and downcomers. It is assumed that the amount of energy absorbed by the fluid in the drum or downcomers from the combustion gases is negligible.

It is assumed that the riser tubes are placed into two locations. One set of riser tubes is located in the combustion chamber and receives radiative energy. The other set of riser tubes is located in the furnace and receives convective energy from hot flue gases. Consequently, the riser fluid is heated twice - once by radiation and once by convection. In order to model this, two riser gas lumped systems were used - one for radiative and one for convective heat transfer. Likewise two riser metal lumped systems were used - one for radiative and one for convective heat transfer. The radiative heat transfer is referred to as the first pass. The convective heat transfer is referred to as the second pass.

4.3.2.2 Assumptions (Drum, Downcomers and Risers)

1. Pressure is uniform throughout the drum, downcomers and risers - pressure head due to drum water is disregarded.
2. All the steam produced in the risers reaches the drum water surface.
3. The feedwater and water from the risers mixes completely with the drum water.
4. The mass flow rate of steam condensing and evaporating in the drum is negligible.
5. The recirculation mass flow rate in the drum, downcomer and risers is constant.
6. Water entering the downcomers is the same as drum water.
7. Water in drum is subcooled.

8. The temperature of the riser fluid lags the saturation temperature.
9. The temperature of the riser fluid changes in response to drum pressure changes only.
10. The heat absorbed by the fluid in the risers is uniformly distributed.
11. Steam quality is constant throughout risers.
12. Vapour and liquid velocities in the risers are identical.
13. No heat is lost through furnace walls.
14. All heat is absorbed in the risers, no heat is absorbed in the drum or downcomers.

4.3.2.3 Continuity and State Variable Equations (Drum, Downcomers and Risers)

Riser Gas Mass Balance, Pass 1

The mass flow rate of gas through the radiative riser gas path is steady state and equal to the mass generation rate of combustion gases.

$$m_{rg1_i} = m_{rg1_o} = m_g \tag{4.48}$$

Riser Gas Energy Balance, Pass 1

Subcooled water enters the risers from the downcomers and is heated to saturation temperature. A certain percentage of the fluid in the risers is further heated until it vaporises. This requires a substantial amount of energy, which is ensured by lining the combustion chamber with riser tubes so that they receive radiant heat from combustion. In this way most of the heat generated by combustion is transferred to the fluid in the risers.

The energy generated by combustion - Q_{cc} is partially absorbed by the riser tubes. The energy remaining in the gases is expelled to the next section of the furnace - the superheater gas path

$$\frac{d}{dt}[\rho_{rg1_o} V_{rg1} c_{rg1} T_{rg1_o}] = Q_{cc} - m_g c_{rg1} T_{rg1_o} - Q_{rgm1} \tag{4.49}$$

Riser Gas Path Temperature. Pass 1

The continuity of energy equation for the radiative riser gas path is rearranged to give an equation for the rate of change of radiative riser gas temperature.

$$\frac{d}{dt}[T_{rg1_o}] = \frac{Q_{cc} - m_g c_{rg1} T_{rg1_o} - Q_{rgm1}}{\rho_{rg1_o} V_{rg1} c_{rg1}} \tag{4.50}$$

Riser Metal Energy Balance, Pass 1

The radiative riser metal energy balance is dependent on the difference between the energy received from the radiative riser gas path and the energy transferred to the riser fluid.

$$\frac{d}{dt}[\rho_{rm1}V_{rm1}c_{rm1}T_{rm1}] = Q_{rgm1} - Q_{rmf1} \tag{4.51}$$

Riser Metal Temperature, Pass 1

The radiative riser metal energy equation is rearranged to give an equation for the rate of change of radiative riser metal temperature.

$$\frac{d}{dt}[T_{rm1}] = \frac{Q_{rgm1} - Q_{rmf1}}{\rho_{rm1}V_{rm1}c_{rm1}} \tag{4.52}$$

Riser Gas Mass Balance, Pass 2

The mass flow rate of gas through the convective riser gas path is steady state and equal to the mass generation rate of combustion gases.

$$m_{rg2_i} = m_{rg2_o} = m_g \tag{4.53}$$

Riser Gas Energy Balance, Pass 2

The flue gases flow from the combustion chamber, over the superheater tubes and then over a second set of riser tubes. The flue gases transfer heat to the second set of riser tubes by convection.

$$\frac{d}{dt}[\rho_{rg2_o}V_{rg2}c_{rg2}T_{rg2_o}] = m_g c_{sg}T_{sg_o} - m_g c_{rg2}T_{rg2_o} - Q_{rgm2} \tag{4.54}$$

Riser Gas Path Temperature, Pass 2

The continuity of energy equation for the radiative riser gas path is rearranged to give an equation for the rate of change of convective riser gas temperature

$$\frac{d}{dt}[T_{rg2_o}] = \frac{m_g c_{rg2}(T_{sg_o} - T_{rg2_o}) - Q_{rgm2}}{\rho_{rg2_o}V_{rg2}c_{rg2}} \tag{4.55}$$

Riser Metal Energy Balance, Pass 2

The convective riser metal energy balance is dependent on the difference between the energy received from the convective riser gas path and the energy transferred to the riser fluid.

$$\frac{d}{dt}[\rho_{rm2}V_{rm2}c_{rm2}T_{rm2}] = Q_{rgm2} - Q_{rmf2} \tag{4.56}$$

Riser Metal Temperature, Pass 2

An equation for the rate of change of convective riser metal temperature is obtained from the continuity of energy equation for the second bank of the riser tubes.

$$\frac{d}{dt}[T_{rm2}] = \frac{Q_{rgm2} - Q_{rmf2}}{\rho_{rm2}V_{rm2}c_{rm2}} \tag{4.57}$$

Drum and Downcomer Water, Riser Fluid Mass Balance

A single mass balance is written for all the water in the evaporation system plus the steam in the risers. The total mass of water in the evaporation system is made up of the water in the drum, the water in the downcomers and the water in the risers. Water enters the evaporation system as feedwater and is removed from the system by evaporation.

$$\frac{d}{dt}[V_{dw}\rho_{dw} + V_r\rho_{r_o} + V_{do}\rho_{do_o}] = m_{fw} - xm_r \tag{4.58}$$

This mass balance includes the mass of the drum water, the downcomer water and the riser fluid only. Riser fluid mass is proportional to riser fluid density and downcomer water mass is proportional to riser water density. Drum water mass is equal to the product of drum water volume and drum water density. Drum water density is rewritten in terms of drum pressure and drum water temperature. Downcomer water density is rewritten in terms of drum pressure and downcomer water temperature. The riser fluid is a mixture of saturated water and steam. The density of this mixture depends on riser fluid temperature, drum pressure and on the proportion of water and steam in the mixture i.e. the riser steam quality. Consequently, the riser fluid density can be rewritten as a function of riser fluid temperature, drum pressure and steam quality.

Based on the above, the mass balance can be rewritten in terms of the following variables: drum

pressure, drum water volume, drum water temperature, downcomer water temperature, riser steam quality and riser fluid temperature.

$$(V_{dw}\frac{\delta\rho_{dw}}{\delta P_d}+V_r\frac{\delta\rho_{r_o}}{\delta P_d}+V_{do}\frac{\delta\rho_{do_o}}{\delta P_d})\frac{d}{dt}[P_d]+\rho_{dw}\frac{d}{dt}[V_{dw}]+V_{dw}\frac{\delta\rho_{dw}}{\delta T_{dw}}\frac{d}{dt}[T_{dw}]$$
$$+\ V_{do}\frac{\delta\rho_{do_o}}{\delta T_{do_o}}\frac{d}{dt}[T_{do_o}]+V_r\frac{\delta\rho_{r_o}}{\delta x}\frac{d}{dt}[x]=m_{fw}-xm_r \tag{4.59}$$

Riser fluid temperature is written on the right hand side of the above equation as it is not an independent state variable. Changes in riser fluid temperature are dictated by changes in drum pressure.

For brevity the continuity equation is rewritten as:

$$a_{31}\frac{d}{dt}[P_d]+a_{33}\frac{d}{dt}[V_{dw}]+a_{35}\frac{d}{dt}[T_{dw}]+a_{36}\frac{d}{dt}[T_{do_o}]+a_{37}\frac{d}{dt}[x]=k_3$$

where

$$a_{31}=V_{dw}\frac{\delta\rho_{dw}}{\delta P_d}+V_r\frac{\delta\rho_{r_o}}{\delta P_d}+V_{do}\frac{\delta\rho_{do_o}}{\delta P_d},\quad a_{33}=\rho_{dw},\ a_{35}=V_{dw}\frac{\delta\rho_{dw}}{\delta P_{dw}} \tag{4.60}$$
$$a_{36}=V_{do}\frac{\delta\rho_{do_o}}{\delta T_{do_o}},\quad a_{37}=V_r\frac{\delta\rho_{r_o}}{\delta x}\quad\text{and}\quad k_3=m_{fw}-xm_r$$

Drum Steam Mass Balance

The mass of steam in the drum is changed by steam entering the drum from the risers and steam leaving the drum to enter the superheater.

$$\frac{d}{dt}[V_{ds}\rho_{ds}]=xm_r-m_{s_i} \tag{4.61}$$

Drum steam mass is a product of steam volume and density. Density can be rewritten as a function of pressure and drum steam temperature. Consequently, drum steam mass can be rewritten in terms of the following variables: drum pressure, drum steam temperature and drum steam volume

$$V_{ds}\frac{\delta\rho_{ds}}{\delta P_d}\frac{d}{dt}[P_d]+V_{ds}\frac{\delta\rho_{ds}}{\delta T_{ds}}\frac{d}{dt}[T_{ds}]+\rho_{ds}\frac{d}{dt}[V_{ds}]=xm_r-m_{s_i} \tag{4.62}$$

The continuity equation can be rewritten as:

$$a_{11}\frac{d}{dt}[P_d]+a_{12}\frac{d}{dt}[T_{ds}]+a_{14}\frac{d}{dt}[V_{ds}]=k_1$$

where

$$a_{11}=V_{ds}\frac{\delta\rho_{ds}}{\delta P_d},\quad a_{12}=V_{ds}\frac{\delta\rho_{ds}}{\delta T_{ds}},\quad a_{14}=\rho_{ds}\ \text{ and }\ k_1=xm_r-m_{s_i} \tag{4.63}$$

Drum Water Energy Balance

The energy of the water in the drum is increased by water entering the drum from the economiser and the risers and decreased by water leaving the drum and entering the downcomers. The mass flow rate of fluid entering and leaving the risers, and the mass flow rate of fluid into the downcomers m_r are assumed to be equal. The water entering the downcomers is assumed to be the same as drum water. This implies that perfect mixing of the drum water, feedwater and water from the risers occurs in the drum.

$$\frac{d}{dt}[V_{dw}\rho_{dw}h_{dw}]=m_{fw}h_{e_o}+(1-x)m_rh_{rw_o}-m_rh_{dw} \tag{4.64}$$

Drum water energy is equal to the product of drum water volume, density and enthalpy. Density and enthalpy can be rewritten as functions of drum pressure and drum water temperature. Consequently, the choice of state variables is: drum pressure, drum water temperature and drum water volume.

$$\begin{aligned}&V_{dw}(h_{dw}\frac{\delta\rho_{dw}}{\delta P_d}+\rho_{dw}\frac{\delta h_{dw}}{\delta P_d})\frac{d}{dt}[P_d]+\rho_{dw}h_{dw}\frac{d}{dt}[V_{dw}]\\&+V_{dw}(h_{dw}\frac{\delta\rho_{dw}}{\delta T_{dw}}+\rho_{dw}\frac{\delta h_{dw}}{\delta T_{dw}})\frac{d}{dt}[T_{dw}]=m_{fw}h_{e_o}+(1-x)m_rh_{rw_o}-m_rh_{dw}\end{aligned} \tag{4.65}$$

The equation is rewritten as:

$$a_{41}\frac{d}{dt}[P_d]+a_{43}\frac{d}{dt}[V_{dw}]+a_{45}\frac{d}{dt}[T_{dw}]=k_4$$

where

$$\begin{aligned}&a_{41}=V_{dw}(h_{dw}\frac{\delta\rho_{dw}}{\delta P_d}+\rho_{dw}\frac{\delta h_{dw}}{\delta P_d}),\quad a_{43}=\rho_{dw}h_{dw},\\&a_{45}=V_{dw}(h_{dw}\frac{\delta\rho_{dw}}{\delta T_{dw}}+\rho_{dw}\frac{\delta h_{dw}}{\delta T_{dw}})\ \text{ and }\ k_4=m_{fw}h_{e_o}+(1-x)m_rh_{rw_o}-m_rh_{dw}\end{aligned} \tag{4.66}$$

Riser Fluid Energy Balance

The riser fluid energy balance traces the total energy of the fluid in the risers and consequently the steam quality of the saturated water steam mixture in the risers. For a given saturation temperature, the higher the energy of the fluid in the risers, the greater the percentage of steam in the mixture.

$$\frac{d}{dt}[V_r \rho_{r_o} h_{r_o}] = m_r (h_{do_o} - h_{r_o}) + Q_{rmf1} + Q_{rmf2} \tag{4.67}$$

Riser fluid energy is proportional to a product of riser density and enthalpy. The riser fluid is a mixture of saturated water and steam. The density and enthalpy of this mixture depends on riser fluid temperature, drum pressure and on the proportion of water and steam in the mixture i.e. the riser steam quality. The riser fluid density and enthalpy can be rewritten as a function of riser fluid temperature, drum pressure and steam quality. The riser fluid energy balance can be rewritten in terms of the following variables: riser fluid temperature, drum pressure and steam quality.

$$V_r (h_{r_o} \frac{\delta \rho_{r_o}}{\delta P_d} + \rho_{r_o} \frac{\delta h_{r_o}}{\delta P_d}) \frac{d}{dt}[P_d] + V_r (h_{r_o} \frac{\delta \rho_{r_o}}{\delta x} + \rho_{r_o} \frac{\delta h_{r_o}}{\delta x}) \frac{d}{dt}[x] = m_r h_{do_o} - m_r h_{r_o} + Q_{rmf1} + Q_{rmf2} \tag{4.68}$$

For brevity, the equation is rewritten as:

$$a_{61} \frac{d}{dt}[P_d] + a_{67} \frac{d}{dt}[x] = k_6$$

where

$$a_{61} = V_r (h_{r_o} \frac{\delta \rho_{r_o}}{\delta P_d} + \rho_{r_o} \frac{\delta h_{r_o}}{\delta P_d}), \quad a_{67} = V_r (h_{r_o} \frac{\delta \rho_{r_o}}{\delta x} + \rho_{r_o} \frac{\delta h_{r_o}}{\delta x}) \tag{4.69}$$

$$\text{and } k_6 = m_r h_{do_o} - m_r h_{r_o} + Q_{rmf1} + Q_{rmf2}$$

Drum Steam Energy Balance

The energy of the steam in the drum is increased by steam entering the drum from the risers. Energy is removed by steam leaving the drum and entering the superheater.

$$\frac{d}{dt}[V_{ds} \rho_{ds} h_{ds}] = x m_r h_{rs_o} - m_{s_i} h_{ds} \tag{4.70}$$

Drum steam energy is a product of drum steam volume, density and enthalpy. Density and enthalpy are rewritten in terms of drum pressure and drum steam temperature. Drum steam

energy can be described by the following state variables: drum pressure, drum steam temperature and drum steam volume

$$V_{ds}(h_{ds}\frac{\delta\rho_{ds}}{\delta P_d}+\rho_{ds}\frac{\delta h_{ds}}{\delta P_d})\frac{d}{dt}[P_d]+V_{ds}(h_{ds}\frac{\delta\rho_{ds}}{\delta T_{ds}}+\rho_{ds}\frac{\delta h_{ds}}{\delta T_{ds}})\frac{d}{dt}[T_{ds}] + \rho_{ds}h_{ds}\frac{d}{dt}[V_s]=xm_r h_{rs_o}-m_{s_i}h_{ds} \tag{4.71}$$

The equation is rewritten as:

$$a_{21}\frac{d}{dt}[P_d]+a_{22}\frac{d}{dt}[T_{ds}]+a_{24}\frac{d}{dt}[V_{ds}]=k_2$$

where

$$a_{21}=V_{ds}(h_{ds}\frac{\delta\rho_{ds}}{\delta P_d}+\rho_{ds}\frac{\delta h_{ds}}{\delta P_d}),\quad a_{22}=V_{ds}(h_{ds}\frac{\delta\rho_{ds}}{\delta T_{ds}}+\rho_{ds}\frac{\delta h_{ds}}{\delta T_{ds}}), \tag{4.72}$$

$$a_{24}=\rho_{ds}h_{ds}\quad\text{and}\quad k_2=xm_r h_{rs_o}-m_{s_i}h_{ds}$$

Downcomer Water Energy Balance

The energy of the water in the downcomers is dependent only on the energy of the water entering the downcomer from the drum. as it is assumed that the fluid in the downcomers receives no energy from the combustion gases.

$$\frac{d}{dt}[V_{do}\rho_{do_o}h_{do_o}]=m_r(h_{dw}-h_{do_o}) \tag{4.73}$$

Downcomer water energy is proportional to a product of downcomer density and enthalpy. Density and enthalpy can be rewritten as functions of drum pressure and downcomer water temperature. The choice of state variables is: drum pressure, downcomer water temperature.

$$V_{do}(h_{do_o}\frac{\delta\rho_{do_o}}{\delta P_d}+\rho_{do_o}\frac{\delta h_{do_o}}{\delta P_d})\frac{d}{dt}[P_d]+V_{do}(h_{do_o}\frac{\delta\rho_{do_o}}{\delta T_{do_o}}+\rho_{do_o}\frac{\delta h_{do_o}}{\delta T_{do}})\frac{d}{dt}[T_{do_o}]=m_r h_{dw}-m_r h_{do_o} \tag{4.74}$$

The equation is rewritten as:

$$a_{51}\frac{d}{dt}[P_d]+a_{56}\frac{d}{dt}[T_{do_o}]=k_5$$

where

$$a_{51}=V_{do}(h_{do_o}\frac{\delta\rho_{do_o}}{\delta P_d}+\rho_{do_o}\frac{\delta h_{do_o}}{\delta P_d}),\quad a_{56}=V_{do}(h_{do_o}\frac{\delta\rho_{do_o}}{\delta T_{do_o}}+\rho_{do_o}\frac{\delta h_{do_o}}{\delta T_{do_o}}) \tag{4.75}$$

$$\text{and}\quad k_5=m_r h_{dw}-m_r h_{do_o}$$

Riser Fluid Temperature Equation

Boiler models generally adopt the assumption that the fluid in the risers is always in saturated equilibrium. This assumption is valid in the steady state but results in some serious modelling anomalies in the transient state. For example, following an increase in drum pressure it is assumed that the modelled fluid temperature rises immediately to the higher saturation temperature and lower saturation density corresponding to the new drum pressure i.e. an increase in modelled drum pressure leads to a decrease in modelled fluid density.

In practice, an increase in drum pressure will result in an increase in fluid density. A rigorous first-principles model of evaporation could be used to model this correctly but would be far too detailed in this context. Instead, a more intuitive approach has been adopted which assumes that riser fluid temperature lags the saturation temperature by a few seconds.

$$\frac{d}{dt}[T_{r_o}] = k_1(T_{sat} - T_{r_o}] \qquad (4.76)$$

With this approach riser fluid density and enthalpy are equated to the saturation values which correspond to the riser fluid temperature. A term which represents the effect of drum pressure is then added on to these saturation values.

This model is a better approximation to the real behaviour of the risers. For example, an increase in drum pressure will result in an increase in fluid density and enthalpy, and a corresponding decrease in the evaporation rate. The decreased evaporation rate allows the fluid to gradually change to its new saturation equilibrium.

The fluid pressure is assumed to be uniform for each of the lumped systems in the evaporation system. This is a reasonable assumption on the basis that the evaporation system is an open system containing mostly liquid, and pressure variations can move through it quickly.

A consequence of this assumption is that all the fluid mass and energy balances share a common state variable - drum pressure P_d. In order to solve the fluid mass and energy balances, they must be rewritten in terms of drum pressure (and the other state variables) and solved simultaneously. This has already been carried out for the continuity equations.

One more equation is required to solve the set of simultaneous equations:

Drum Constant Volume Equation

The relationship between drum water volume and drum steam volume is also required to solve the balances.

$$\frac{d}{dt}[V_{dw}]+\frac{d}{dt}[V_{ds}]=0 \tag{4.77}$$

The fluid balances - equations 4.58 to 4.76 and the drum constant volume equation - equation (4.77) can now be solved to give the following equations for the state variables:

Drum Pressure

$$\frac{d}{dt}[P_d]=\frac{l_1k_1+l_2k_2+l_3k_3+l_4k_4+l_5k_5+l_6k_6}{(a_{11}a_{22}-a_{12}a_{21})(a_{33}a_{45}-a_{35}a_{43})(a_{56}a_{67}-a_{12}a_{24}-a_{14}a_{22})(a_{31}a_{45}a_{56}a_{67}-a_{36}a_{51}a_{45}a_{67}-a_{37}a_{45}a_{56}a_{61}-a_{35}a_{\cdot}}$$

where

$$\begin{aligned}
l_1&=(a_{33}a_{45}-a_{35}a_{43})a_{56}a_{67}a_{22}\\
l_2&=-(a_{33}a_{45}-a_{35}a_{43})a_{56}a_{67}a_{12}\\
l_3&=-(a_{12}a_{24}-a_{14}a_{22})a_{45}a_{56}a_{67}\\
l_4&=(a_{12}a_{24}-a_{14}a_{22})a_{35}a_{56}a_{67}\\
l_5&=(a_{12}a_{24}-a_{14}a_{22})a_{36}a_{67}\\
l_6&=(a_{12}a_{24}-a_{14}a_{22})a_{37}a_{56}
\end{aligned} \tag{4.78}$$

Drum Water Volume

$$\frac{d}{dt}[V_{dw}]=\frac{a_{22}k_1-a_{12}k_2-(a_{11}a_{22}-a_{12}a_{21})\frac{d}{dt}[P_d]}{a_{12}a_{24}-a_{14}a_{22}} \tag{4.79}$$

Drum Steam Temperature

$$\frac{d}{dt}[T_{ds}]=\frac{k_1-a_{11}\frac{d}{dt}[P_d]-a_{14}\frac{d}{dt}[V_{ds}]}{a_{12}} \tag{4.80}$$

Drum Water Temperature

$$\frac{d}{dt}[T_{dw}]=\frac{k_4-a_{41}\frac{d}{dt}[P_d]-a_{43}\frac{d}{dt}[V_{dw}]}{a_{45}} \tag{4.81}$$

Downcomer Water Temperature

$$\frac{d}{dt}[T_{do_o}] = \frac{k_5 - a_{51}\frac{d}{dt}[P_d]}{a_{56}} \tag{4.82}$$

Riser Steam Quality

$$\frac{d}{dt}[x] = \frac{k_6 - a_{61}\frac{d}{dt}[P_d]}{a_{67}} \tag{4.83}$$

Superheater Fluid Inlet Momentum Balance

The steady-state continuity of momentum equation is used to calculate the velocity of steam from the drum to the superheater. The compressibility of the steam in the superheater is assumed to be negligible. This is justifiable on the basis that the velocity of steam in the superheater is considerably lower than acoustic velocities (Roberson and Crowe (1993))

$$\frac{P_d}{\rho_{ds} g} - \frac{P_{s_o}}{\rho_{ds} g} = \frac{v_{s_i}^2}{2g}(H\_LOSS\_D) \tag{4.84}$$

Drum Pressure Head	-	Super-heater Pressure Head	=	Kinetic Energy	×	Drum Frictional Loss

The mass flow rate of fluid can be calculated from velocity using the steady state continuity of mass equation for the fluid at the superheater inlet.

$$m_{s_i} = N_s v_{s_i} A_s \rho_{ds} \tag{4.85}$$

4.3.2.4 Thermodynamic State Equations (Drum, Downcomers and Risers)

The thermodynamic state equations encode the relationship between the thermodynamic properties of a fluid such as pressure, temperature, density, enthalpy, steam quality. Thermodynamic relationships are needed to substitute between the thermodynamic properties which are used in the continuity equations such as density and enthalpy and the measurable thermodynamic properties which are used as state variables such as temperature and pressure. The thermodynamic relationship depends primarily on the state of the substance - e.g. subcooled water, saturated water, gas. However for simplicity it may be reasonable to make some

additional assumptions about the relationship between the thermodynamic variables without loss of accuracy.

Table 4.11 gives the assumed state of each thermodynamic variable and its relationship to the state variables. It also highlights any other simplifying assumptions.

Thermodynamic Property	State	Thermodynamic Relations	Assumptions
Drum Steam Density	Superheated Steam	$\rho_{ds} = f(P_d T_{ds})$ (4.86)	None
Derivative of Drum Steam Density	Superheated Steam	$\frac{d}{dt}[\rho_{ds}] = \frac{\delta \rho_{ds}}{\delta T_{ds}} \frac{d}{dt}[T_{ds}] + \frac{\delta \rho_{ds}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.87)	None
Drum Steam Enthalpy	Superheated Steam	$h_{ds} = f(P_d T_{ds})$ (4.88)	None
Derivative of Drum Steam Enthalpy	Superheated Steam	$\frac{d}{dt}[h_{ds}] = \frac{\delta h_{ds}}{\delta T_{ds}} \frac{d}{dt}[T_{ds}] + \frac{\delta h_{ds}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.89)	None
Riser Steam Density	Nearly Saturated Steam	$\rho_{rs_o} = f(T_{r_o} P_d)$ (4.90)	See Note 1
Riser Steam Enthalpy	Nearly Saturated Steam	$h_{rs_o} = f(T_{r_o}, P_d)$ (4.91)	See Note 1
Riser Water Density	Nearly Saturated Water	$\rho_{rw_o} = f(T_{r_o}, P_d)$ (4.92)	See Note 1
Riser Water Enthalpy	Nearly Saturated Water	$h_{rw_o} = f(T_{r_o}, P_d)$ (4.93)	See Note 1
Riser Fluid Density	Nearly Saturated Fluid	$\rho_{r_o} = f(T_{r_o}, x, P_d,)$ (4.94)	See Note 1
Derivative of Riser Fluid Density	Nearly Saturated Fluid	$\frac{d}{dt}[\rho_{r_o}] = \frac{\delta \rho_{r_o}}{\delta T_{r_o}} \frac{d}{dt}[T_{r_o}] + \frac{\delta \rho_{r_o}}{\delta x} \frac{d}{dt}[x] + \frac{\delta \rho_{r_o}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.95)	See Note 1
Riser Fluid Enthalpy	Nearly Saturated Fluid	$h_{r_o} = f(T_{r_o}, x, P_d)$ (4.96)	See Note 1
Derivative of Riser Fluid Enthalpy	Nearly Saturated Fluid	$\frac{d}{dt}[h_{r_o}] = \frac{\delta h_{r_o}}{\delta T_{r_o}} \frac{d}{dt}[T_{r_o}] + \frac{\delta h_{r_o}}{\delta x} \frac{d}{dt}[x] + \frac{\delta h_{r_o}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.97)	See Note 1
Riser Saturation Temperature	Saturated Fluid	$T_{sat} = f(P_d)$ (4.98)	None
Drum Water Density	Subcooled Water	$\rho_{dw} = f(P_d, T_{dw})$ (4.99)	None
Derivative of Drum Water Density	Subcooled Water	$\frac{d}{dt}[\rho_{dw}] = \frac{\delta \rho_{dw}}{\delta T_{dw}} \frac{d}{dt}[T_{dw}] + \frac{\delta \rho_{dw}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.100)	None

(continued)..

Thermodynamic Property	State	Thermodynamic Relations	Assumptions
Drum Water Enthalpy	Subcooled Water	$h_{dw} = f(P_d, T_{dw})$ (4.101)	None
Derivative of Drum Water Enthalpy	Subcooled Water	$\frac{d}{dt}[h_{dw}] = \frac{\delta h_{dw}}{\delta T_{dw}} \frac{d}{dt}[T_{dw}] + \frac{\delta h_{dw}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.102)	None
Downcomer Water Density	Subcooled Water	$\rho_{do_o} = f(P_d, T_{do_o})$ (4.103)	None
Derivative of Downcomer Water Density	Subcooled Water	$\frac{d}{dt}[\rho_{do_o}] = \frac{\delta \rho_{do_o}}{\delta T_{do_o}} \frac{d}{dt}[T_{do_o}] + \frac{\delta \rho_{do_o}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.104)	None
Downcomer Water Enthalpy	Subcooled Water	$h_{do_o} = f(P_d, T_{do_o})$ ...(4.105)	None
Derivative of Downcomer Water Enthalpy	Subcooled Water	$\frac{d}{dt}[h_{do_o}] = \frac{\delta h_{do_o}}{\delta T_{do_o}} \frac{d}{dt}[T_{do_o}] + \frac{\delta h_{do_o}}{\delta P_d} \frac{d}{dt}[P_d]$ (4.106)	None
Riser Gas Specific Heat at Constant Pressure, Pass 1	Gas	$c_{rg1_o} = f(T_{rg1_o})$ (4.107)	Pressure effect negligible
Riser Gas Density, Pass 1	Gas	$\rho_{rg1_o} = f(T_{rg1_o})$ (4.108)	Pressure effect negligible
Riser Gas Specific Heat at Constant Pressure, Pass 2	Gas	$c_{rg2_o} = f(T_{rg2_o})$ (4.109)	Pressure effect negligible
Riser Gas Density, Pass 2	Gas	$\rho_{rg2_o} = f(T_{rg2_o})$ (4.110)	Pressure effect negligible
Riser Metal Specific Heat at Constant Pressure, Pass 1	Solid	$c_{rm1} = c_m$ (4.111)	Pressure and temperature effect negligible
Riser Metal Density, Pass 1	Solid	$\rho_{rm1} = \rho_m$ (4.112)	Pressure and temperature effect negligible
Riser Metal Specific Heat at Constant Pressure, Pass 2	Solid	$c_{rm2} = c_m$ (4.113)	Pressure and temperature effect negligible
Riser Metal Density, Pass 2	Solid	$\rho_{rm2} = \rho_m$ (4.114)	Pressure and temperature effect negligible

Table 4.11 Thermodynamic Relationships for Drum, Downcomers and Risers

Note 1: A fluid in saturated equilibrium is a function of two thermodynamic variables only - any thermodynamic property such as pressure or temperature and a second variable which represents the relative proportions of water and steam present. It has not been assumed that the fluid in the risers is constantly in saturated equilibrium. Instead it is assumed

that the riser fluid temperature lags the saturation temperature which corresponds to drum pressure. The thermodynamic properties of the saturated fluid are obtained by using the saturation properties corresponding to riser fluid temperature and adding a term which represents the effect of drum pressure.

4.3.2.5 Equations for Dependent Variables (Drum, Downcomers and Risers)

1. *Riser Gas-Metal Heat Transfer Rate, Pass 1*

In the combustion chamber, heat is assumed to be transferred from the riser gases to the first bank of riser tubes by radiation. The equation for calculating the rate of heat transfer is empirical (Kwan and Anderson (1970)).

$$Q_{rgm1} = k_{rgm1}(T_{rg1_o}{}^4 - T_{rm1}{}^4) \tag{4.115}$$

2. *Riser Gas-Metal Heat Transfer Rate, Pass 2*

Heat transfer from riser gas to metal - Q_{rgm2} is assumed to be convective at the second bank of riser tubes. The rate of heat transfer is calculated using an empirical equation (Kwan and Anderson (1970)).

$$Q_{rgm2} = k_{rgm2} m_g{}^{0.6}(T_{rg2_o} - T_{rm2}) \tag{4.116}$$

3. *Riser Metal-Fluid Heat Transfer Rate, Pass 1*

Metal-fluid heat transfer is assumed to be radiative at the first bank of riser tubes. It is calculated using the following empirical equation (Kwan and Anderson (1970)).

$$Q_{rmf1} = k_{rmf1}(T_{rm1} - T_{r_o})^3 \tag{4.117}$$

4. *Riser Metal-Fluid Heat Transfer Rate, Pass 2*

The heat transfer from the riser metal to fluid is assumed to be convective at the second bank of riser tubes. The following empirical equation is used to calculate the rate of heat transfer (Kwan and Anderson (1970)).

$$Q_{rmf2} = k_{rmf2} m_r{}^{0.8}(T_{rm2} - T_{r_o}) \tag{4.118}$$

4.3.2.6 Summary of Variables (Drum, Downcomers and Risers)

Inputs	
Feedwater Mass Flow Rate	m_{fw}

Table 4.12 Drum, Downcomers and Risers Inputs

System Parameters	
Drum-Superheater Frictional Loss	H_LOSS_D
Riser Gas-Metal Heat Transfer Coefficient, Pass 1	k_{rgm1}
Riser Gas-Metal Heat Transfer Coefficient, Pass 2	k_{rgm2}
Riser Metal-Fluid Heat Transfer Coefficient, Pass 1	k_{rmf1}
Riser Metal-Fluid Heat Transfer Coefficient, Pass 2	k_{rmf2}
Recirculation Mass Flow Rate	m_r
Number of Superheater Tubes	N_s
Saturation Equilibrium Time Constant	τ_{eq}
Volume of Downcomers	V_{do}
Volume of Drum	V_{dr}
Volume of Risers	V_r
Volume of Riser Gas, Pass 1	V_{rg1}
Volume of Riser Gas, Pass 2	V_{rg2}
Volume of Riser Metal, Pass 1	V_{rm1}
Volume of Riser Metal, Pass 2	V_{rm2}

Table 4.13 Drum, Downcomers and Risers System Parameters

State Variables	
Riser Gas Temperature, Pass 1	T_{rg1_o}
Riser Metal Temperature, Pass 1	T_{rm1}
Riser Gas Temperature, Pass 2	T_{rg2_o}
Riser Metal Temperature, Pass 2	T_{rm2}
Riser Fluid Temperature	T_{r_o}
Steam Quality	x
Drum Steam Temperature	T_{ds}
Drum Water Temperature	T_{dw}
Downcomer Water Temperature	T_{do_o}
Drum Water Volume	V_w

Table 4.14 Drum, Downcomers and Risers State Variables

4.3.3 Superheater

4.3.3.1 Lumped Systems (Superheater)

Saturated steam from the drum enters the superheater and is superheated to the required temperature. The superheater model comprises the following lumped systems.

1. Superheater gas path
2. Superheater metal
3. Superheater fluid

4.3.3.2 Assumptions (Superheater)

1. Steam is superheated
2. Steam has negligible inertia
3. Compressibility of steam is assumed to be negligible with respect to calculating steam velocity
4. No heat is lost through furnace walls

4.3.3.3 Continuity and State Variable Equations (Superheater)

Superheater Gas Mass Balance

The mass flow rate of combustion gas past the superheater is assumed to be static and equal to the mass generation rate of combustion gas.

$$m_{sg_o} = m_g \tag{4.119}$$

Superheater Gas Energy Balance

Hot combustion gases flow from the first bank of risers tubes and are directed by baffles over the superheater tubes. The gases impart a portion of their energy to the superheater as they flow over the superheater tubes.

$$\frac{d}{dt}[\rho_{sg_o} V_{sg} c_{sg} T_{sg_o}] = c_{sg}(T_{rgl_o} - T_{sg_o}) - Q_{sgm} \tag{4.120}$$

Superheater Gas Temperature

The superheater gas energy balance is rewritten in terms of the superheater gas temperature.

$$\frac{d}{dt}[T_{sg_o}] = \frac{Q_{sgm} - Q_{smf}}{\rho_{sg_o} V_{sg} c_{sg}} \tag{4.121}$$

Superheater Metal Energy Balance

The superheater metal energy receives energy from the superheater gas path and transfers energy to the superheated steam in the superheater.

$$\frac{d}{dt}[\rho_{sm}V_{sm}c_{sm}T_{sm}] = Q_{sgm} - Q_{smf} \tag{4.122}$$

Superheater Metal Temperature

The superheater metal energy balance is rewritten in terms of the superheater metal temperature.

$$\frac{d}{dt}[T_{sm}] = \frac{Q_{sgm} - Q_{smf}}{\rho_{sm}V_{sm}c_{sm}} \tag{4.123}$$

Superheater Fluid Mass Balance

The mass of fluid in the superheater is determined by the difference between the mass flow rate into the superheater and the mass flow rate out of the superheater

$$\frac{d}{dt}[V_s\rho_{s_o}] = m_{s_i} + m_{at} - m_{s_o} \tag{4.124}$$

The superheater steam density is a function of steam pressure and temperature. The fluid mass balance can be expanded using the product rule and chain rule in terms of pressure and temperature.

$$V_{su}\frac{\delta\rho_{s_o}}{\delta P_{s_o}}\frac{dP_{s_o}}{dt} + V_{su}\frac{\delta\rho_{s_o}}{\delta T_{s_o}}\frac{dT_{s_o}}{dt} = m_{s_i} + m_{at} - m_{s_o} \tag{4.125}$$

It can be rewritten for brevity as:

$$a_{11}\frac{dP_{s_o}}{dt} + a_{12}\frac{dT_{s_o}}{dt} = k_1$$

where

$$a_{11} = V_{su}\frac{\delta\rho_{s_o}}{\delta P_{s_o}}, \quad a_{12} = V_{su}\frac{\delta\rho_{s_o}}{\delta T_{s_o}} \text{ and } k_1 = m_{s_i} + m_{at} - m_{s_o} \tag{4.126}$$

Superheater Fluid Energy Balance

The steam entering the superheater has the same enthalpy as the steam in the drum. The steam leaving the superheater has a higher enthalpy than the steam in the drum due to the transfer of

energy from the superheater metal to the superheater fluid.

$$\frac{d}{dt}[V_s \rho_{s_o} h_{s_o}] = m_{s_i} h_{ds} + m_{at} h_{fw} - m_{s_o} h_{s_o} + Q_{smf} \tag{4.127}$$

The superheated steam density and enthalpy are both functions of the temperature and pressure. The fluid energy balance can be expanded and rewritten in terms of density and enthalpy.

$$\begin{aligned} V_{su}[h_{s_o} \frac{\delta \rho_{s_o}}{\delta P_{s_o}} + \rho_{s_o} \frac{\delta h_{s_o}}{\delta P_{s_o}}] \frac{dP_{s_o}}{dt} + V_{su}[h_{s_o} \frac{\delta \rho_{s_o}}{\delta T_{s_o}} + \rho_{s_o} \frac{\delta h_{s_o}}{\delta T_{s_o}}] \frac{dT_{s_o}}{dt} \\ = m_{s_i} h_{ds} + m_{at} h_{fw} - m_{s_o} h_{s_o} + Q_{smf} \end{aligned} \tag{4.128}$$

This can be rewritten more concisely as:

$$a_{21} \frac{dP_{s_o}}{dt} + a_{22} \frac{dT_{s_o}}{dt} = k_2$$

where

$$a_{21} = V_{su}[\rho_{s_o} \frac{\delta h_{s_o}}{\delta P_{s_o}} + h_{s_o} \frac{\delta \rho_{s_o}}{\delta P_{s_o}}], \quad a_{22} = V_{su}[\rho_{s_o} \frac{\delta h_{s_o}}{\delta T_{s_o}} + h_{s_o} \frac{\delta \rho_{s_o}}{\delta T_{s_o}}] \tag{4.129}$$

and $k_2 = m_{s_i} h_{ds} + m_{at} h_{fw} - m_{s_o} h_{s_o} + Q_{smf}$

The superheater steam temperature and superheater steam pressure were chosen as state variables for two reasons. Firstly, both variables are included in other model equations e.g. momentum balances and heat transfer equations. Secondly, both variables are desirable model outputs.

The fluid mass and energy balances are solved to yield expressions for the derivatives of pressure and temperature:

Superheater Steam Pressure

$$\frac{dP_{s_o}}{dt} = \frac{k_1 - a_{12} \frac{k_2}{a_{22}}}{a_{11} - a_{21} \frac{a_{12}}{a_{22}}} \tag{4.130}$$

Superheater Steam Temperature

$$\frac{dP_{s_o}}{dt} = \frac{k_1 - a_{11} \frac{dP_d}{dt}}{a_{12}} \tag{4.131}$$

Superheater Fluid Outlet Momentum Balance

The continuity of momentum equation is used to calculate the velocity of steam from the superheater. The compressibility of the steam is assumed to be negligible, on the basis that the velocity of steam in the superheater is considerably lower than acoustic velocities. The equation is used in steady-state format as the inertia of the fluid is assumed to be negligible.

$$\frac{P_{s_o}}{\rho_{ds} g} + \frac{v_{s_i}^2}{2g} - \frac{P_i}{\rho_{ds} g} = \frac{v_{s_o}^2}{2g}(H\_LOSS\_S) \qquad (4.132)$$

Super-heater Pressure Head	+	Inlet Kinetic Energy	-	Load Pressure Head	=	Outlet Kinetic Energy	×	Super-heater Frictional Loss

The mass flow rate of steam is calculated using the steady-state continuity of mass equation.

$$m_{s_o} = N_s v_{s_o} A_s \rho_{s_o} \qquad (4.133)$$

4.3.3.4 Thermodynamic State Equations (Superheater)

The relationship between the superheater thermodynamic properties must be known to solve the equations for the superheater state variables. Table 4.15 gives the assumed state of each thermodynamic variable and its relationship to the state variables. It also highlights any other simplifying assumptions.

Thermodynamic Property	State	Thermodynamic Relations	Assumptions
Superheater Steam Density	Superheated Steam	$\rho_{s_o} = f(P_{s_o} T_{s_o})$ (4.134)	None
Derivative of Superheater Steam Density	Superheated Steam	$\frac{d}{dt}[\rho_{s_o}] = \frac{\delta \rho_{s_o}}{\delta T_{s_o}} \frac{d}{dt}[T_{s_o}] + \frac{\delta \rho_{s_o}}{\delta P_{s_o}} \frac{d}{dt}[P_{s_o}]$ (4.135)	None
Superheater Steam Enthalpy	Superheated Steam	$h_{s_o} = f(P_{s_o} T_{s_o})$ (4.136)	None
Derivative of Superheater Steam Enthalpy	Superheated Steam	$\frac{d}{dt}[h_{s_o}] = \frac{\delta h_{s_o}}{\delta T_{s_o}} \frac{d}{dt}[T_{s_o}] + \frac{\delta h_{s_o}}{\delta P_{s_o}} \frac{d}{dt}[P_{s_o}]$ (4.137)	None
Superheater Gas Specific Heat at Constant Pressure	Gas	$c_{sg_o} = f(T_{sg_o})$ (4.138)	Pressure effect negligible
Superheater Gas Density	Gas	$\rho_{sg_o} = f(T_{sg_o})$ (4.139)	Pressure effect negligible
Superheater Metal Specific Heat at Constant Pressure	Solid	$c_{sml} = c_m$ (4.140)	Pressure and temperature effect negligible
Superheater Metal Density	Solid	$\rho_{sml} = \rho_m$ (4.141)	Pressure and temperature effect negligible

Table 4.15 Thermodynamic Relationships for Superheater

4.3.3.5 Equations for Dependent Variables (Superheater)

1. Superheater Gas-Metal Heat Transfer Rate

The is calculated using an empirical equation for convective heat transfer (Kwan and Anderson (1970)).

$$Q_{sgm} = k_{sgm} m_g^{0.6} (T_{sg_o} - T_{sm}) \tag{4.142}$$

2. Superheater Metal-Fluid Heat Transfer Rate

The heat transfer from the superheater metal to the superheater steam is calculated using an empirical equation for convective heat transfer (Kwan and Anderson (1970)).

$$Q_{smf} = k_{smf} m_{s_o}^{0.8} (T_{sm} - T_{s_o}) \tag{4.143}$$

4.3.3.6 Summary of Variables (Superheater)

Inputs	
Attemperating Water Mass Flow Rate	m_{at}
Temperature of Feedwater	h_{fw}

Table 4.16 Superheater Inputs

System Parameters	
Area of Superheater Tubes	A_s
Superheater Gas-Metal Heat Transfer Coefficient	k_{sgm}
Superheater Metal-Fluid Heat Transfer Coefficient	k_{smf}
Number of Superheater Tubes	N_s
Superheater Exit Loss	H_LOSS_S
Volume of Superheater Fluid	V_{su}
Volume of Superheater Gas	V_{sg}
Volume of Superheater Metal	V_{sm}

Table 4.17 Superheater System Parameters

State Variables	
Superheater Gas Path Temperature	T_{sg_o}
Superheater Metal Temperature	T_{sm}
Superheater Steam Pressure	P_{s_o}
Superheated Steam Temperature	T_{s_o}

Table 4.18 Superheater State Variables

4.3.4 Downstream Process

4.3.4.1 Lumped Systems (Downstream Process)

The downstream process is any arbitrary process which utilises the generated steam. It is modelled by a short section of tubes lying after the superheater which is terminated by a bank of valves. The load demand is varied by increasing and decreasing the number of open valves - N_v in the bank. The downstream process model comprises one lumped system.

1. Downstream process fluid

4.3.4.2 Assumptions (Downstream Process)

1. Steam is superheated
2. Compressibility of steam is negligible with respect to calculation of steam velocity.
3. Steam has negligible inertia
4. No heat is absorbed by the steam after it leaves the superheater.

4.3.4.3 Continuity and State Variable Equations (Downstream Process)

Downstream Process Fluid Mass Balance

The mass of steam stored by the downstream process is determined by the difference between the mass flow rate into the process and the mass flow rate out of the superheater

$$\frac{d}{dt}[V_l \rho_{l_o}] = m_{s_o} - m_{l_o} \tag{4.144}$$

The steam mass balances are rewritten in terms of steam pressure and temperature. Firstly the steam mass and energy balances are expanded using the product rule. Density and enthalpy are then substituted with expressions for pressure and temperature.

$$V_l \frac{\delta \rho_{l_o}}{\delta P_{l_o}} \frac{dP_{l_o}}{dt} + V_l \frac{\delta \rho_{l_o}}{\delta T_{l_o}} \frac{dT_{l_o}}{dt} = m_{s_o} - m_{l_o} \tag{4.145}$$

For brevity, the equation can be rewritten as:

$$a_{11} \frac{dP_{s_o}}{dt} + a_{12} \frac{dT_{s_o}}{dt} = k_1$$

where

$$a_{11} = V_l \frac{\delta \rho_{ls_o}}{\delta P_{l_o}}, \quad a_{12} = V_l \frac{\delta \rho_{ls_o}}{\delta T_{l_o}} \quad \text{and} \quad k_1 = m_{s_o} - m_{l_o} \tag{4.146}$$

Downstream Process Fluid Energy Balance

The energy of steam stored by the downstream process is determined by the difference between the energy flow rate into the process and the energy flow rate out of the process.

$$\frac{d}{dt}[V_l \rho_{l_o} h_{l_o}] = m_{s_o} h_{s_o} - m_{l_o} h_{l_o} \tag{4.147}$$

This equation is rewritten in terms of steam pressure and temperature.

$$V_l[h_{l_o} \frac{\delta \rho_{l_o}}{\delta P_{l_o}} + \rho_{l_o} \frac{\delta h_{l_o}}{\delta P_{l_o}}] \frac{dP_{l_o}}{dt} + V_l[h_{l_o} \frac{\delta \rho_{l_o}}{\delta T_{l_o}} + \rho_{l_o} \frac{\delta h_{l_o}}{\delta T_{l_o}}] \frac{dT_{l_o}}{dt} = m_{s_o} h_{s_o} - m_{l_o} h_{l_o} \tag{4.148}$$

For brevity, the equation can be rewritten as:

$$a_{21} \frac{dP_{l_o}}{dt} + a_{22} \frac{dT_{l_o}}{dt} = k_2$$

where

$$a_{21} = V_l[\rho_{l_o} \frac{\delta h_{l_o}}{\delta P_{l_o}} + h_{l_o} \frac{\delta \rho_{ls_o}}{\delta P_{l_o}}], \quad a_{22} = V_l[\rho_{l_o} \frac{\delta h_{l_o}}{\delta T_{l_o}} + h_{l_o} \frac{\delta \rho_{ls_o}}{\delta T_{l_o}}] \tag{4.149}$$

and $k_2 = m_{s_o} h_{s_o} - m_{l_o} h_{l_o}$

Equations 4.146 and 4.149 are solved for the derivatives of pressure and temperature:

Downstream Process Steam Pressure

$$\frac{dP_{l_o}}{dt} = \frac{k_1 - a_{12} \frac{k_2}{a_{22}}}{a_{11} - a_{21} \frac{a_{12}}{a_{22}}} \tag{4.150}$$

Downstream Process Steam Temperature

$$\frac{dT_{l_o}}{dt} = \frac{k_1 - a_{11} \frac{dP_{l_o}}{dt}}{a_{12}} \tag{4.151}$$

Downstream Process Momentum Balance

The continuity of momentum equation is used to calculate the mass flow rate of steam through the downstream process. The steam is assumed to be incompressible on the basis that the

velocity of steam is considerably lower than acoustic velocities. The equation is used in steady-state format as the inertia of the fluid is assumed to be negligible.

$$m_{l_o} = k_v N_v \sqrt{P_{l_o} - P_{low}} \tag{4.152}$$

4.3.4.4 Thermodynamic State Equations (Downstream Process)

Thermodynamic Property	State	Thermodynamic Relations	Assumptions
Downstream Process Steam Density	Superheated Steam	$\rho_{l_o} = f(P_{l_o} T_{l_o})$ (4.153)	None
Derivative of Downstream Process Steam Density	Superheated Steam	$\frac{d}{dt}[\rho_{l_o}] = \frac{\delta \rho_{l_o}}{\delta T_{l_o}} \frac{d}{dt}[T_{l_o}] + \frac{\delta \rho_{l_o}}{\delta P_{l_o}} \frac{d}{dt}[P_{l_o}]$ (4.154)	None
Downstream Process Steam Enthalpy	Superheated Steam	$h_{l_o} = f(P_{l_o} T_{l_o})$ (4.155)	None
Downstream Process Steam Enthalpy	Superheated Steam	$\frac{d}{dt}[h_{l_o}] = \frac{\delta h_{l_o}}{\delta T_{l_o}} \frac{d}{dt}[T_{l_o}] + \frac{\delta h_{l_o}}{\delta P_{l_o}} \frac{d}{dt}[P_{l_o}]$ (4.156)	None

Table 4.19 Thermodynamic Relationships for Downstream Process

4.3.4.5 Summary of Variables (Downstream Process)

Inputs	
Number of open valves (load demand)	N_v

Table 4.20 Downstream Process Inputs

System Parameters	
Valve mass flow rate coefficient	k_v
Pressure behind bank of valves	P_{low}
Volume of downstream process	V_l

Table 4.21 Downstream Process System Parameters

State Variables	
Downstream Process Pressure	P_{l_o}
Downstream Process Temperature	T_{l_o}

Table 4.22 Downstream Process State Variables

4.4 Model Implementation and Testing

4.4.1 Implementation

Implementation Environment

The model is implemented in the MATLAB environment (MATLAB User's Guide (1993), SIMULINK User's Guide (1993)). MATLAB is a computing environment for tasks such as matrix computation and numerical analysis. It has its own command language and numerous built-in functions, including SIMULINK - a package for simulating dynamical systems.

The models is implemented are first defined in SIMULINK and then simulated. The model definition is a combination of both built-in SIMULINK blocks and user defined blocks.

User Interface

The user can interface with the model in one of two ways:

1. SIMULINK User Interface

   A diagram of the user interface is shown in Fig. 4.4. The user can set the form (ramp, step, sine wave) and value of the model inputs by choosing and editing built-in SIMULINK blocks via a graphical interface. The simulation parameters (e.g. start time, stop time) are set and simulations are started via a menu interface.

2. Parameter Definition File

   All of the model parameters are defined in a single text file which is loaded at the start of simulation. The user can change parameter settings by editing this file

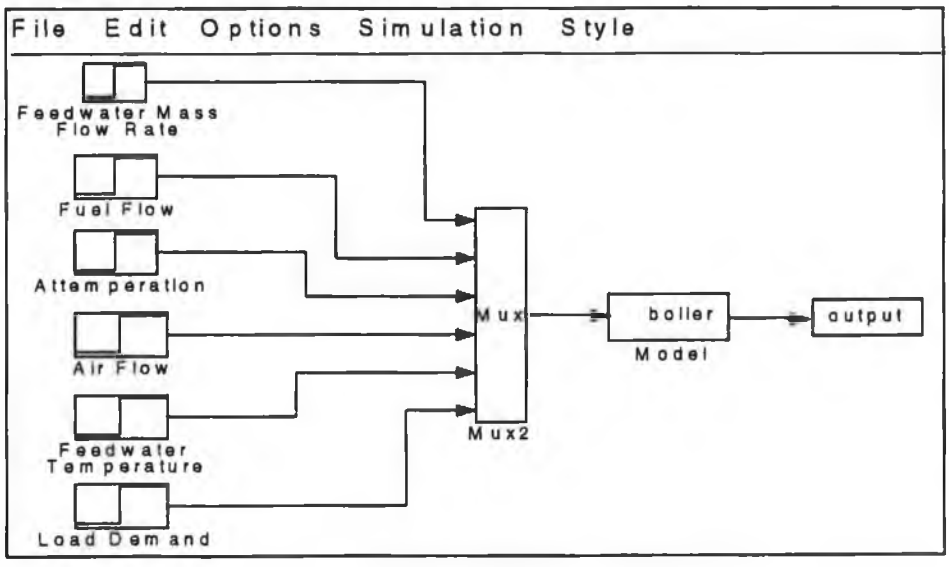


Fig. 4.4 Model User Interface

Model Implementation

The main body of code for the boiler model is contained in a user-defined block. This file calculates the state variables and other required outputs. It includes three types of equations

1. Equations for derivatives of state variables
2. Static equations for mass flow rate, heat transfer rate and drum level
3. Implementation of thermodynamic state equations.

Thermodynamic relations are implemented for:

- Subcooled Water
- Saturated (or nearly saturated) fluid
- Superheated steam
- Gas

The thermodynamic relations for subcooled water, saturated fluid and superheated steam are implemented using a large number of polynomial functions. For example, for superheated steam enthalpy, a fourth order polynomial is fitted to the steam tables which relates enthalpy to steam temperature at a particular pressure. A set of these polynomials is required to cover the entire pressure range of the boiler (usually one every 0.5 MPa). A combination of polynomial evaluation and interpolation is used to find the value of enthalpy which corresponds to a particular temperature and pressure. Checks are also included to ensure that the steam is in fact superheated. This method is extremely accurate - the calculated variables generally deviate from the steam table value by less than 0.1%

The gas thermodynamic properties are calculated by fitting a single polynomial at one pressure. Pressure variations do not have an important effect on gas density or enthalpy as the forced and induced draft fans hold pressure relatively constant.

4.4.2 Simulation

SIMULINK provides a number of integration tools for simulating models: The Gear predictor corrector method (Kahaner *et al* (1989)) is used to simulate the model as it has been designed to simulate stiff, nonlinear systems. The boiler model is highly nonlinear and stiff. For example, the time constants associated with metal lumped systems are considerably lower than the time constants for the fluid lumped systems.

It is not necessary to select a simulation step size as the Gear method is a predictor-corrector method, and uses a variable number of steps between outputs.

4.4.3 Test Procedure

It is very important to perform some preliminary model testing prior to model validation. Testing merely involves checking that model equations have been implemented correctly and that the simulation results are as expected. However, testing does not verify if the model accurately represents the behaviour of a real boiler, which is the function of validation. For a model of this complexity, testing is vital as it eliminates software errors before validation is undertaken.

The aim of testing is to verify that:

- The equations for dependent variables are coded correctly. This was checked by visually examining the code.
- The continuity equations are solved correctly and the resulting equations for state variables are coded correctly.
- All the thermodynamic state equations have been implemented correctly. This ensures that the substitution of thermodynamic variables are correct e.g. pressure and temperature for density and enthalpy.

The following procedure was devised to verify the last two statements in the above list for each continuity equation. It is described here as it was applied to the drum water energy balance.

1. The rate of change of drum water energy is forced equal to zero by setting the input energy equal to the output energy. This is done by manipulating the enthalpy of the incoming feedwater.

$$\begin{aligned} &\frac{d}{dt}[V_{dw}\rho_{dw}h_{dw}] = m_{e_o}h_{e_o} + (1-x)m_r h_{rw_o} - m_r h_{dw} + Q_{dmf} = 0 \\ &\Rightarrow h_{e_o} = \frac{1}{m_{e_o}}(m_r h_{dw} - (1-x)m_r h_{rw_o} - Q_{dmf}) \end{aligned} \tag{4.157}$$

2. The model is simulated and the following simulation outputs are saved in a file: drum water volume - V_{dw}, drum water density - ρ_{dw} and drum water enthalpy - h_{dw}.

3. The value of fluid energy during simulation is calculated by multiplying drum water volume, drum water density and drum water enthalpy.

$$E_{dw} = V_{dw}\rho_{dw}h_{dw} \tag{4.158}$$

If the resulting energy - E_{dw} is constant throughout simulation this demonstrates that drum water volume, density and enthalpy have all been calculated correctly. This implies that the model equations have been implemented correctly in this respect.

4.4.4 Test Results

4.4.4.1 Combustion Side

The equations used to model the combustion-side of the boiler are in general, less complex than the fluid-side equations. In particular, the equations used to model heat generation were simple to implement and test. Consequently, only simulation results for the model of the combustion gas composition will be given. Fig 4.5 shows the effect of increasing the percentage excess air on the percentage of O_2 and CO_2 in the combustion gases.

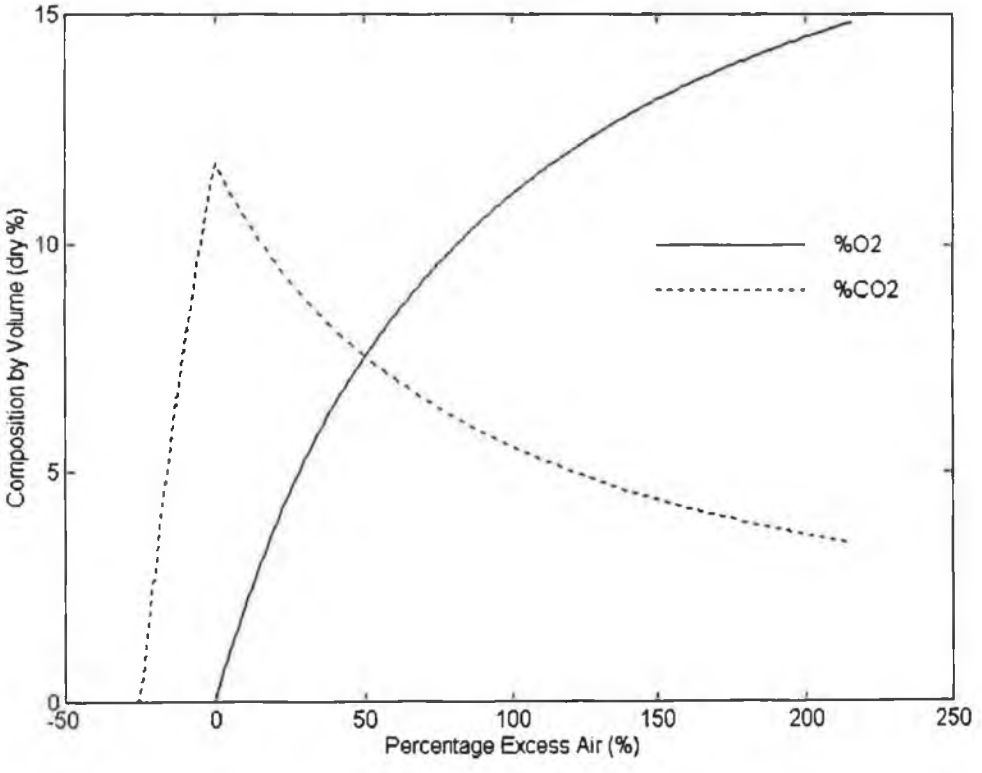


Fig. 4.5 Flue Gas Composition for Natural Gas *versus* Excess Air

4.4.4.2 Fluid Side

The effect of step variations in the model inputs will be examined. In each case, it can be seen that the model response is as expected.

Response to a step change in feedwater flow - m_{fw}

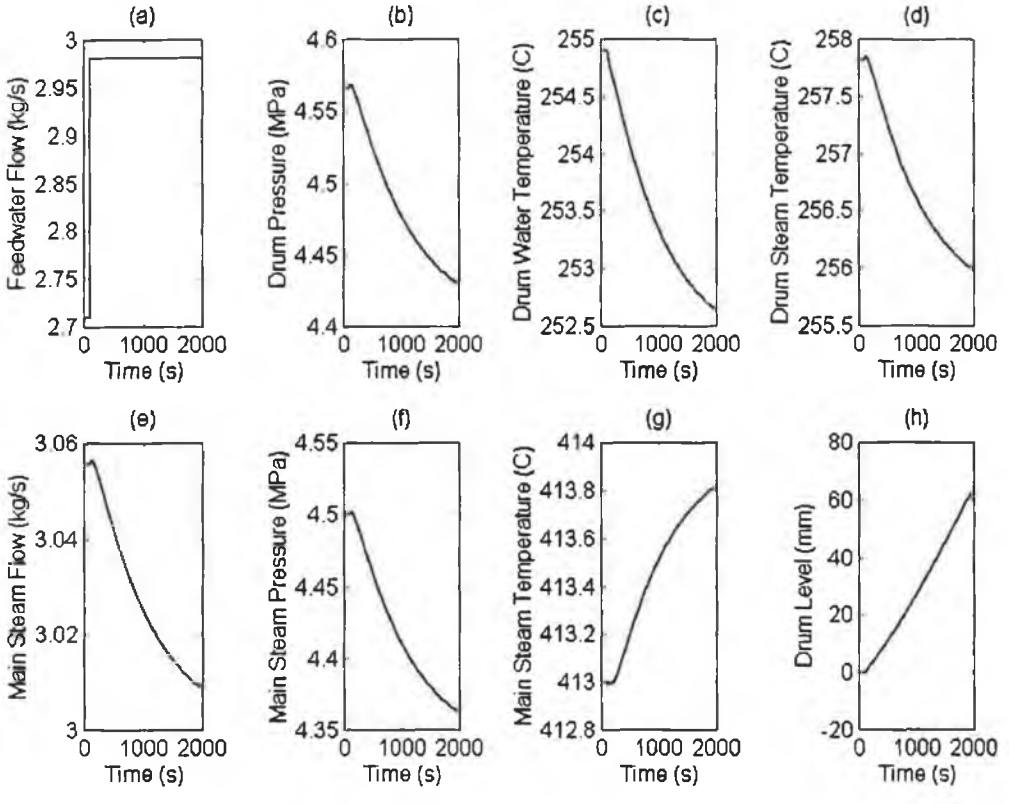


Fig. 4.6 Response to a step change in feedwater flow

An increase in the flow rate of the subcooled feedwater (Fig. 4.6a) results in a decrease in drum water temperature (Fig. 4.6c). This cooler drum water eventually reaches the risers and causes a decrease in the rate of evaporation and, consequently, in the rate of steam flow from the boiler (Fig. 4.6e). As the rate of evaporation decreases, the drum pressure (Fig. 4.6b) and superheater pressure (Fig. 4.6f) can also be expected to decrease. The evaporation temperature of water decreases with pressure, which results in a decrease in the temperature of steam in the drum (Fig. 4.6d). The drum water level (Fig. 4.6g) can be expected to rise as the feedwater flow rate is now greater than the steam flow rate from the drum. The superheater steam temperature (Fig. 4.6h) increases because the mass flow rate of steam through the superheater has decreased while the amount of heat available from the flue gases remains practically constant.

Response to a step change in fuel flow - m_{fu}

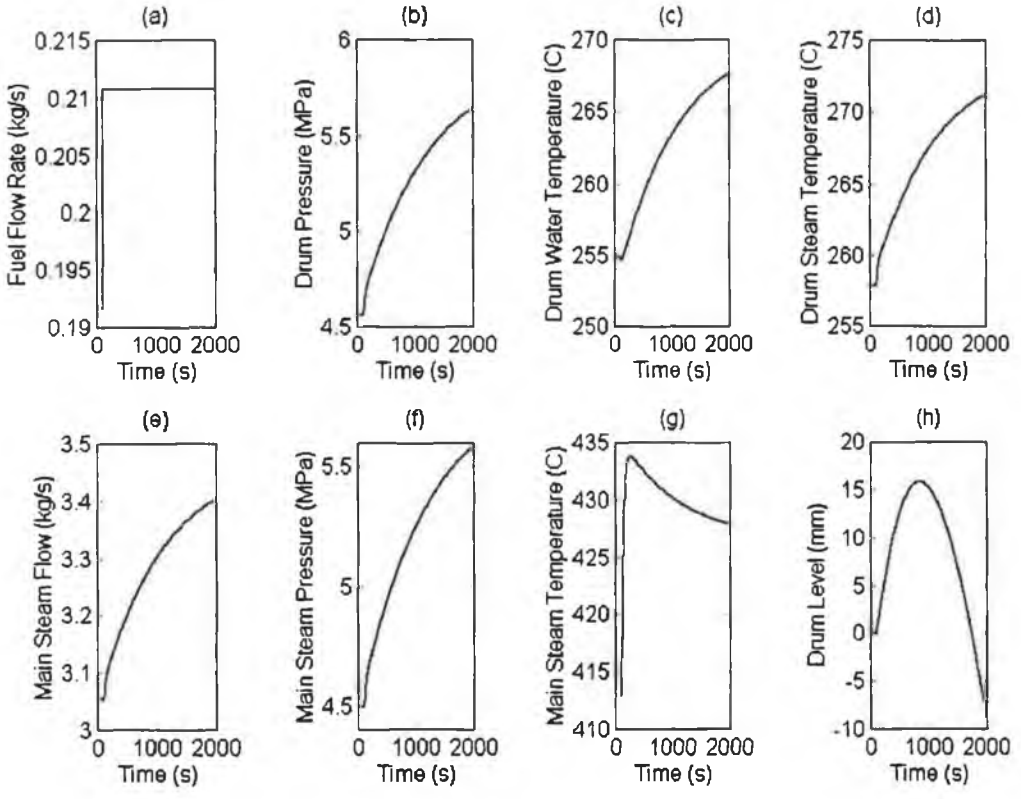


Fig. 4.7 Response to a step change in fuel flow

An increase in the fuel flow rate (Fig. 4.7a) causes an increase in the rate of evaporation in the risers. Increased evaporation leads to an increase in drum (Fig. 4.7b) and superheater pressure (Fig. 4.7f), and to an increase in steam flow (Fig. 4.7e) from the boiler. The drum level (Fig. 4.7h) first increases due to the decrease in the density of the water and steam in the drum. It then decreases as a result of the increased rate of evaporation. This phenomenon is known as '*Shrink and Swell*'. The temperature of the fluid entering the drum from the risers increases and causes a corresponding increase in the drum water temperature (Fig. 4.7c) and drum steam temperature (Fig. 4.7d). The main steam temperature (Fig. 4.7g) decreases because the mass flow rate of steam through the superheater has increased, while the heat flow rate to the superheater has not increased significantly.

Response to a step change in the flow rate of attemperating water - m_{at}

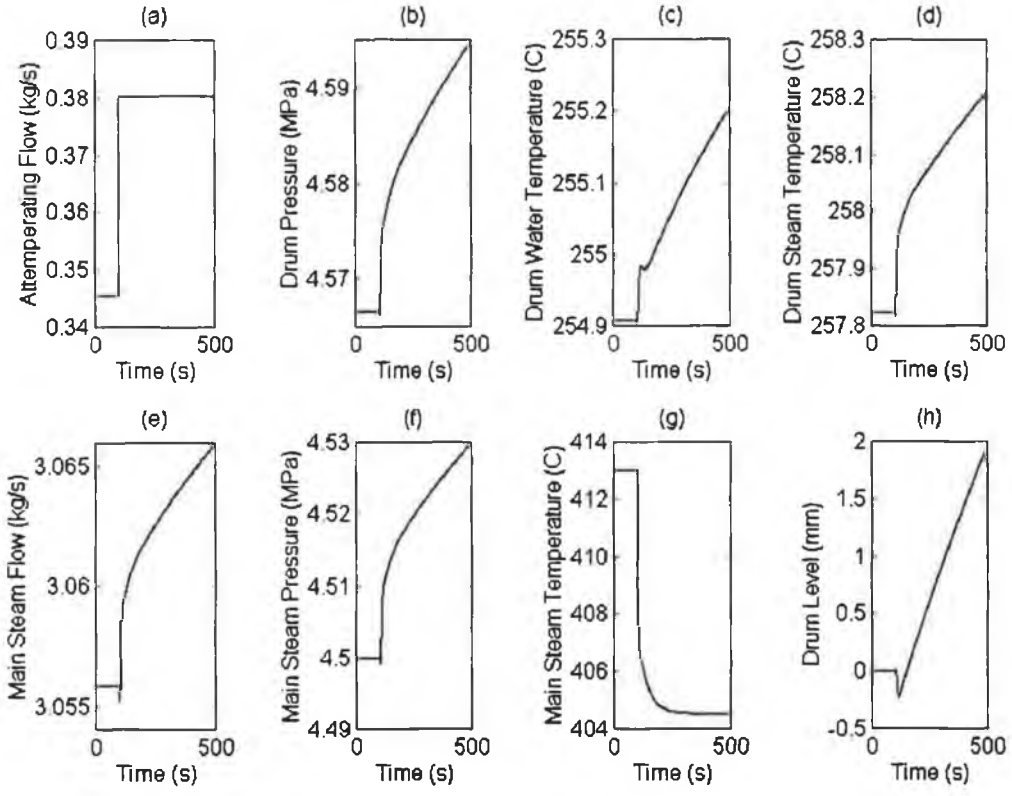


Fig. 4.8 Response to a step change in the attemperating flow rate

A step increase in the attemperating water mass flow rate (Fig. 4.8a) causes an immediate decrease in the main steam temperature. (Fig. 4.8g) Main steam pressure (Fig. 4.8f) increases as the mass flow rate of fluid in to the superheater has been increased by the increase in the attemperating flow rate. This increase in main steam pressure causes a corresponding increase in main steam flow (Fig. 4.8e). The increase in the main steam pressure also reduces the mass flow rate of steam from the drum causing an increase in drum pressure (Fig. 4.8b), drum steam temperature (Fig. 4.8c), drum water temperature (Fig. 4.8d) and drum level (Fig. 4.8h).

Response to a step change in load demand - N_v

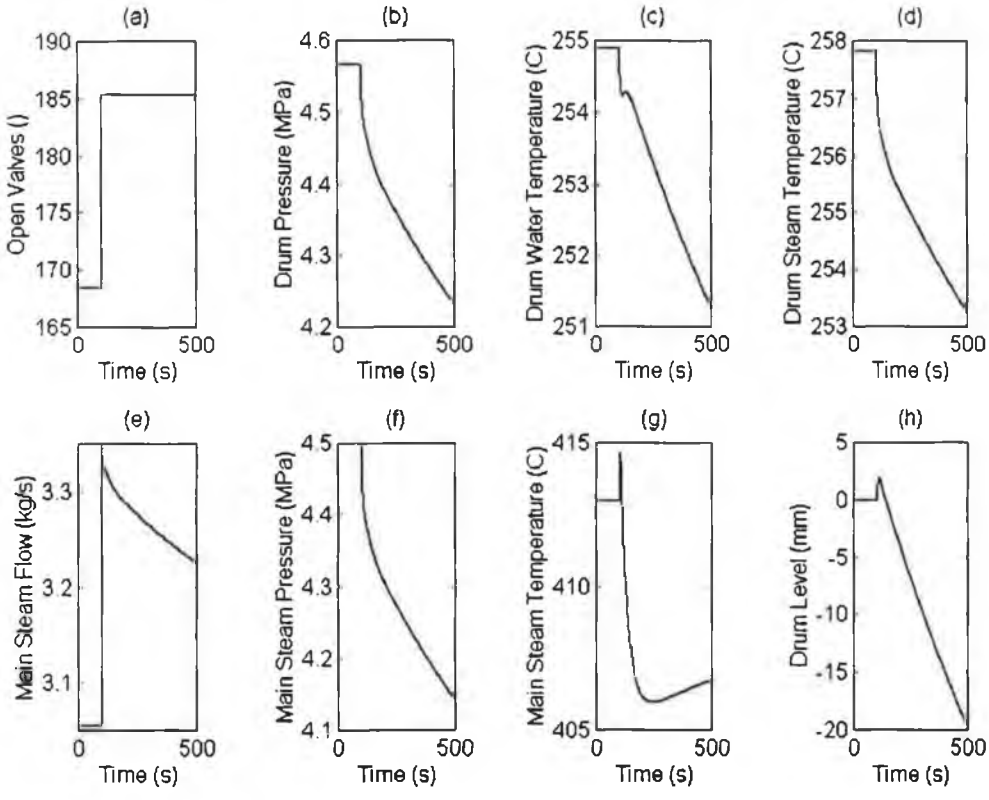


Fig. 4.9 Response to a step change in load demand

A step increase in the number of open valves (Fig. 4.9a) causes a sharp increase in the steam flow rate (Fig. 4.9e) from the boiler. (The flow rate then starts to reduce as the drum steam reserves are depleted. The increase in steam flow rate results in a decrease in main steam (Fig. 4.9f) and drum pressure (Fig. 4.9b). Main steam temperature (Fig. 4.9g) has a very small and short-lived increase, due to the sudden pressure decrease. It then decreases as steam flow rate through the superheater has increased. Drum water temperature (Fig. 4.9c) and drum steam temperature (Fig. 4.9d) decrease because the saturation temperature of the fluid entering from the riser decreases with the decrease in drum pressure. Drum level (Fig. 4.9h) decreases as the flow rate of steam from the drum exceeds the feedwater flow rate. However, it can be seen that there is a temporary increase in drum level, due to the decrease in density of the water in the drum and downcomers and the water-steam mixture in the risers. This is a non-minimum phase effect referred to as '*Shrink and Swell*'.

4.5 Model Validation

4.5.1 Overview

The validity of simulation results depends heavily on model validity. Little credence can be given to simulation results, unless the model has first been validated by comparison to the actual process. Validation provides some measure of the accuracy of model simulation results. It is important to note that validation only provides a *measure* of accuracy because it is not possible to compare the model and the process for all possible input data sets. The validity of the model can only be absolutely guaranteed for the set of inputs and conditions which are included in the model validation data set. Nevertheless, it is possible to have reasonable confidence in the model over all operating conditions, provided the model validation input set has been well chosen.

A good validation data set tests the model over the full process operating range. For example, it should include input data sets, at specific operating points over the operating range e.g. 10%, 30%. At each of these operating points, the data set should include variations in each of the model inputs. Finally the process should be operating in open-loop mode while the process validation measurements are collected, so that there will be no feedback correlation between model output and input data.

It was not possible to obtain a data validation set which matched the above requirements for several reasons. The boiler in question was operating on-line, producing steam for several processes downstream. The boiler load was changing constantly as these processes came on- and off-line. It was not possible to set the boiler load at a particular level without disrupting these processes. Likewise, it was not considered feasible to operate the boiler in open-loop mode.

The validation data set was chosen from previously logged boiler operational data. It covers a 4 hour time interval, during which time there were considerable changes in boiler load and the boiler inputs. This ensured that the model was tested as fully as is possible with closed loop data. The available validation set included measurements of the following model inputs:

<u>Measured Boiler Inputs</u>

1. Feedwater flow
2. Feedwater temperature
3. Fuel flow
4. Air flow

The validation set does not include a measurement for superheater attemperation. More importantly,

it does not include a measurement for boiler load demand, which is constantly changing. Changes in boiler load demand result in a change in steam flow (load). One of the first manifestations of a load demand change is a change in pressure downstream from the superheater outlet. A number of boilers (about 8) feed steam into a common steam header. The pressure in this header is the downstream pressure as referred to above. If the load demand increases, downstream pressure will decrease. If the load demand decreases, downstream pressure will increase.

The downstream pressure is provided as an input to the model to represent boiler load demand. However downstream pressure does not depend completely on load demand. It also depends on the amount of steam generated in the boilers. If more steam is generated in a boiler, downstream pressure will also increase, provided that load demand is constant. If the steam generation rate is decreased, downstream pressure will also decrease, provided that load demand is constant. In other words, downstream pressure is an output as well as an input.

It is however reasonable to neglect the effect of changes in steam generation rate in any boiler on downstream pressure. Firstly, changes in load demand have a much more dramatic effect on downstream pressure than changes in fuel flow or feedwater rate. Secondly, as eight or more boilers feed into a common header, as shown in 4.10 the effect of changes in pressure in any one of those boilers on downstream pressure is not very significant.

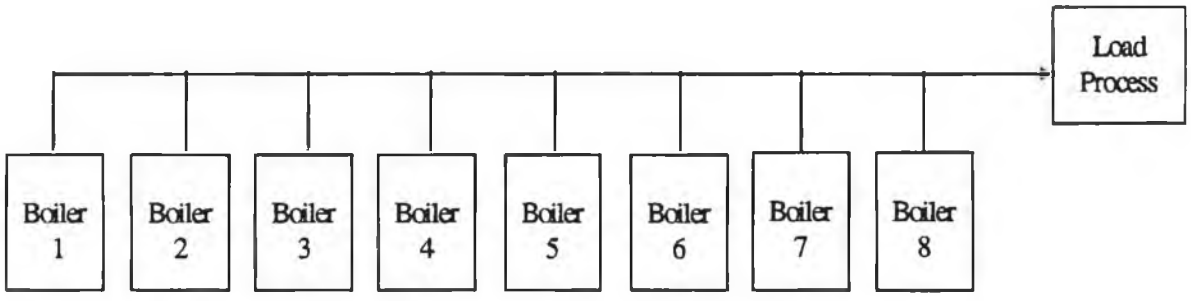


Fig. 4.10 Plant Configuration

A consequence of this assumption, is that load pressure variations are transmitted faithfully along the header back to the boiler outlet. In other word, the pressure at the boiler outlet (superheater pressure) can be assumed equal to the pressure in the header delayed by a certain time. Likewise, the pressure in the header is equal to the pressure in the superheater advanced by a certain time. This was in fact how downstream pressure was calculated as no downstream pressure measurement was available. The length of delay was taken to be about 10 seconds, but this value does not have an important effect on model operation.

Ideally, the model validation set should include a measurement signal corresponding to each of the model state variables and outputs. A direct comparison could then be made between each of the model variables and the corresponding plant measurement. This would require a very large number of transducers which are usually available on very large boilers only. In addition, it may not be possible to directly measure some of the model state variables or outputs, such as steam quality or evaporation rate. The following measurement signals were logged from the boiler used for validation:

Measured Boiler Outputs

1. Economiser Outlet Temperature
2. Drum Pressure
3. Drum Level
4. Steam Flow
5. Superheater Outlet Steam Pressure
6. Superheater Outlet Steam Temperature

The location of each transducer is shown in Fig. 4.11.

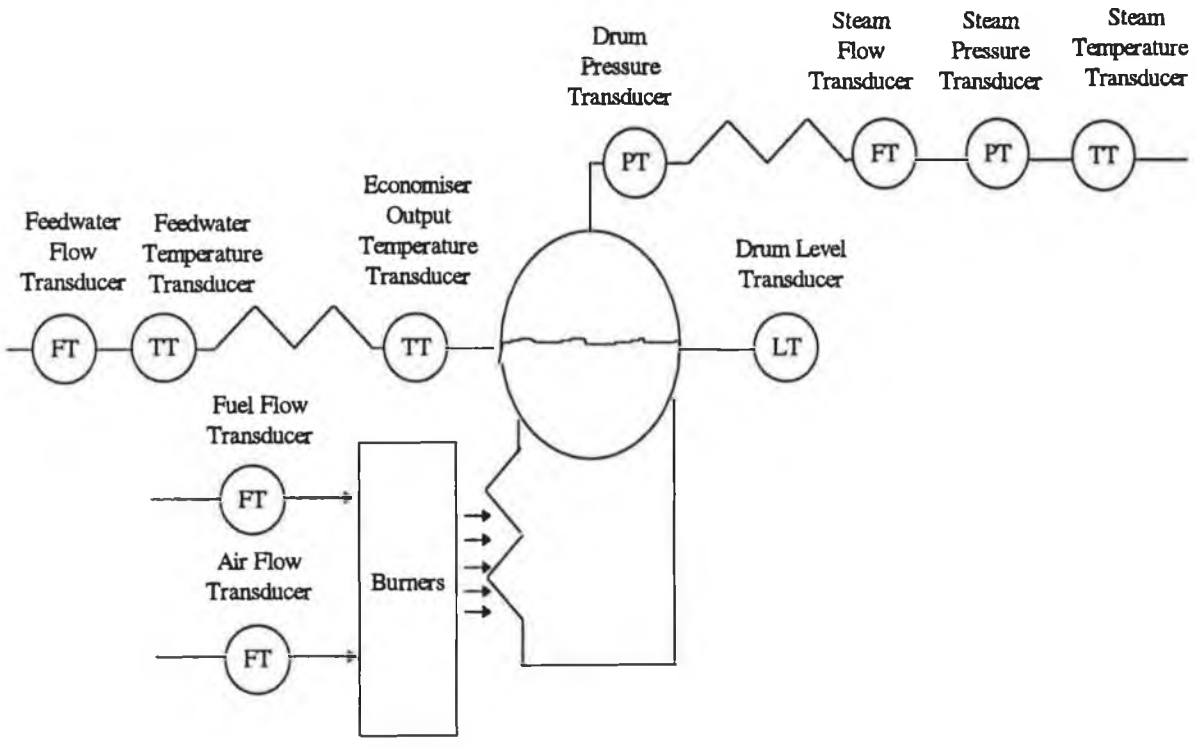


Fig. 4.11 Location of Transducers on each Boiler

4.5.2 Initialisation of State Variables and System Parameters

Correlation between model and boiler outputs does not just depend on the model structure. It also depends on the value of constants used in the model - system parameters and initial conditions. The correlation between boiler and model outputs can be improved by changing these constants. As a result, model validation consist of an iterative process of simulating the model, comparing simulation results with plant measurements and changing the model system parameters and initial conditions until it is felt that no further improvement in model performance can be achieved.

A variety of methods, described below, were used to calculate initial estimates of model constants.

4.5.2.1 Combustion Process

System Parameter		Method of Initialisation	Initial Values
Measured Value of $\%O_2$	$\%O_2$	Calculated using initial fuel and air mass flow rates	3%

Table 4.23 Initialisation of State Variables of Combustion Process Model

System Parameter		Method of Initialisation	Initial Values
Calorific Value of Fuel	C_{fu}	Fuel suppliers data sheet	33800 kJ kg^{-1}
Time Lag in Combustion Chamber	τ_{gc}	Assumed	3 s
Length of path from burners to O_2 sensor	L_{furn}	boiler design data	20 m
Cross-sectional area of furnace	A_{furn}	boiler design data	1 m^2
Sensor Measurement Delay	T_{sensor}	sensor manufacturer data sheet	5 s

Table 4.24 Initialisation of System Parameters of Combustion Process Model

4.5.2.2 Economiser

State Variables		Method of Initialisation	Initial Values
Economiser Gas Path Temperature	T_{eg_o}	Boiler design data	200 °C
Economiser Metal Temperature	T_{em}	$T_{em} = \frac{T_{eg_o} + T_{e_o}}{2}$ (4.159)	184 °C
Economiser Fluid Temperature	T_{e_o}	Measurement data	186 °C

Table. 4.25 Initialisation of State Variables of Economiser Model

System Parameters		Method of Initialisation	Initial Values
Economiser Gas Path Volume	V_{eg}	Boiler design data	12.31m^3
Economiser Metal Volume	V_{em}	Not available from boiler design data - assumed equal to riser metal volume	4.05 m^3
Economiser Volume	V_e	trial and error during validation	0.8 m^3
Economiser Gas-Metal Heat Transfer Coefficient	k_{egm}	trial and error during validation	6.57 kJ kg$^{-0.6}$ s$^{-0.4}$ °C^{-1}
Economiser Metal-Fluid Heat Transfer Coefficient	k_{emf}	trial and error during validation	4.86 kJ kg$^{-0.8}$ s$^{-0.2}$ °C^{-1}

Table. 4.26 Initialisation of System Parameters of Economiser Model

4.5.2.3 Drum, Downcomers and Risers

State Variables		Method of Initialisation	Initial Values
Riser Gas Path Temp. Pass 1	T_{rg1_o}	Boiler design data	1200 °C
Riser Metal Temp. Pass 1	T_{rm1}	$T_{rm1} = \frac{T_{rg1_o} + T_{r_o}}{2}$ (4.160)	729 °C
Riser Gas Path Temp., Pass 2	T_{rg2_o}	Boiler design data	400 °C
Riser Metal Temp., Pass 2	T_{rm2}	$T_{rm2} = \frac{T_{rg2_o} + T_{r_o}}{2}$ (4.161)	329 °C
Drum Pressure	P_d	Measurement data	4.596e6 Pa
Drum Water Volume	V_d	The measured value of drum level is used to calculate the drum water volume. It is assumed that the drum is cylindrical and that the drum level is equal to zero when the drum is exactly half full of water. If the drum water level is zero, the volume of water in the drum is equal to half of the total drum volume i.e. $V_{dw_0} = \frac{1}{2}\pi\left(\frac{D_d}{2}\right)^2 L_d$ (4.162) If the drum water level is above or below zero, the volume of water added or subtracted can be calculated by assuming that the extra volume of water has the shape of a rhombohedron. This assumption neglects the curve of the drum walls. The total drum water volume is then equal to: $V_{dw} = \frac{1}{2}\pi\left(\frac{D_d}{2}\right)^2 L_d + dl \times D_d$ (4.163)	5 m^3
Drum Steam Temperature	T_{do}	Assumed to be equal to the saturation temperature corresponding to drum pressure.	259 °C
Drum Water Temperature	T_{dw}	Assumed to be 5°C below the saturation temperature corresponding to drum pressure.	254 °C
Downcomer Water Temperature	T_{do_o}	Assumed to be equal to the drum water temperature.	254 °C

(Table 4.27 continued...)

State Variables		Method of Initialisation	Initial Values
Riser Fluid Temperature	T_{r_o}	Assumed to be equal to the saturation temperature corresponding to drum pressure.	259 °C
Riser Steam Quality	x	The riser steam quality is equal to the mass flow rate of steam from the drum divided by the mass recirculation rate of fluid through the risers. This ensure that the initial evaporation mass flow rate is equal to the initial mass flow rate from the drum $x = \frac{m_{s_i}}{m_r}$ (4.164)	0.0278

Table 4.28 Initialisation of State Variables of Drum, Downcomers and Risers Model

System Parameters		Method of Initialisation	Initial Values
Downcomer Volume	V_r	boiler design data	2.91 m^3
Drum Diameter Drum Length Drum Volume	D_d L_d V_d	boiler design data	1.2 m 8.5 m 9.61 m^3
Drum-Superheater Frictional Loss	H_LOSS_D	trial and error during validation	774.54
Number of Superheater Tubes	N_s	boiler design data	56
Recirculation Mass Flow Rate	m_r	trial and error during validation	50 kg/s
Riser Gas Path Volume, Pass 1	V_{rg1}	boiler design data	14.48 m^3
Riser Gas Path Volume, Pass 2	V_{rg2}	boiler design data	14.48 m^3
Riser Metal Volume, Pass 1	V_{rm1}	boiler design data	0.0058 m^3
Riser Metal Volume, Pass 2	V_{rm2}	boiler design data	0.0058 m^3
Riser Volume	V_r	boiler design data	4.04 m^3
Riser Gas-Metal Heat Transfer Rate Coefficient, Pass 1	k_{rgm1}	trial and error during validation	1.32e-9 kJ s^{-1} $°C^{-4}$
Riser Gas-Metal Heat Transfer Coefficient, Pass 2	k_{rgm2}	trial and error during validation	25.30 kJ $kg^{-0.6}$ $s^{-0.4}$ $°C^{-1}$
Riser Metal-Fluid Heat Transfer Rate, Coefficient, Pass 1	k_{rmf1}	trial and error during validation	4.67e-5 kJ s^{-1} $°C^{-3}$
Riser Metal-Fluid Heat Transfer Coefficient, Pass 2	k_{rmf2}	trial and error during validation	2.51 kJ $kg^{-0.8}$ $s^{-0.2}$ $°C^{-1}$

Table 4.29 Initialisation of System Parameters of Drum, Downcomers and Risers Model

4.5.2.4 Superheater

State Variables		Method of Initialisation	Initial Values
Superheater Gas Path Temperature	T_{sg_o}	Boiler design data	800 °C
Superheater Metal Temperature	T_{sm}	$T_{sm} = \frac{T_{sg_o} + T_{s_o}}{2}$ (4.165)	599 °C
Superheater Steam Pressure	P_{s_o}	Measurement data	4.6014e6 Pa
Superheater Steam Temperature	T_{s_o}	Measurement data	398 °C

Table 4.30 Initialisation of State Variables of Superheater Model

System Parameters		Method of Initialisation	Initial Values
Area of a Superheater Tube	D_s	boiler design data	7.94e-4 m^2
Number of Superheater Tubes	N_s	boiler design data	56
Superheater Exit Frictional Loss	H_LOSS_S	trial and error during validation	310
Superheater Gas Path Volume	V_{sg}	boiler design data	3.43 m^3
Superheater Metal Volume	V_{sm}	boiler design data	0.058 m^3
Superheater Volume	V_s	boiler design data	4.61 m^3
Superheater Gas-Metal Heat Transfer Coefficient	k_{sgm}	trial and error during validation	2.40 kJ $kg^{-0.6}$ $s^{-0.4}$ $°C^{-1}$
Superheater Metal-Fluid Heat Transfer Coefficient	k_{smf}	trial and error during validation	2.18 kJ $kg^{-0.8}$ $s^{-0.2}$ $°C^{-1}$

Table 4.31 Initialisation of System Parameters of Superheater Model

4.5.2.5 Downstream Process

State Variables		Method of Initialisation	Initial Values
Downstream Process Pressure	P_{l_o}	Calculated using steady state values of superheater steam pressure and main steam flow rate	
Downstream Process Temperature	T_{l_o}	Assumed equal to superheater steam temperature	

Table 4.32 Initialisation of State Variables of Downstream Process Model

System Parameters		Method of Initialisation	Initial Values
Low Pressure behind valves	P_{low}	Arbitrary load - value not important	$8e^{-5}$ Pa
Valve Mass Flow Rate Coefficient	k_v	Arbitrary load - value not important	$8.5e^{-6}$
Volume of Downstream Process	V_l	Arbitrary load - value not important	-

Table 4.33 Initialisation of System Parameters of Downstream Process Model

4.5.3 Validation Results

The model is simulated for 14000s (nearly four hours) using the measured boiler input signals as the model inputs. A plot of boiler measurements versus model outputs is given below for each of the available boiler measurands.

1. Economiser Outlet Temperature

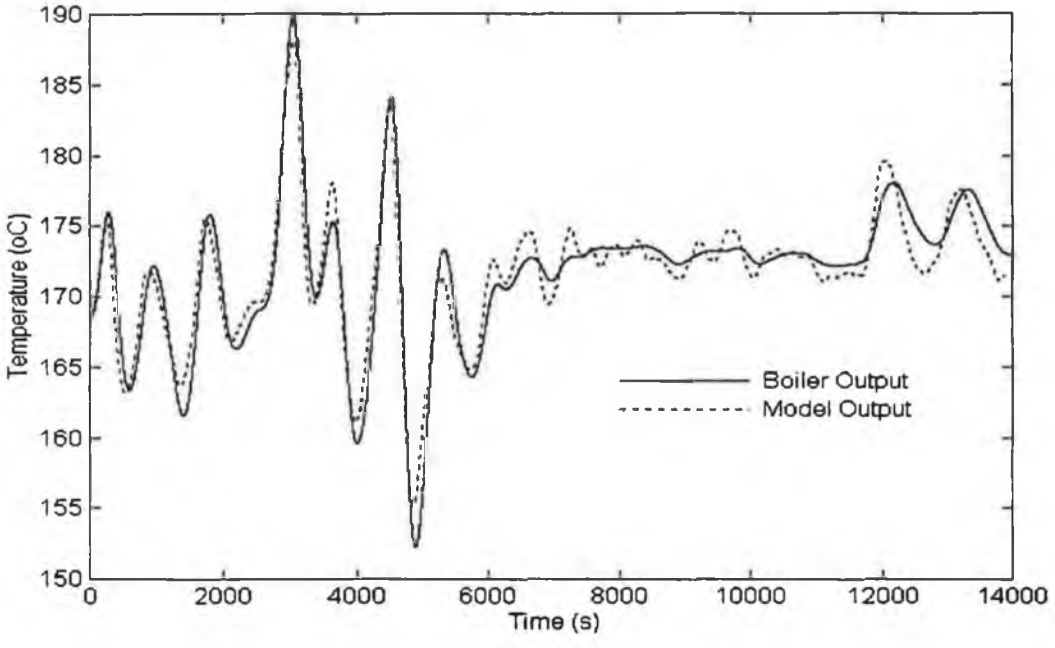


Fig. 4.12 Measured *versus* Modelled Economiser Water Temperature

The calculated value of economiser temperature is in close agreement with the measured value, especially over the first two hours of the logged data. Between 6000 and 1200 seconds, the measured value economiser temperature remains practically constant. The model output is changing slightly over this period as it responds to rapid and noisy variations in the measured value of feedwater flow. It is likely that the variations in feedwater flow are in fact a random measurement error. This would explain why they have no effect on the actual boiler outputs.

2. Drum Pressure

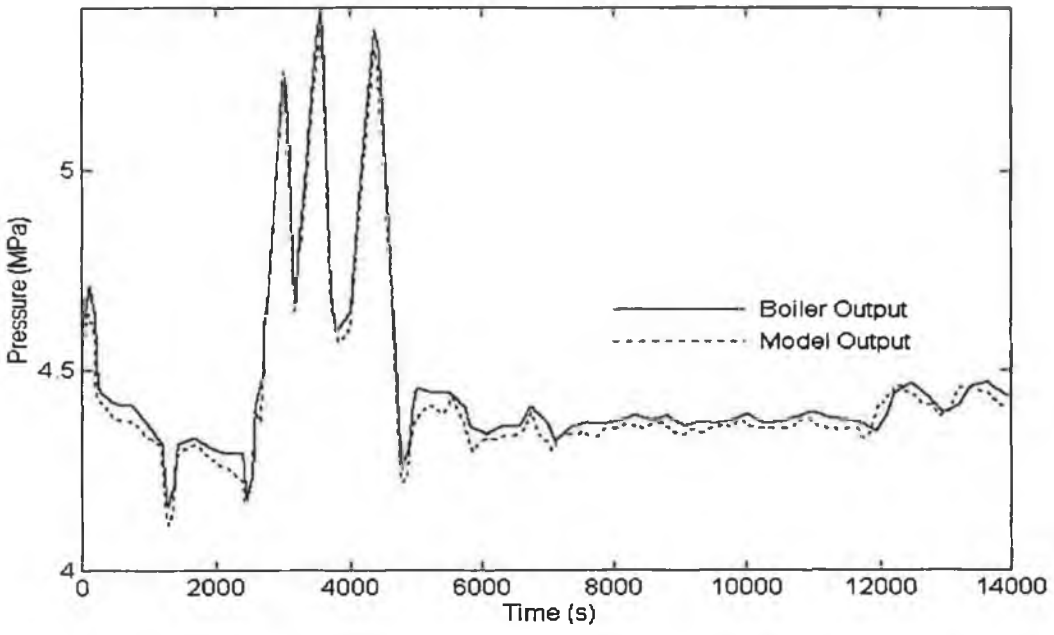


Fig. 4.13 Measured *versus* Modelled Drum Pressure

There is a very close agreement between the actual boiler output and the model output over the complete time span.

3. Drum Level

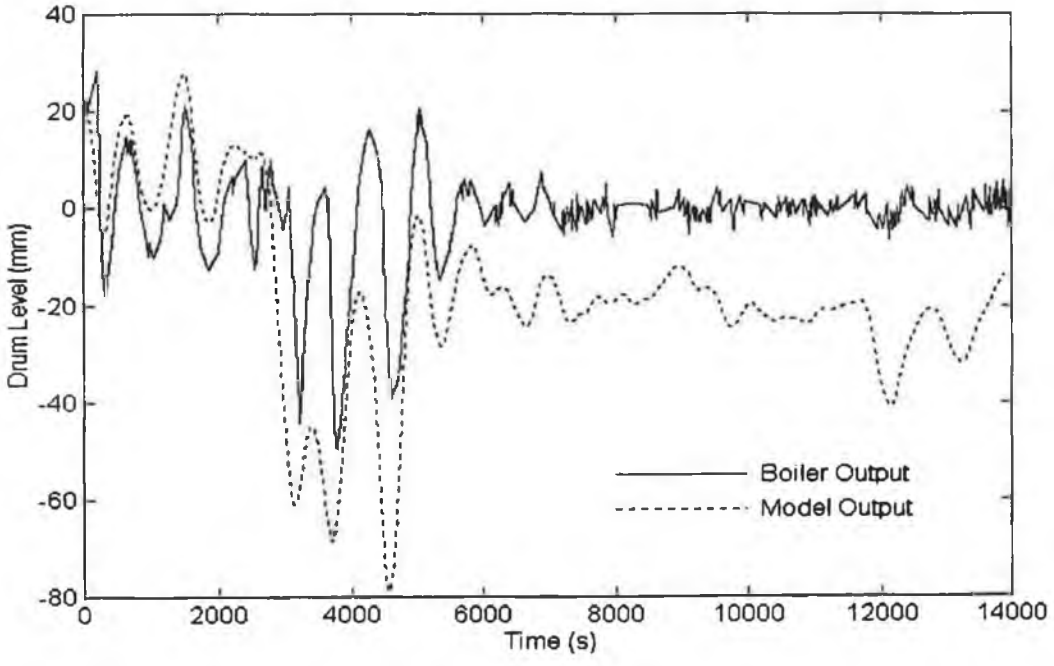


Fig. 4.14 Measured *versus* Modelled Drum Level

It is clear that the simulated level signal follows the trend of the measured drum level signal. There is however, a noticeable d.c. difference between the simulated drum level signal and the measured drum level signal, which increases over the time period of the simulation. This is because the boiler drum is an integrator and errors in drum level accumulate over the simulation time.

4. Steam Flow

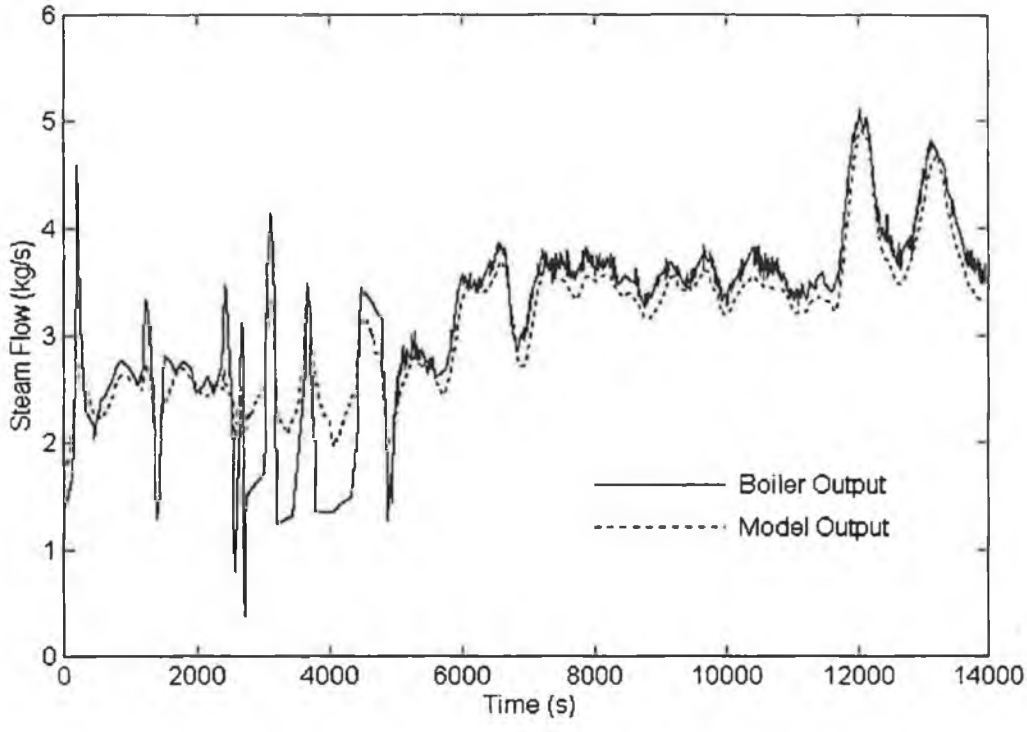


Fig. 4.15 Measured *versus* Modelled Steam Flow

There is good agreement between the simulated value and measured value of steam flow. Discrepancies are most noticeable whenever there are large and fast variations in the measured steam flow. It was not possible to model these large, fast variations for the following reason. Steam flow is calculated as a function of the difference between drum and superheater pressure. However, if measured drum pressure and superheater pressure are used to calculate steam flow, the resulting steam flow sequence does not include these large and fast variations. This may be due to insufficient or irregular pressure measurements. As downstream pressure is used a measure of load, the modelled steam flow signal is also subject to this problem and does not include the large, fast variations in pressure.

5. Superheater Outlet Steam Pressure

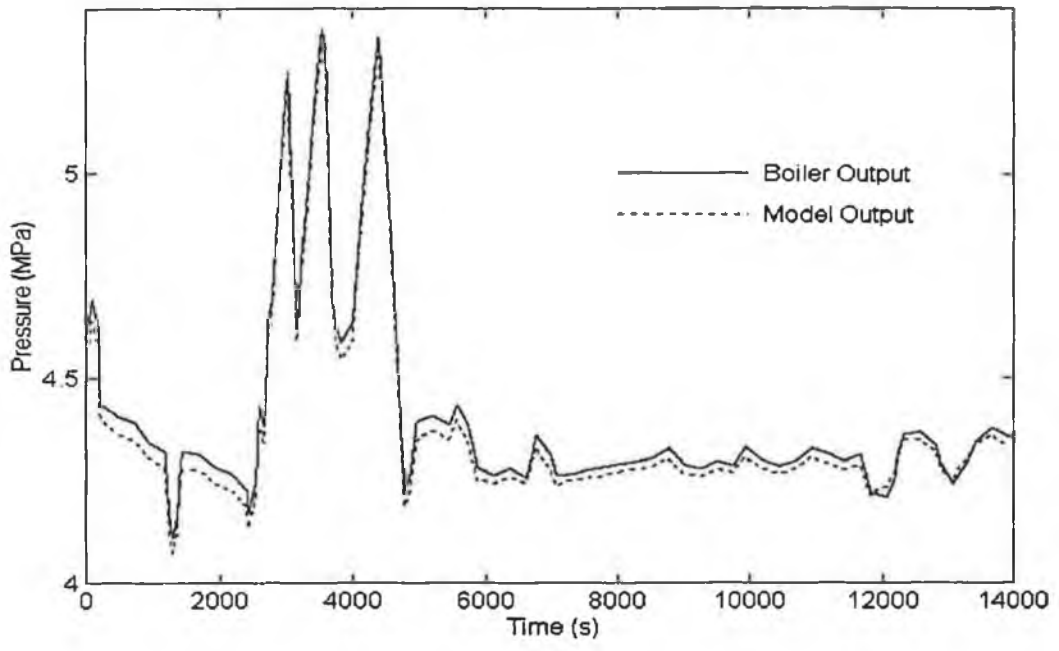


Fig. 4.16 Measured *versus* Modelled Superheater Steam Pressure

There is excellent agreement between the measured and calculated values of superheater steam pressure.

6. Superheater Outlet Steam Temperature

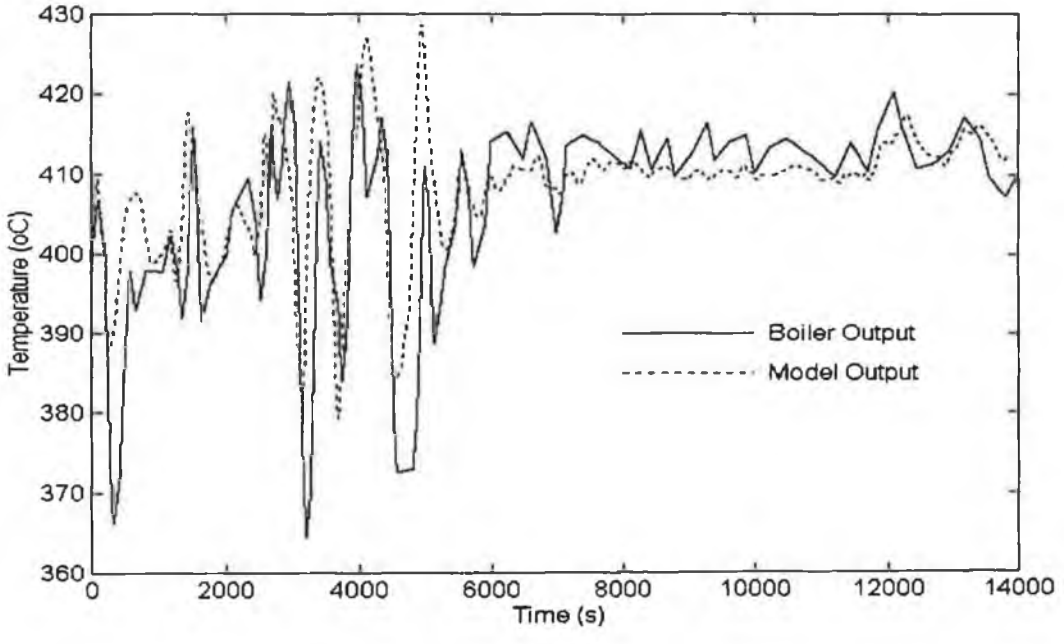


Fig. 4.17 Measured *versus* Modelled Superheater Steam Temperature

The simulated value of superheater steam temperature follows the trends in the measured temperature very well. Discrepancies may be due to superheater attemperation, which has been assumed equal to zero as the degree of attemperation is not measured on the boiler.

4.6 Conclusions

The development, implementation and validation of a first principles model has been described in this chapter. This was a lengthy process and the benefits of developing a first principles model must be carefully considered. A good first principles model:

- Is suitable for use as a plant simulator to test controller designs
- Can be customised using plant data to represent any drum-type boiler
- Provides insight into boiler processes

These benefits are contingent upon the model representing the plant well. This is ensured by the detail and accuracy of the implemented model. Model accuracy has been obtained in a number of ways:

1. The process is broken into as many lumped systems as is feasible. For example, the economiser is modelled by three separate lumped systems - a gas lumped system, a metal lumped system and a fluid lumped system.
2. The minimum number of assumptions have been adopted. For example, steam in the drum is not assumed to be saturated.
3. Steam and water tables are modelled very accurately (typical error is less than 0.1%) by using a large number of polynomial functions.
4. The common assumption that the fluid in the drum is always in steady-state saturated equilibrium has been modified. For this model, it is assumed that following a change in drum inputs, that saturated equilibrium conditions are established over a given time span. This eliminates some serious modelling anomalies which arise as a result of assuming continuous saturated equilibrium.

The model has been carefully validated against actual plant data. Model validation confirms that there is good agreement between the model output and actual plant data.

5. Development of Linearised Boiler Models

5.1 Introduction

Predictive control is a model based strategy. In theory, the model may be of any type, the choice being dictated by the characteristics of the system to be controlled and by the controller performance specifications. For example, a simple linear system may be modelled by a simple linear model. If the system is nonlinear, there are two alternatives: a nonlinear model or a set of linearised models. A single nonlinear model is capable of representing the system at all operating points. A single linearised model can represent a nonlinear system at one operating point only. In order to model the plant behaviour over the complete operating range, it is necessary to generate several linear models at different operating points. However, if hard nonlinearities such as hard constraints exist, these can only be modelled by a nonlinear model.

The choice of model is also influenced by practical considerations such as the time available for computation, however. Linear models are generally chosen for both practical and historical reasons. Their limitations with respect to modelling system nonlinearities and constraints are offset by some important advantages. Linear controllers are well understood theoretically. An unconstrained predictive controller based upon a linear model has an analytical control solution which uses predetermined controller coefficients and can be easily computed. There are many modelling packages available which can generate linear models from plant data. In addition to generating a plant model, system identification using linear modelling techniques can also provide a good deal of information about the plant such as the order of the plant or the length (if any) of plant time delay.

5.2 Structure of Linearised Models

The internal controller model is required to predict the following variables:

1. Main steam pressure
2. Main steam temperature
3. Drum level
4. Percentage O_2 in stack gases (not including the sensor measurement delay)

A model-based predictive control strategy requires that the future output of each of these variables can be predicted for a given set of inputs. A single model could be used to predict the future output of all four variables. However, for practical reasons, it was decided to use three different models:

- There are strong interactions between main steam pressure and drum level. As a result, both variables are controlled by a single multivariable controller and likewise modelled by a single multivariable model.

- Main steam temperature has much faster dynamics than either main steam pressure or drum level and must be controlled by a controller with a shorter sampling period. It is not necessary to include a model of the complete boiler for main steam temperature control. It is sufficient to employ a model of the superheater only, with measured drum pressure, drum steam temperature and the combustion gas temperature as feedforward variables. This model has fewer states than the model of the complete boiler and results in a controller of reduced computational complexity.

- There is little interaction between the percentage of O_2 in the stack gases and the three fluid-side variables (pressure, temperature and drum level). As a result it is controlled by a separate controller which uses an internal model of the combustion process only. Again, this model has fewer states than the model of the complete boiler process and so reduces the controller computional overhead.

In summary, the following three linearised models are required:

1. Multivariable model of main steam pressure and drum level
2. Model of main steam temperature
3. Model of percentage O_2 in stack gases, (not including the sensor measurement delay)

The inputs, state variables and output variables for each of these models are given in Table 5.1, 5.1 and 5.3

Model of Main Steam Pressure and Drum Level		
Model Inputs	**Model State Variables**	**Model Outputs**
1. Feedwater Mass Flow Rate 2. Fuel Mass Flow Rate 3. Attemperating Mass Flow Rate 4. Air Mass Flow Rate 5. Feedwater Temperature 6. Number of Valves	1. Economiser Gas Temperature 2. Economiser Metal Temperature 3. Economiser Water Temperature 4. Riser Gas Temperature, Pass 1 5. Riser Metal Temperature, Pass 1 6. Riser Gas Temperature, Pass 2 7. Riser Metal Temperature, Pass 2 8. Drum Steam Temperature 9. Drum Water Temperature 10. Downcomer Water Temperature 11. Riser Fluid Temperature 12. Riser Steam Quality 13. Drum Pressure 14. Drum Water Volume 15. Superheater Gas Temperature 16. Superheater Metal Temperature 17. Superheater Steam Pressure 18. Superheater Steam Temperature 19. Main Steam Temperature 20. Main Steam Pressure	1. Main Steam Pressure 2. Drum Level

Table 5.1 Inputs, State Variables and Outputs of Main Steam Pressure and Drum Level Model

Model of Main Steam Temperature		
Model Inputs	**Model State Variables**	**Model Outputs**
1. Fuel Mass Flow Rate 2. Attemperating Mass Flow Rate 3. Air Mass Flow Rate 4. Estimated Valve Position 5. Measured Drum Pressure 6. Measured Steam Pressure 7. Measured Combustion Gas Temperature	1. Superheater Gas Temperature 2. Superheater Metal Temperature 3. Superheater Steam Pressure 4. Superheater Steam Temperature 5. Main Steam Temperature 6. Main Steam Pressure	1. Main Steam Temperature

Table 5.2 Inputs, State Variables and Outputs of Main Steam Temperature Model

Model of Percentage O_2 in Stack Gases		
Model Inputs	**Model State Variables**	**Model Outputs**
1. Fuel Mass Flow Rate 2. Air Mass Flow Rate	1. Percentage O_2 in stack gases (undelayed)	1. Percentage O_2 in stack gases (undelayed)

Table 5.3 Inputs, State Variables and Outputs of Percentage O_2 in Stack Gases Model

5.3 Linearisation

A linearised model can be obtained either by system identification using plant data or by linearisation of the nonlinear boiler model. Linearisation of the nonlinear model was chosen as a good nonlinear model was available which has been shown to represent the plant well. Linearisation of the nonlinear model generates a state space model which is computationally efficient and can be used by multivariable controllers. System identification for a process of this complexity would require a large amount of open loop plant data, which was not available.

A linearised model of a nonlinear plant is valid for small deviations about a stable equilibrium point of the nonlinear plant. It is assumed that the system is in equilibrium under the conditions, (x_o, u_o), i.e.

$$\dot{x} = f(x_o, u_o) = 0$$

The system is linearised about the equilibrium point (x_o, u_o) by expanding it into a Taylor series and neglecting second-order terms and higher. These higher order terms can be neglected on the basis that the deviation about the equilibrium point, $(x_j - x_{jo})$, is very small. For the i^{th} state equation, the expansion is:

$$\dot{x}_i = f_i(x_o, u_o) + \sum_{j=1}^{n} \left.\frac{\partial f_i(x,u)}{\partial x_j}\right|_{\substack{x=x_o \\ u=u_o}} (x_j - x_{jo}) + \sum_{k=1}^{mn} \left.\frac{\partial f_i(x,u)}{\partial u_k}\right|_{\substack{x=x_o \\ u=u_o}} (u_k - u_{ko}) \qquad (5.1)$$

The variation about the operating point may be defined as follows:

$$\begin{aligned} \tilde{x}_j &= x_j - x_{j0} \;\; \therefore \;\; \dot{\tilde{x}}_j = \dot{x}_j \\ \tilde{u}_k &= u_k - u_{k0} \end{aligned} \qquad (5.2)$$

Given that $f(x_o, u_o) = 0$, the Taylor series for the i^{th} state equation can be rewritten as:

$$\dot{x}_i = \sum_{j=1}^{n} \left.\frac{\partial f_i(x,u)}{\partial x_j}\right|_{\substack{x=x_o \\ u=u_o}} \tilde{x}_j + \sum_{k=1}^{mn} \left.\frac{\partial f_i(x,u)}{\partial u_k}\right|_{\substack{x=x_o \\ u=u_o}} \tilde{u}_k \qquad (5.3)$$

The complete state of linearised state equations can now be written as:

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u}$$

where

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \cdots & \frac{\partial f_2}{\partial u_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \cdots & \frac{\partial f_n}{\partial u_n} \end{bmatrix} \qquad (5.4)$$

It is not feasible to obtain analytical expressions for the partial derivatives in this case, due to the complexity of the nonlinear model. Instead, the partial derivatives are calculated numerically. The nonlinear function is first calculated at equilibrium conditions. The function is then recalculated for a variation in each state variable. Only one state variable is perturbed at a time. For a perturbation in state variable, x_j, only, the change in the i^{th} state equation is:

$$\begin{aligned} \Delta f_{i_{xj}} &= f_i(x,u)\Big|_{u=u_{ko}}^{x=x_j} - f_i(x,u)\Big|_{u=u_{ko}}^{x=x_{jo}} \\ &= f_i(x,u)\Big|_{u=u_{ko}}^{x=x_j} \end{aligned} \qquad (5.5)$$

Similarly, for a perturbation in the input variable, u_k, only, the change in the i^{th} state equation is:

$$\begin{aligned} \Delta f_{i_{uk}} &= f_i(x,u)\Big|_{u=u_k}^{x=x_{j0}} - f_i(x,u)\Big|_{u=u_{ko}}^{x=x_{jo}} \\ &= f_i(x,u)\Big|_{u=u_k}^{x=x_{j0}} \end{aligned} \qquad (5.6)$$

The resulting linearised state equations are:

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u}$$

where

$$A = \begin{bmatrix} \frac{\Delta f_{1_{x1}}}{\tilde{x}_1} & \frac{\Delta f_{1_{x2}}}{\tilde{x}_2} & \cdots & \frac{\Delta f_{1_{xn}}}{\tilde{x}_n} \\ \frac{\Delta f_{2_{x1}}}{\tilde{x}_1} & \frac{\Delta f_{2_{x2}}}{\tilde{x}_2} & \cdots & \frac{\Delta f_{2_{xn}}}{\tilde{x}_n} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ \frac{\Delta f_{n_{x1}}}{\tilde{x}_1} & \frac{\Delta f_{nx2}}{\tilde{x}_2} & \cdots & \frac{\Delta f_{n_{xn}}}{\tilde{x}_n} \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{\Delta f_{1_{u1}}}{\tilde{u}_1} & \frac{\Delta f_{1_{u2}}}{\tilde{u}_2} & \cdots & \frac{\Delta f_{1_{um}}}{\tilde{u}_m} \\ \frac{\Delta f_{2_{u1}}}{\tilde{u}_1} & \frac{\Delta f_{2_{u2}}}{\tilde{u}_2} & \cdots & \frac{\Delta f_{2_{um}}}{\tilde{u}_m} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ \frac{\Delta f_{n_{u1}}}{\tilde{u}_1} & \frac{\Delta f_{n_{u2}}}{\tilde{u}_2} & \cdots & \frac{\Delta f_{n_{um}}}{\tilde{u}_m} \end{bmatrix} \qquad (5.7)$$

The same approach is used to obtain the system output equations. The system outputs are a function of the state variables and model inputs. For the p^{th} output equation, the relationship is:

$$y_p = g_p(x,u) \qquad (5.8)$$

The variation in each output variable is calculated for a perturbation in each individual state variables from its equilibrium value. For a perturbation in state variable, x_j, only, the change in the p^{th} output equation is:

$$\tilde{y}_{p_{xj}} = g_p(x,u)\Big|_{u=u_{ko}}^{x=x_j} - g_p(x,u)\Big|_{u=u_{ko}}^{x=x_{jo}} \qquad (5.9)$$

For a perturbation in model input, u_k, only, the change in the p^{th} output equation is:

$$\tilde{y}_{p_{uk}} = g_p(x,u)\Big|_{u=u_k}^{x=x_{j0}} - g_p(x,u)\Big|_{u=u_{ko}}^{x=x_{jo}} \qquad (5.10)$$

The resulting output equations are:

$$\tilde{y} = C\tilde{x} + D\tilde{u}$$

where

$$C = \begin{bmatrix} \frac{\tilde{y}_{1_{x1}}}{\tilde{x}_1} & \frac{\tilde{y}_{1_{x2}}}{\tilde{x}_2} & \cdots & \frac{\tilde{y}_{1_{xn}}}{\tilde{x}_n} \\ \frac{\tilde{y}_{2_{x1}}}{\tilde{x}_1} & \frac{\tilde{y}_{2_{x2}}}{\tilde{x}_2} & \cdots & \frac{\tilde{y}_{2_{xn}}}{\tilde{x}_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\tilde{y}_{l_{x1}}}{\tilde{x}_1} & \frac{\tilde{y}_{l_{x2}}}{\tilde{x}_2} & \cdots & \frac{\tilde{y}_{l_{xn}}}{\tilde{x}_n} \end{bmatrix}$$

$$D = \begin{bmatrix} \frac{\tilde{y}_{1_{u1}}}{\tilde{u}_1} & \frac{\tilde{y}_{1_{u2}}}{\tilde{u}_2} & \cdots & \frac{\tilde{y}_{1_{um}}}{\tilde{u}_m} \\ \frac{\tilde{y}_{2_{u1}}}{\tilde{u}_1} & \frac{\tilde{y}_{2_{u2}}}{\tilde{u}_2} & \cdots & \frac{\tilde{y}_{2_{um}}}{\tilde{u}_m} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\tilde{y}_{l_{u1}}}{\tilde{u}_1} & \frac{\tilde{y}_{l_{u2}}}{\tilde{u}_2} & \cdots & \frac{\tilde{y}_{l_{um}}}{\tilde{u}_m} \end{bmatrix} \qquad (5.11)$$

Best results were obtained for very small perturbations. The state vector and input vector perturbation was set equal to a factor of $1e^{-8}$ of the steady-state state vector and input vector values respectively.

5.4 Modification to Drum Level Model

A linear model can represent a nonlinear plant extremely well when the operating point is close to the equilibrium point of the linear model. However, in normal operation, the linear model must represent the plant over quite a large operating range. As the operating point moves away from the equilibrium point, the discrepancy between the plant and the linear model becomes more significant. An examination of the seven slowest eigenvalues of the reduced linearised model provides evidence that modelling errors may have serious consequences:

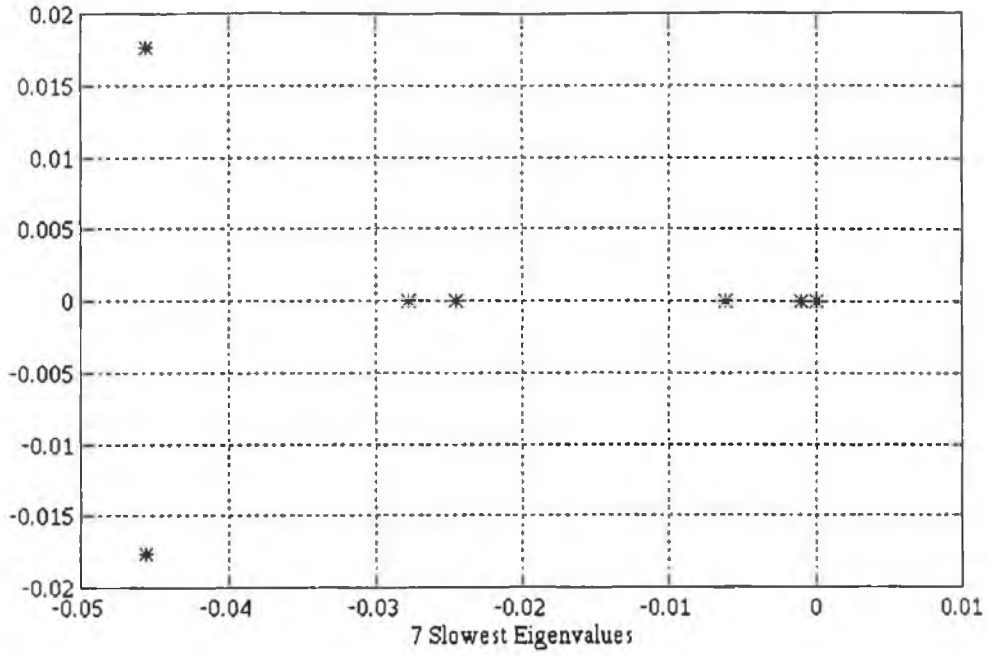


Fig. 5.1 7 Slowest Eigenvalues of Linearised Model at 50% Load

Fig. 5.1 reveals an eigenvalue lying at the origin. This eigenvalue is associated with the mass of water in the drum, downcomers and risers, which is a pure integrator. The effect of a pure integrator on plant response can be demonstrated by investigating a SISO discretised plant, modelled as:

$$\begin{aligned} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) + Du(n) \end{aligned} \tag{5.12}$$

If it is assumed that the plant input, $u(n)$, is constant, the predicted output at time k is:

$$\begin{aligned} x(n+k+1) &= A^k x(n) + B\sum_{i=1}^{k} A^{i-1}u(n) \\ y(n+k) &= CA^k x(n) + CB\sum_{i=1}^{k} A^{i-1}u(n) + Du(n) \end{aligned} \tag{5.13}$$

If the plant is open-loop stable, then $|A| < 1$. In this case the output of the plant after an infinite time is:

$$\begin{aligned} \lim_{k\to\infty} y(n+k) &= \lim_{k\to\infty} [CA^k x(n) + CB\sum_{i=1}^{k} A^{i-1}u(n) + Du(n)] \\ &= \text{constant} \end{aligned} \tag{5.14}$$

However, if the plant is an integrator, then $|A| = 1$. In which case, the output of the plant after an infinite time is:

$$\begin{aligned} \lim_{k\to\infty} y(n+k) &= \lim_{k\to\infty} [CA^k x(n) + (k-1)CBu(n) + Du(n)] \\ &= \text{constant } if\ |u(n)| = 0 \\ \text{but,} \quad &= \infty\ if\ |u(n)| > 0 \end{aligned} \tag{5.15}$$

It follows that the mass of water in the drum, downcomers and risers will increase indefinitely, if the integrator input is not equal to zero. The integrator input in this instance is the difference between the mass flow rate of feedwater into the drum (a model input) and the mass evaporation rate of steam from the drum (a model output). The mass of water is modelled by:

$$M_w(n+1) = M_w(n) + m_{fw}(n) + m_{at}(n) - m_{evap}(n)$$

where

M_w = mass of water in drum, downcomers and riser
m_{fw} = mass flow rate of feedwater
m_{at} = mass flow rate of attemperating water
m_{evap} = mass evaporation rate of steam from drum (5.16)

The mass of water in the drum, downcomers and risers will remain constant, provided that the mass flow rate of feedwater into the drum is equal to the mass evaporation rate of steam from the drum. However, if the modelled mass evaporation rate is calculated incorrectly, and consequently is not equal to the actual feedwater flow rate, this will cause a continuously increasing error in the modelled drum level. One way to overcome this problem is to use the actual mass evaporation rate of steam from the drum instead of the modelled value as the integrator input. (In practice, the mass flow rate of steam from the boiler must be used in preference to the mass evaporation rate, as the latter is not measurable. At steady state, the mass flow rate of steam from the drum is equal to the mass evaporation rate.) The mass flow rate of steam from the boiler is a function of the load process pressure and the load demand, which is modelled by a bank of valves. The mass flow rate of steam is modelled by:

$$m_{l_o} = f(P_{l_o}, N_v)$$

where

m_{l_o} = modelled mass flow rate of steam generated by boiler (model output)
P_{l_o} = Pressure of load process (state variable)
N_v = Number of open valves at boiler output i.e. load demand (model input) (5.17)

After linearisation, this equation is:

$$m_{l_o} = a_1 P_{l_o} + a_2 N_v$$

where (5.18)

a_1, a_2 = coefficients of linearised equation

This linearised equation can be rearranged, to give an expression for load demand in terms of steam flow and load pressure:

$$N_v = \frac{m_{l_o} - a_{ij} P_{l_o}}{b_{ik}} \tag{5.19}$$

The modelled steam flow is now replaced with the measured steam flow, so that the "correct" load demand signal can be estimated i.e. the load demand signal which must be used to ensure that the modelled steam flow is exactly equal to the measured steam flow:

$$N_v(est) = \frac{m_{l_o}(meas) - a_{ij} P_{l_o}}{b_{ik}} \tag{5.20}$$

The mass of water in the drum, downcomers and risers is modelled using this estimated value of load demand.

$$M_w(n+1) = M_w(n) + m_{fw}(n) + m_{at}(n) - a_{ij} P_{l_o}(n) + b_{ik} N_v(est)(n)$$

where

M_w = Mass of water in drum, downcomers and risers (5.21)

m_{fw} = mass flow rate of feedwater into drum

m_{at} = mass flow rate of attemperating water in superheater

This ensures that the modelled and measured steam mass flow rate are equal.

A further modification must be made to linearised model before this strategy can be implemented. Drum level is an model output, calculated in terms of the state variables of the model - it is primarily a function of the volume of water in the drum, downcomers and risers, however it also depends on other pressure and temperature state variables to some extent. The original linearised

equation for drum level in terms of the model state variables is:

$$L_d = c_1 P_d + c_2 V_{dw} + c_3 T_{do_o} + c_4 T_{r_o} + c_5 x$$

where

L_d = drum level
P_d = drum pressure
V_{dw} = drum water volume
T_{do_o} = temperature of downcomer water
T_{r_o} = temperature of riser fluid
x = steam quality
$c_1 \ldots c_5$ = coefficients of linearised equation

(5.22)

It can be seen from equation (5.22) that the original linearised model uses drum water volume as a state variable instead of water mass. Drum water volume is not a pure integrator - it depends on the pressure in the drum and the temperature of the drum water as well as on drum water mass. The strategy requires that the mass of water in the drum, downcomers and risers is used to calculate drum level, since this is a pure integrator. Consequently, it is necessary to rewrite equation (5.22), replacing water mass with drum water volume. The water mass is related to the original state variables as follows:

$$M_{dw} = V_w \rho_{dw}(P_d, T_{dw}) + V_r \rho_{r_o}(P_d, x, T_{r_o}) + V_{do_o} \rho_{do_o}(P_d, T_{do_o})$$

where

ρ_{dw} = density of drum water
ρ_{r_o} = density of riser fluid
ρ_{do_o} = density of downcomer fluid
V_r = volume of risers
V_{do} = volume of downcomers
T_{dw} = temperature of drum water

(5.23)

This equation is linearised to provide a linear expression for the water mass in terms of the original state variables:

$$M_w = a_1 P_d + a_2 V_w + a_3 T_{do_o} + a_4 T_{r_o} + a_5 x + a_6 T_{dw}$$

where $a_1 \ldots a_6$ = coefficients of linearised equation

(5.24)

Equation (5.24) can be rearranged to give an expression for drum water volume in terms of the mass of water in the evaporation system and the original state variables.

$$V_w = (M_w - a_1 P_d - a_3 T_{do_o} - a_4 T_{r_o} - a_5 x - a_6 T_{dw}) / a_2 \qquad (5.25)$$

This equation for water volume is then substituted into the linearised equation for drum level - equation (5.22), yielding

$$L_d = (c_1 - \frac{a_1}{a_2})P_d + \frac{c_2}{a_2}M_w + (c_3 - \frac{a_3}{a_{21}})T_{do_o} + (c_4 - \frac{a_4}{a_2})T_{r_o} + (c_5 - \frac{a_5}{a_1})x - \frac{a_6}{a_2}T_{dw} \tag{5.26}$$

Likewise, a substitution must be made for drum water volume in each of the state variable equations. For an arbitrary state variable, x_j, the state variable equation can be written as:

$$x_j = A'x' + a_{dw}V_{dw}$$

where

A' = coefficients of all state variables, except V_{dw} (5.27)

x' = all state variables except V_{dw}

a_{dw} = coefficient of V_{dw}

Equation (5.25) can now be used to substitute for drum water volume, V_{dw}, yielding:

$$x_j = A'x' + a_{dw}(M_w - a_1P_d - a_3T_{do_o} - a_4T_{r_o} - a_5x - a_6T_{dw})/a_2 \tag{5.28}$$

This method further requires that the measured feedwater and steam flow are accurately calibrated. Any discrepancy between the actual and the measured steam flow will cause a continuous increase or decrease in the modelled drum level. The following strategy has been employed to cancel any drift in sensor calibration. The error between the measured and modelled drum level is integrated and the integrator output is then added to the measured steam flow to give the "corrected" steam flow. The integrator gain is very small so that short-term differences between the measured and modelled drum level do not significantly effect the integrator output. This strategy is shown in Fig. 5.2.

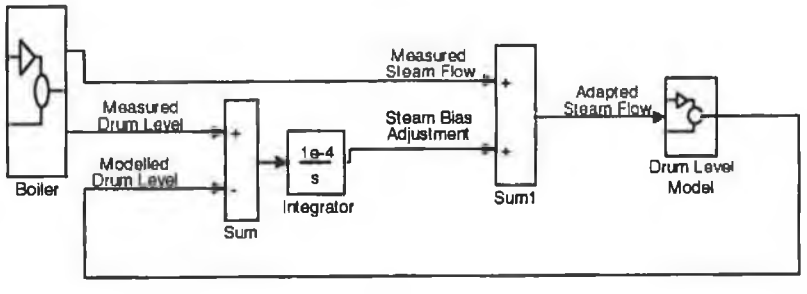


Fig. 5.2. Strategy for On-line Correction of Steam Flow Measurement

Figure 5.3 compares drum level for the plant, the original linear model and the modified linear model when the plant steps from the 50% steady state operating point to the 55% operating point.

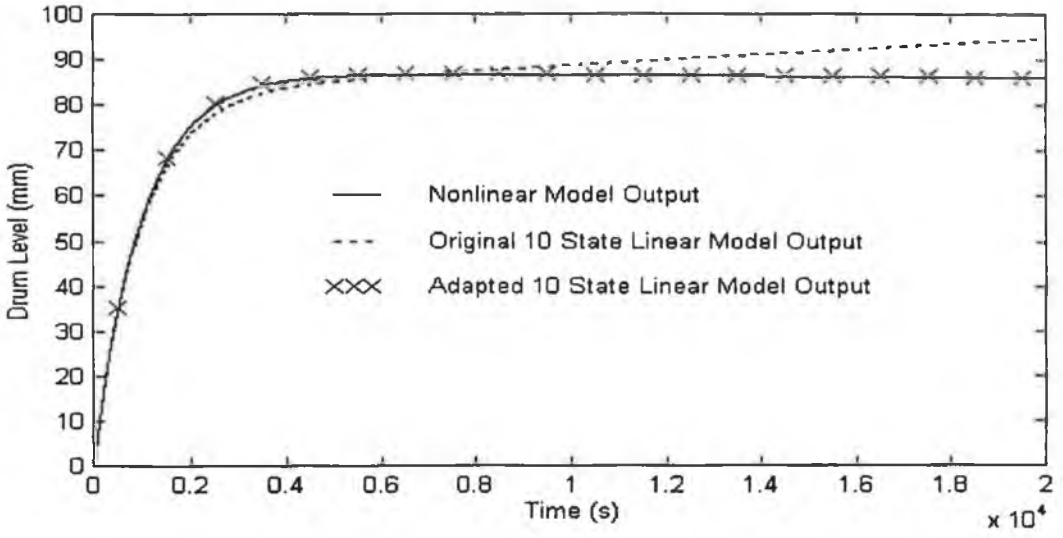


Fig. 5.3 Comparison of Drum Level for Plant, Original Linearised Model and Modified Linearised Model

Fig. 5.3 demonstrates that the original linearised model cannot correctly model steam flow over an extended period of time. The modified linearised model can however correctly model the nonlinear model output.

5.6 Results

Two important criteria determine the suitability of a model for use as an internal controller model - computational overhead and the ability of the model to adequately represent the plant.

Computational overhead is not a problem in this instance as the linearised models are compact and computationally efficient. These results focus on the ability of the linearised models to represent the boiler plant for both small and large disturbances from their equilibrium points.

The performance of the linearised models for large disturbances is significant, as it dictates how many linearised models are required to cover the specified operating range of the boiler plant.

5.6.1 Assessment of Linearised Model Validity for Small Disturbances

For comparison purposes, the nonlinear model and the linearised discrete models are excited by the same set of inputs - square waves of random amplitude and frequency. Each model input is allowed to vary by up to 5% of its maximum value around its steady state operating value at 50% load. Fig. 5.4 to 5.7 show the simulated value of main steam pressure, main steam temperature,

drum level and percentage O_2 as calculated by the nonlinear plant and by the appropriate linearised model when random square wave type signals are applied to the model inputs.

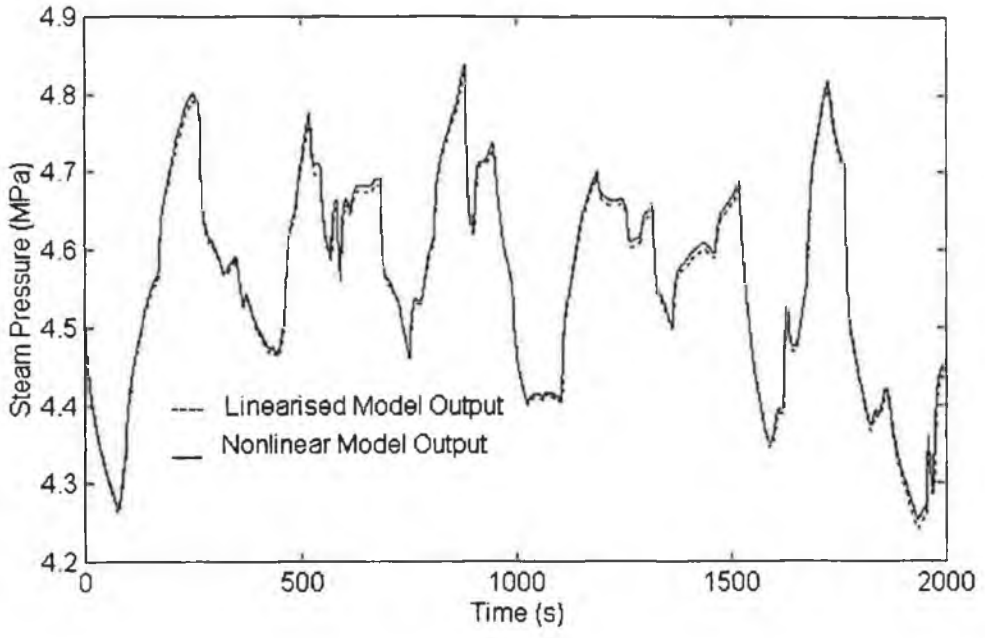


Fig. 5.4 Comparison of Modelled Main Steam Pressure for Linearised and Nonlinear Model

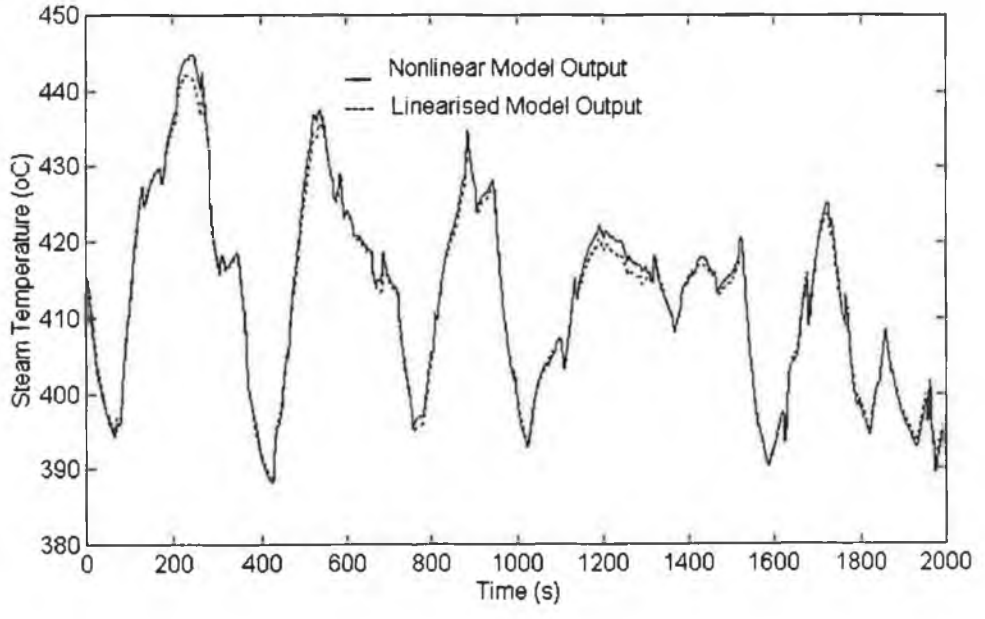


Fig. 5.5 Comparison of Modelled Main Steam Temperature for Linearised and Nonlinear Model

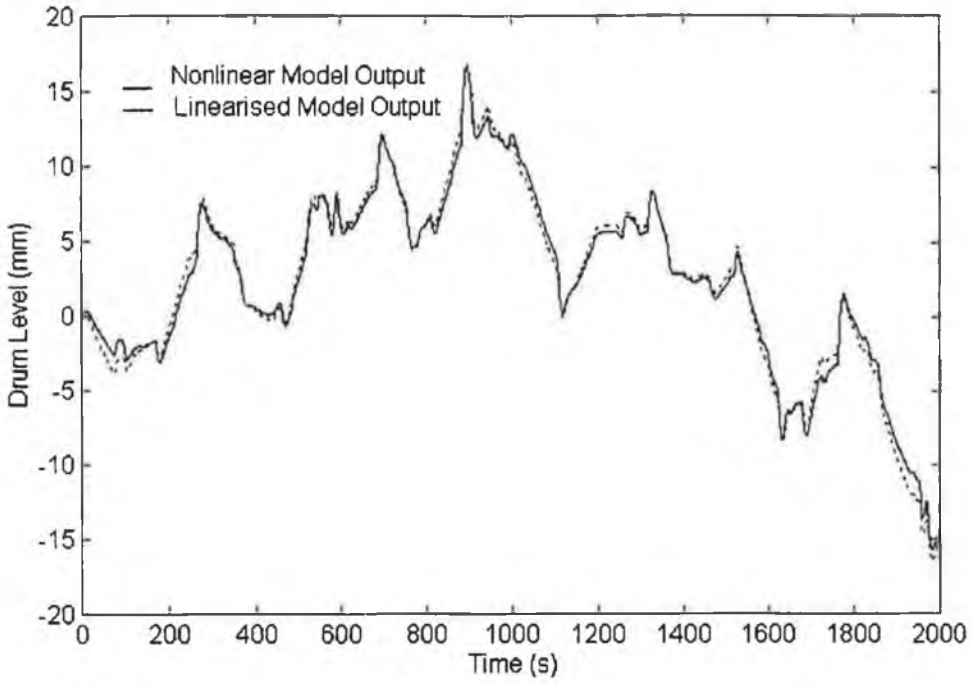


Fig. 5.6 Comparison of Modelled Drum Level for Linearised and Nonlinear Model

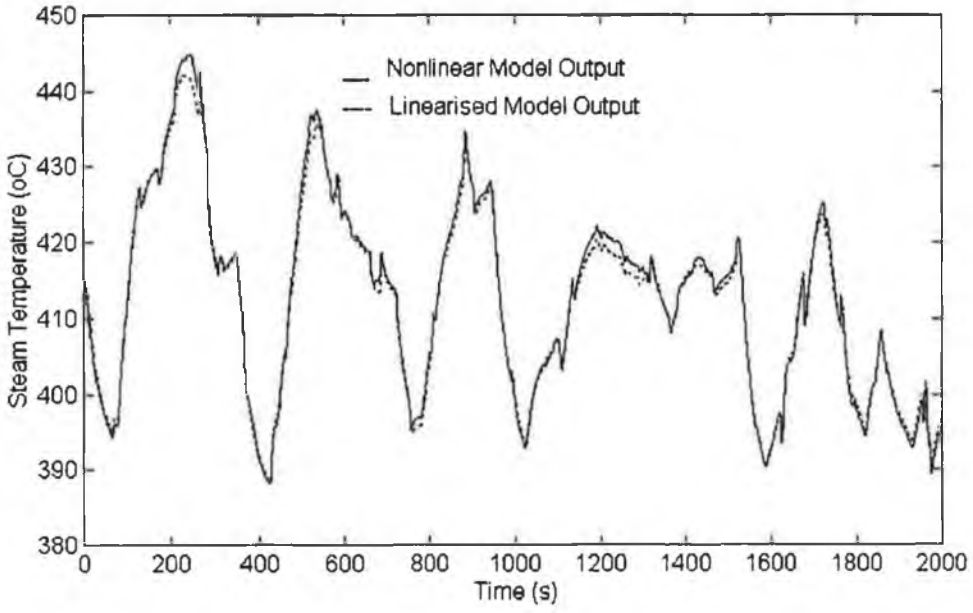


Fig. 5.7 Comparison of Modelled Percentage O_2 for Linearised and Nonlinear Model

The linearised model output is in good agreement with the nonlinear model output for all four modelled variables for a 5% perturbation around the linearised model equilibrium point.

5.6.2 Assessment of Linearised Model Validity for Large Disturbances

If the operating point moves away from the model equilibrium point, it is to be expected that the

linear model will not be able to model the nonlinear plant quite as closely. It is not possible to investigate this hypothesis by simply exciting the linearised model and nonlinear model with larger perturbations and comparing the output of both models. The problem is that random perturbations of greater than 5% may cause the nonlinear model to become unstable.

The solution is to generate validation data by perturbing the nonlinear plant about a steady-state operating point quite different from that at which the linearised model was obtained. In this instance the data was generated by exciting the nonlinear model around the 70% operating point. Again, the maximum perturbation of each input is 5% of the maximum allowable value for that input.

Two different linearised models are assessed. The first is obtained at equilibrium conditions for a 50% load, the second at equilibrium conditions for a 90% load. Both models are initialised in steady-state for a 70% load. For example, for the 90% linearised model, the steady-state state vector is:

$$x0_{70}^{90} = (I - A)^{-1} B(u0_{70} - u0_{90})$$

where

$x0_{70}^{90}$ = steady - state state vector for 90% linearised model at 70% load

$u0_{70}$ = steady - state input vector at 70% load

$u0_{90}$ = steady - state input vector at 90% load

The output of the 90% linearised model under these initial conditions is:

$$y0_{70}^{90} = Cx0_{70}^{90} + B(u0_{70} - u0_{90}) + y0_{90}^{90}$$

where

$y0_{70}^{90}$ = steady - state output vector for 90% linearised model at 70% load

$y0_{90}^{90}$ = steady - state output vector for 90% linearised model at 90% load

Under these initial conditions, there exists an error between the output of the nonlinear model at 70% and the output of 90% linearised model, which is equal to

$$e0_{70}^{90} = y0_{70}^{90} - y0_{70}^{70}$$

where

$y0_{70}^{90}$ = steady - state output vector for 90% linearised model at 70% load

$y0_{70}^{70}$ = steady - state output vector for 70% linearised model at 70% load

The validation input signals are then applied to both the 50% and 90% linearised models. This test assesses the performance of each model for perturbations of up to 25%. Fig. 5.8 to Fig. 5.11 shows the simulated value of the modelled variables as calculated by the nonlinear plant and by

the two linearised models The constant initial error has been subtracted from the output of both linearised models as it has the effect of obscuring other non-constant errors, which are more significant to control.

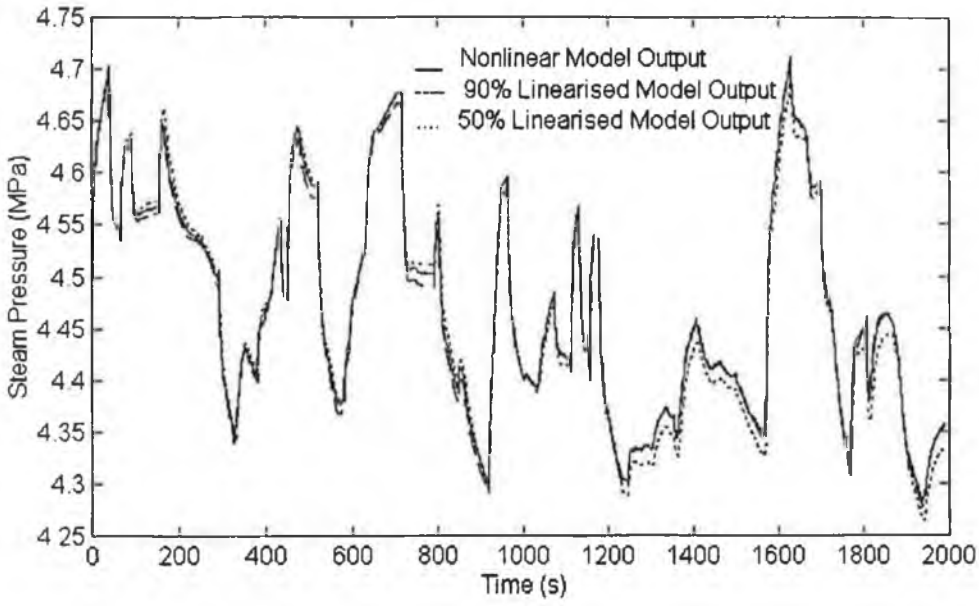


Fig. 5.8 Comparison of Modelled Main Steam Pressure for 50% Linearised, 90% Linearised and Nonlinear Model

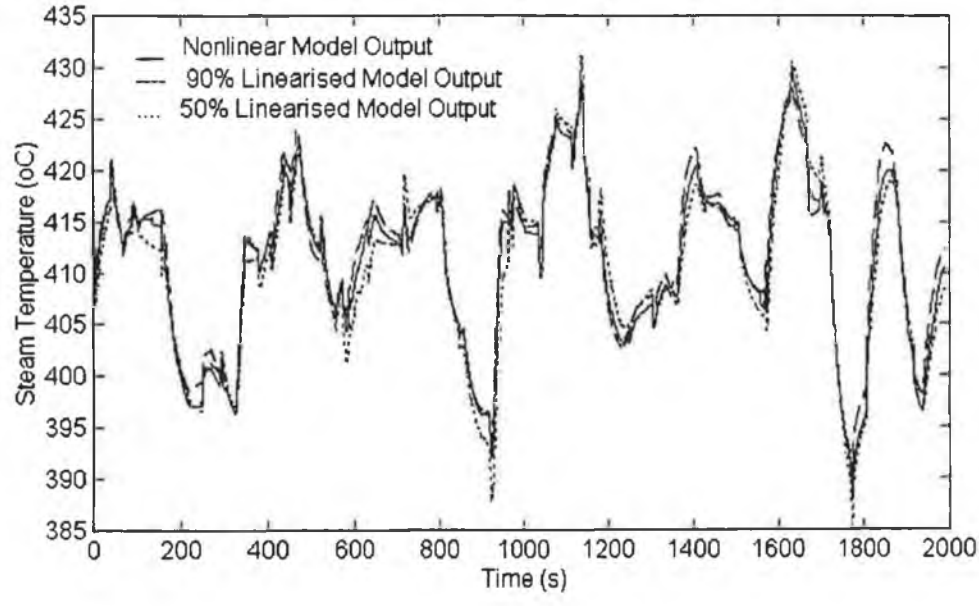


Fig. 5.9 Comparison of Modelled Main Steam Temperature for 50% Linearised, 90% Linearised and Nonlinear Model

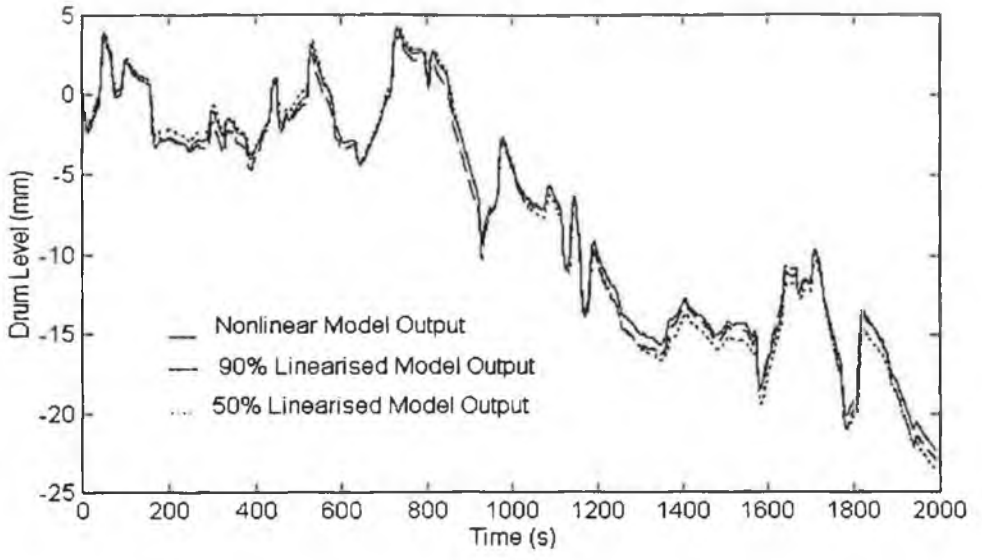


Fig. 5.10 Comparison of Modelled Drum Level for 50% Linearised, 90% Linearised and Nonlinear Model

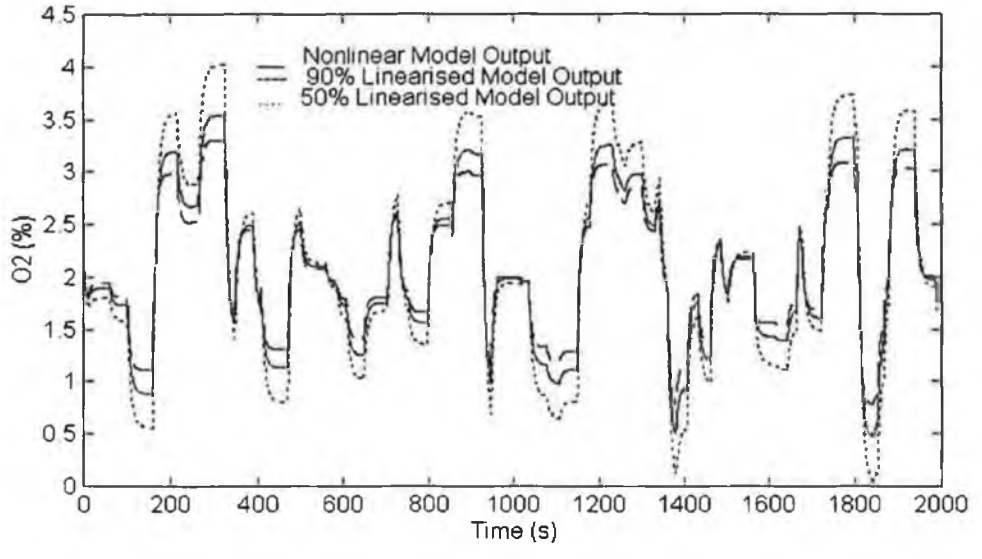


Fig. 5.11 Comparison of Modelled $\%O_2$ for 50% Linearised, 90% Linearised and Nonlinear Model

It can be seen from the graphs that there is some degradation in the performance of the linearised models at larger perturbations. It is interesting to compare the modelling errors for the 50% linearised model with those for the 90% linearised model. In many cases, the 50% modelling errors are opposite in sign to the 90% modelling errors - an effect which is particularly noticeable for the graph of percentage O_2 in the stack gases. This is due to the fact that the plant gains and frequency response increase (or decrease) monotonically from a 50% operating point to a 90% operating point. A consequence of this phenomenon is that the output of the 50% linearised model and 90% linearised model may be interpolated to generate an output which is closer to the

actual nonlinear plant output at 70%. A comparison of the sum squared error for the 50% linearised model output, the 90% linearised model output and their interpolated output, given in Table 5.4, confirms this:

Variable	SSE for 50% Model Output	SSE for 90% Model Output	SSE for Interpolated Output	Percentage Improvement
Steam Pressure	3.0434e11	1.1609e11	1.0439e11	101.4%
Steam Temperature	5.4012e3	3.7079e3	0.9943e3	358.1%
Drum Level	837.4874	546.0300	489.8968	41.2%
Percentage O_2	109.6867	34.8287	5.6803	1172.1%

Table 5.4 Comparison of Sum Squared Error for Linearised Models and Interpolated Models.

The second and third columns of Table 5.4 detail the sum squared error (SSE) over the output of the 50% linearised model and 90% linearised model respectively. The fourth column is the sum squared error over the interpolated output. The final column is the percentage improvement obtained by interpolating the output of the two linearised models. It is expressed as a percentage of the average error of the two linearised models, i.e.

$$\text{Improvement} = \frac{\text{SSE for interpolated output} - \text{mean(SSE for 50\%, SSE for 90\%)}}{\text{mean(SSE for 50\%, SSE for 90\%)}} \times 100 \qquad (5.29)$$

The interpolated output is plotted against the nonlinear model output for each of the controlled variables to demonstrate the improvement achieved by interpolation:

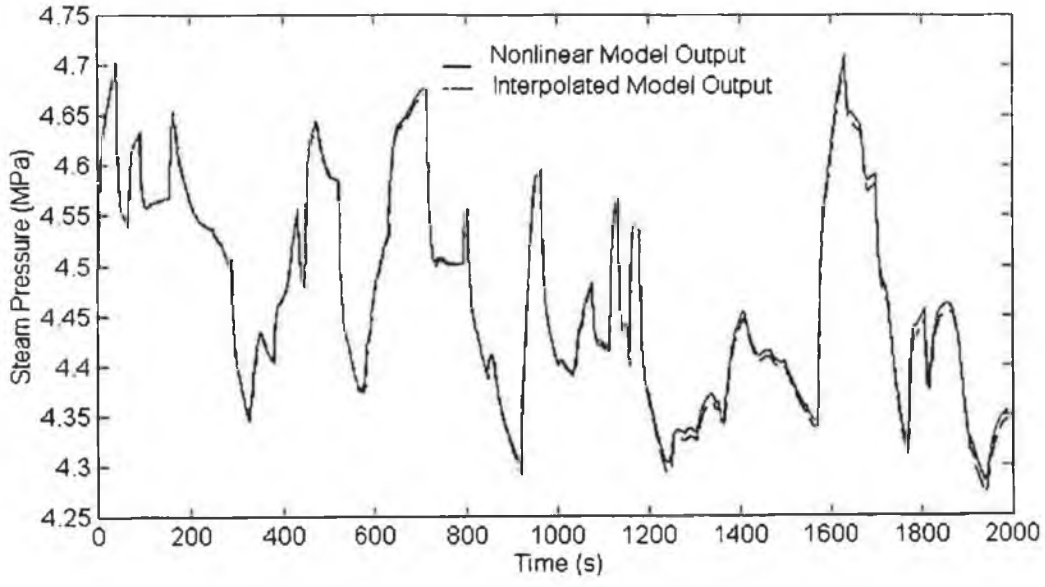


Fig. 5.12 Comparison of Modelled Main Steam Pressure for Interpolated Linearised Output and Nonlinear Model

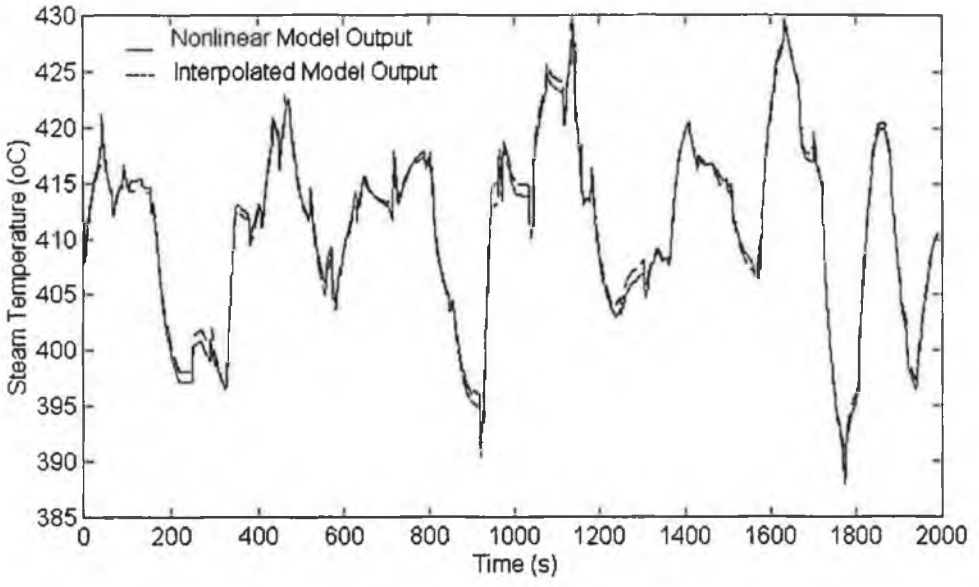


Fig. 5.13 Comparison of Modelled Main Steam Temperature for Interpolated Linearised Output and Nonlinear Model

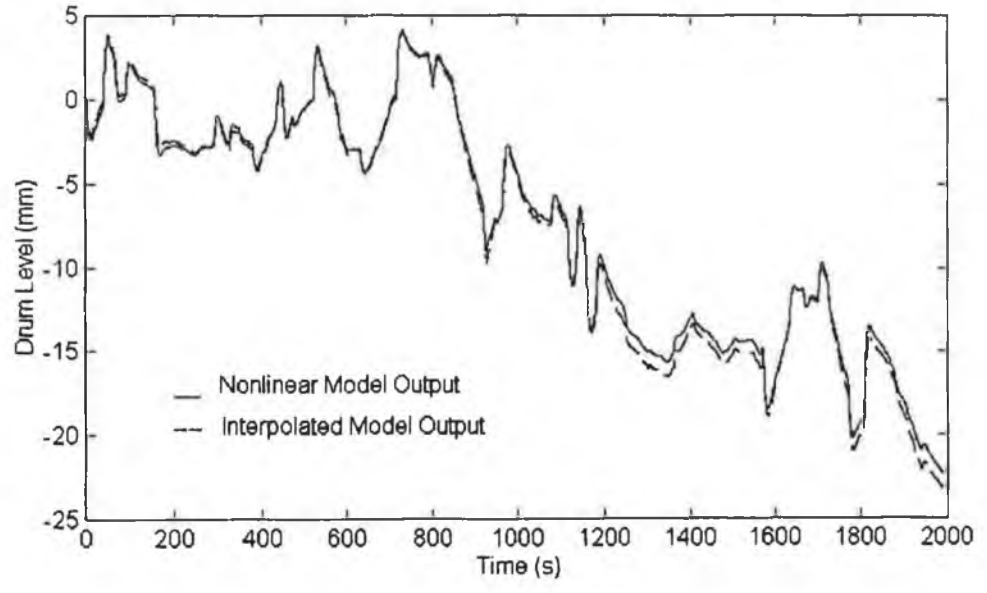


Fig. 5.14 Comparison of Modelled Drum Level for Interpolated Linearised Output and Nonlinear Model

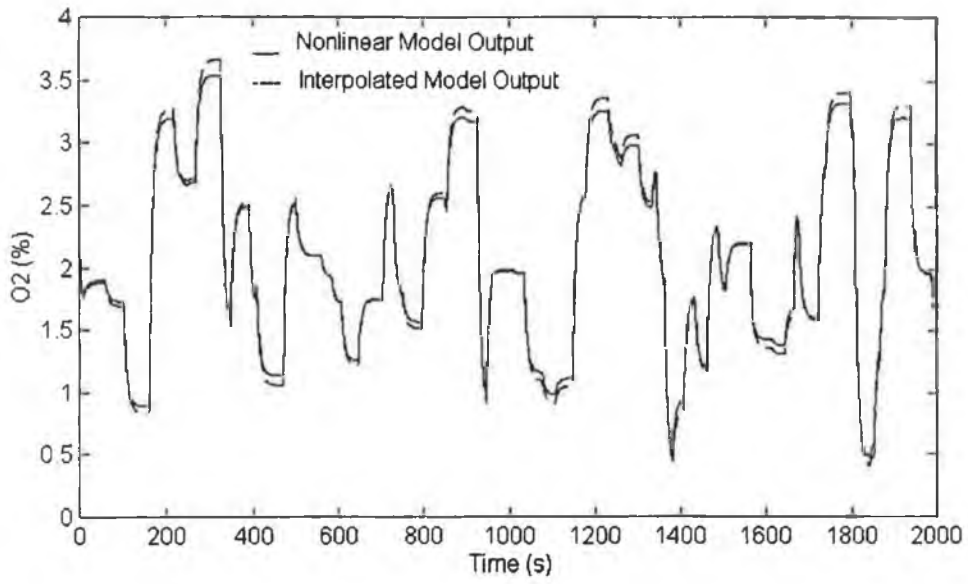


Fig. 5.15 Comparison of Modelled Percentage O_2 for Interpolated Linearised Output and Nonlinear Model

In each instance a considerable improvement has been attained by interpolating the output of the two linearised models.

5.7 Conclusions

The results confirm that it is possible to generate good linearised models of the boiler process. The linearised models represent the nonlinear plant very well for small perturbations around their equilibrium points. However, for larger perturbations, some degradation in performance is evident. This has direct consequences for model-based control - if the internal controller model is not capable of representing the plant well, controller performance will suffer. One way of overcoming this problem is to increase the number of linearised models used to cover the plant operating range. An alternative strategy is to interpolate between the available linearised models at those operating points which are between the equilibria points of the linearised models. The results show that interpolation can be used to improve modelling performance without increasing the number of linearised models. In this case, it was shown that provided interpolation is used, three linearised models, obtained at the 10%, 50% and 90% operating point can adequately model the boiler over its entire operating range.

6. Development of Neural Network Boiler Models

6.1 Introduction

Predictive control is a model-based strategy which may use any type of model to predict the future behaviour of the plant. Typically, the model is linear for reasons of simplicity and familiarity. However, given the nonlinearity of the boiler process it is likely that better control can be achieved through a nonlinear approach as a nonlinear model should be capable of modelling the plant more accurately than a set of linear models.

A first principles nonlinear model of the boiler has already been developed. However, this model is not suitable for use as an internal controller model, on account of its complexity - it comprises 20 nonlinear differential equations and a large number of ancillary equations. Instead, neural network models which have a smaller computational overhead are employed as the internal controller models. Neural networks offer an interesting alternative to conventional modelling and control methods. There are no restrictions on what can be modelled as it has been shown that neural networks are capable of approximating arbitrary nonlinear functions (Narenda and Parthasarathy (1990)). The process model can be quickly generated as neural networks "learn" to model the system using plant data. Neural networks have a parallel structure which promises speed and fault tolerance. They can be adapted on-line to cater for time-variant processes. They are multivariable systems and can accept a variety of inputs simultaneously e.g. qualitative and quantitative inputs. Finally, if trained correctly, they have ability to generalise and cope with situations not presented in the training data (Hunt *et al* (1992))

The neural networks models, described in this chapter, are built and trained using algorithms from the "*MATLAB Neural Network Toolbox*" (Demuth and Beale (1994)). This is a neural network development tool for PC's, which offers a variety of network structures and training algorithms.

6.2 Description of Boiler Models

There are four variables to be controlled:

1. Main steam pressure
2. Main steam temperature
3. Drum level
4. Percentage O_2 in stack gases (not including the pure transportation delay)

A model-based predictive control strategy requires that the future output of each of these fours variable can be predicted for a given set of inputs. In the case of the linear predictive controller, it was decided to use the following three models to predict these variables:

1. Model of main steam pressure and drum level
2. Model of main steam temperature
3. Model of percentage O_2 in stack gases, (not including the pure delay)

The rationale for this choice of models, as well as the input and output variables of these models are described in Chapter 5.

6.3 Overview of Neural Networks

Artificial neural networks provide a nonlinear mapping from an input, u, to an output, y:

$$\begin{aligned} & y = f(u) \\ \text{where} \quad & u, y \text{ are either scalors or vectors} \end{aligned} \tag{6.1}$$

The mapping mechanism originated as an attempt to copy the behaviour and thus performance of biological nervous systems. Neural networks are composed of many nonlinear computational elements operating in parallel and connected by weighted links. These weights are adapted during training to generate a network with specific capabilities such as pattern recognition or dynamical system modelling. A variety of network structures have been devised, each of which has particular ability to perform a certain type of task. This description is focused on the multi-layer perceptron structure, which has proven ability in the field of system modelling. (Bhat *et al* (1990), Irwin *et al* (1995)). A sample two layer perceptron network structure with 3 inputs in the first (hidden) layer and 2 neurons in the output layer, is shown in Fig. 6.1.

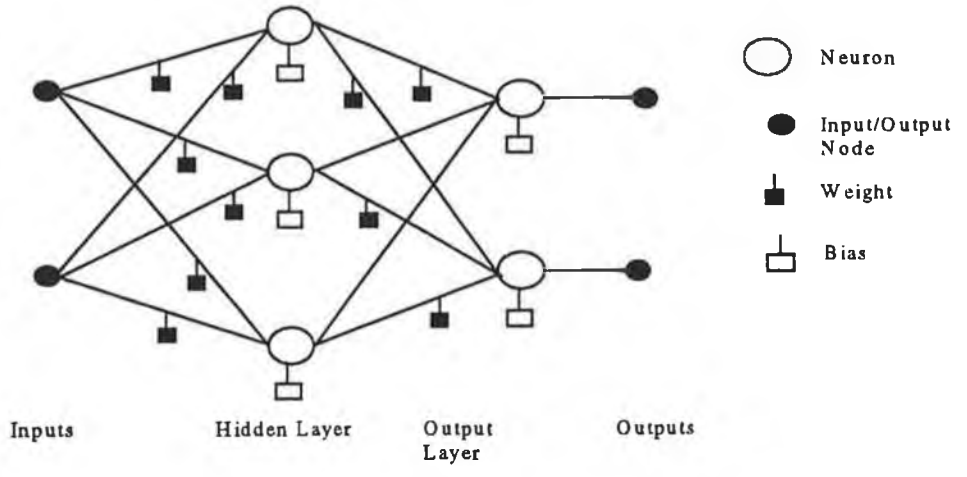


Fig. 6.1 2-Layer Perceptron Structure

Each neuron sums its weighted inputs and, optionally, biases the sum, to produce an activation value, v_i, defined as:

$$v_i = \sum_j w_{ij} x_j + b_i$$

where

v_i = activation for neuron, i

w_{ij} = weight from hidden neuron, j, or from an input, j, to neuron i (6.2)

b_i = bias for neuron, i

x_j = output of hidden neuron, j, or input, j

The neuron then performs a nonlinear transformation on the activation value to generate the neuron output, x_i:

$$x_i = F(v_i)$$

where $F(v_i)$ = nonlinear transformation (6.3)

The type of nonlinear transformation used is determined by the intended application of the neural network. A hard nonlinearity, is suitable for networks which are to perform classification tasks, such as voice recognition. Smooth nonlinearities, such as tan-sigmoid or log-sigmoid are more suitable for networks which are to model continuous variables. Fig. 6.2 shows a hard limiter, a tan-sigmoid and a log-sigmoid transfer function.

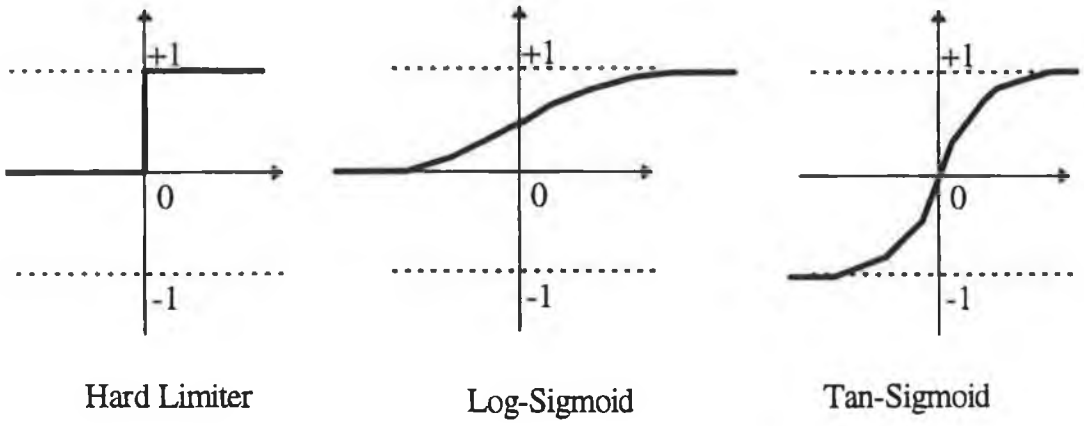


Fig. 6.2 Hard Limiter, Log-Sigmoid and Tan-Sigmoid Nonlinearities

The neurons in the output layer, often use a linear transformation in preference to a nonlinear transformation as this increases the network output range beyond [-1,1].

The effect of any input on a neuron output is determined by level of the input and by the

magnitude of the weight applied to that input. By correctly selecting the network weights, it is possible for a neural network to model many types of nonlinear behaviour. Selection of the network weights is an iterative process, referred to as network training. It requires a set of input and output data which is representative of the process that the neural network is to emulate. This input and output data, referred to as training data, is crucial to the success of the neural network. The network is trained purely by example, so the training data should cover the whole range of possible input data behaviour. For systems with memory, the training data should also cover the range of possible output data behaviour.

Prior to training, the neural network weights and biases are initialised to small random values. Good initialisation of the weights prevents saturation of the network nonlinearities, which hinders network training. The network is presented with input training data, and the network output is calculated. The network output is compared with the output (or target) training data and the weights are adapted in a manner which should reduce the error between the network output and the target data. The network is again presented with training data and the procedure is repeated for a specified number of iterations or until a specified error criteria has been achieved. After training, it is important to validate the network against a separate set of data, since it is possible that a network may "learn" the training set very well, but be incapable of generalising (Hechtl-Neilsen, (1990). The training and validation sequence is summarised in Fig. 6.3 as follows:

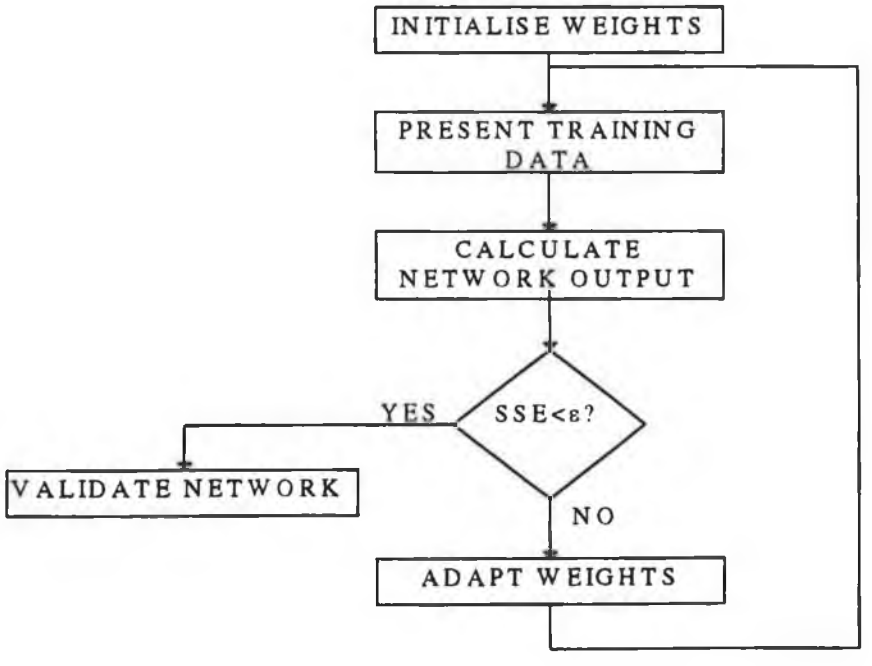


Fig. 6.3 Neural Network Training and Validation Sequence

The termination criterion used in the network training and validation sequence described in Fig. 6.3 is equal to the sum square error (SSE), which is defined as:

$$SSE = \sum_{k=1}^{n} (T(k) - \hat{x}(k))^2$$

where

SSE = sum squared error (6.4)

$\hat{x}(k)$ = output predicted by network, for training vector, k

$T(k)$ = target (or true) output, for training vector, k

n = size of training set

Training is stopped when the SSE is lower than a specified criterion.

The way in which the weights are adapted is specified by the training algorithm. The mostly widely used training algorithm for multi-layer perceptrons is the back-propagation algorithm, which was first proposed by Werbos (1974) and further developed by Rumelhart and McClelland (1986). Previous to the development of the back-propagation algorithm, there was no effective means of training multi-layer perceptrons. For single layer perceptrons, the weights could be adapted using the error between the network output and the target data. The same weight adaptation technique could not be applied to hidden layer neurons however, as there was no method available for calculating the error at the output of a hidden layer neuron. The back-propagation algorithm established a method for back-propagating the errors at the output layer neurons back through the network to the hidden layer neurons.

To adapt the weights in the output layer of a neural network, the derivative of the error at the network output is calculated with respect to the weights. For output neuron, i, the error derivative, δ_{ij}, is calculated as follows:

$$E = T_i - x_i$$

$$\delta_{ij} = \frac{\delta E_i}{\delta w_{ij}}$$

where

T_i = Target value for neuron i (6.5)

x_i = Output of neuron i

w_{ij} = Weight to be adapted

δ_{ij} = Error derivative with respect to w_{ij}

The sign of the derivative informs whether the weights should be increased or decreased. The magnitude of the weight adaptation is dictated by an adaptation gain, η, as follows:

$$w_{ij}(t+1) = w_{ij}(t) - \eta\delta_{ij}$$

where (6.6)

η = adaptation gain

For a hidden layer neuron, the error derivative for each of the output layer neurons must be back-propagated through the weights which connect the hidden neuron to the output neurons. This is forward propagation in reverse - the error derivatives for each of the interconnecting weights are weighted and summed. The partial derivative of the result is found with respect to the hidden layer neuron weights. The resulting error derivative for a hidden layer neuron equals:

$$\delta_{pq} = \frac{\delta F'}{\delta w_{pq}} \sum_i w_{ip}\delta_{ip}$$

where

δ_{pq} = partial derivative of error at hidden layer neuron, p

$\frac{\delta F'}{\delta w_{ip}}$ = derivative of nonlinear transformation at hidden layer neuron, p (6.7)

w_{ip} = weight from hidden layer neuron, p, to output layer neuron, i

δ_{ip} = partial derivative of error from hidden layer neuron, p, to output layer neuron, i

Using this approach, the error is back propagated through each of the network hidden layers until the input layer is reached. The hidden layer weights are adapted in the same way as the output layer weights.

The weight adaptation uses a gradient descent algorithm, so-called because the weight adaptation proceeds along the negative gradient of an error surface. The major flaw of gradient descent algorithms is that learning may finish in a local, instead of a global minima. The problem is demonstrated in Fig. 6.4, using the error surface of a neural network with a single weight.

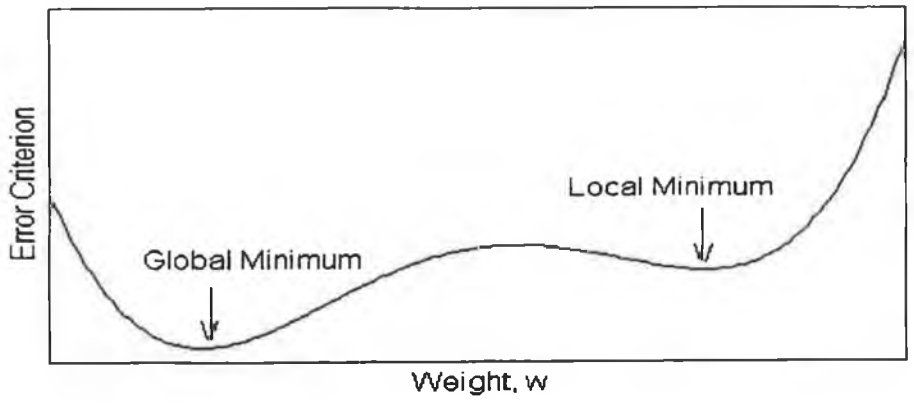


Fig. 6.4 One dimensional error surface

It is evident that if training finishes in a local minima that the performance of the trained network will be sub-optimal. A further and important consequence of this problem, is that the performance of a neural network is very sensitive to the initial condition of the network weights. Two networks of identical structure but initialised with different sets of weights, could yield very different results.

The possibility of training terminating in a local minima can be reduced slightly by the inclusion of a momentum term, α. Each new weight adaptation is equal to a weighted sum of the proposed weight adaptation and the previous weight adaptation. Incorporating a previous weight adaptation continues weight adaptation in a direction which was previously useful and may be sufficient to drive the network over small "bumps" in the error surface. The weight adaptation rule, including momentum is:

$$w_{ij}(t+1) = w_{ij}(t) + (1-\alpha)\eta i \delta_{ij} x_j' + \alpha(w_{ij}(t) - w_{ij}(t-1) \tag{6.8}$$

where

α = momentum term $(0 < a < 1)$

6.4 Neural Network Model Development

6.4.1 Overview

A neural network is defined by the type of nonlinearities used, the configuration or topology of these nonlinearities and by the network training algorithm. In particular, the following decisions must be made:

1. Network Structure
    - Data Pre-processing
    - Network Input Structure
    - Network Topology
    - Type of Nonlinearities
    - Network Size

2. Network Training
    - Training Algorithm
    - Initialisation of Weights
    - Training Data Set
    - Training Period

To date, there are few hard and fast rules about how to choose the best neural network structure or training method for a particular task. The best approach seems to be a combination of prior knowledge and systematic comparison of particular methods.

The back-propagation algorithm does not guarantee that training will finish in a global minimum, with the consequence that neural networks are sensitive to initial conditions. Given the stochastic nature of neural network initialisation, a systematic comparison of each method should be based on a large number of Monte-Carlo training runs. This was not feasible in this instance on account of the large training set and network size required for this task. A typical training run requires up to 24 hours of computation on a 160 MHz Pentium (586) PC with 16 MB of RAM. Instead, 3 training runs are carried out for each type of network, and the minimum validation error achieved over these 3 runs is presented. The minimum validation error is used in preference to the mean or median error, for two reasons. Firstly, the ultimate goal of these comparisons is to identify the network with the minimum validation error. Secondly, the only true basis of comparison for any two networks is the validation error at the global minima of both networks. If two networks of identical structure, yield different validation errors due to different initialisations, this implies that training for one or both networks finished in a local minimum. The network with the smaller validation error is closer to the true global minima and thus provides more information about the true potential of the structure.

The validation error criterion used throughout this chapter is equal to the mean squared error (MSE) over the validation set, for a 10-step ahead prediction:

$$MSE = \frac{\sum_{n} (y(n+10) - \hat{y}(n+10))^2}{n} \tag{6.9}$$

where MSE = mean square error
$y(n)$ = plant output at n
$\hat{y}(n)$ = predicted output at n
n = length of validation set

6.4.2 Neural Network Training

6.4.2.1 Training Algorithm

The network is trained using the back-propagation algorithm with momentum and an adaptive learning rate. The adaptation gain is increased if weight adaptation has resulted in an increased error and the new weights are discarded. In effect, this produces a very small adaptation gain

which ensures that the network consistently converges. The training data is presented in batch to the network for faster training.

6.4.2.2 Training and Validation Data

The training and validation data for each of three neural network models is generated using the first principles boiler model, described in Chapter 4. This data is generated using a different section of the first principles model for each of the neural network models. In the case of the neural network model of the entire boiler process, the training data is generated by exciting the inputs of the first-principles model such as feedwater flow rate and fuel flow rate and simulating the first principles model to find the resulting output signals. Fig. 6.5 illustrates the data collection strategy for the neural network model of a full boiler process, using broader lines to represent input and output probes. The input probes excite the first principles model while the output probes measure the resulting model output. The following legend is used:

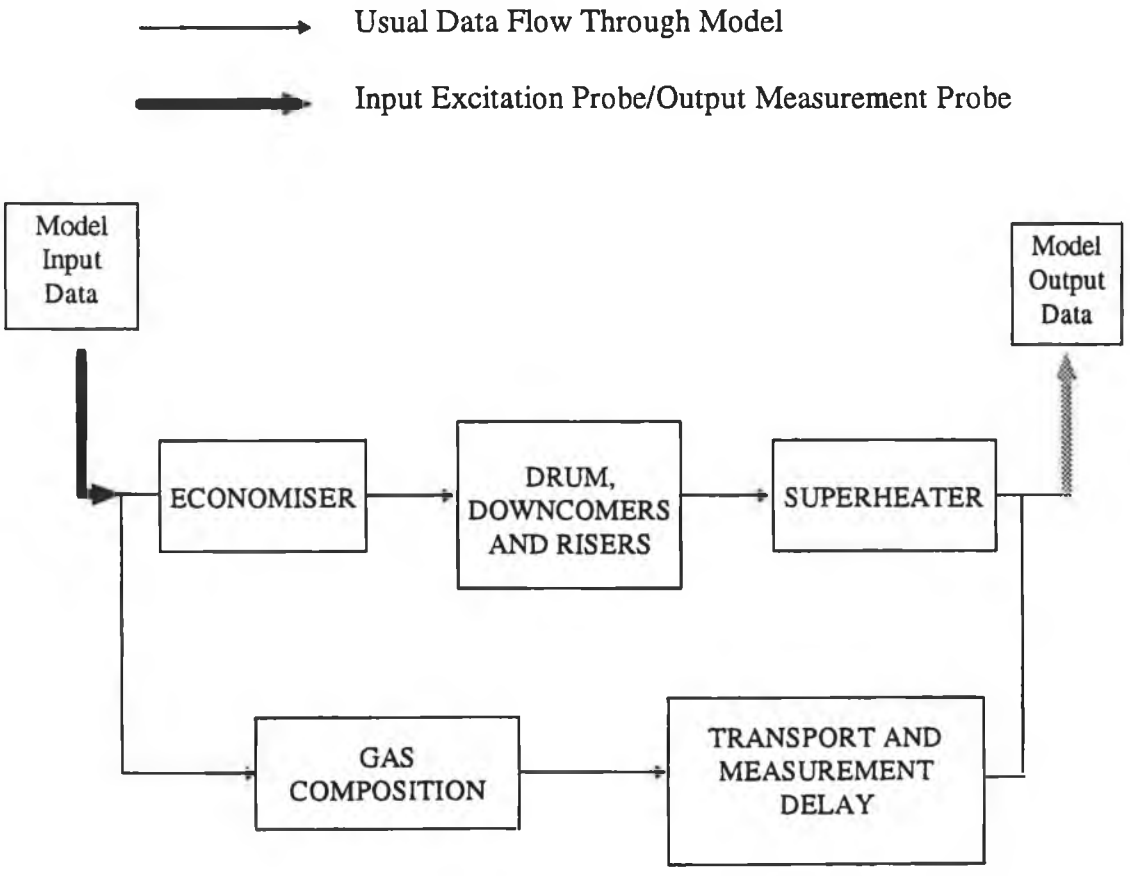


Fig. 6.5 Training Data Generation for Neural Network Model of Entire Boiler Process

The neural network gas composition model is required to model gas composition in the furnace and stack, excluding pure time delays. The gas composition dynamics includes a time delay of

approximately 3s to 8s, depending on the plant operating point. Using a neural network to model this time delay, would necessitate over 80 lagged inputs, as the sampling time for the non-delay part of the gas composition model is approximately 0.1s. The inclusion of this many, inactive inputs would seriously compromise the performance of the neural network model. This problem is overcome by using a non-neural network model to estimate and model the time delay and a neural network model to model the gas composition process, excluding time delay.

The training data for the neural network model is generated using the first principles model of gas composition excluding pure time delays. The training data is collected by isolating one section of the first-principles model as shown in Fig 6.6. The inputs of this section are true boiler inputs - air flow rate and fuel flow rate. The outputs of the isolated section are not measurable variables. It follows that it is impossible to collect an equivalent set of training data from a real boiler process.

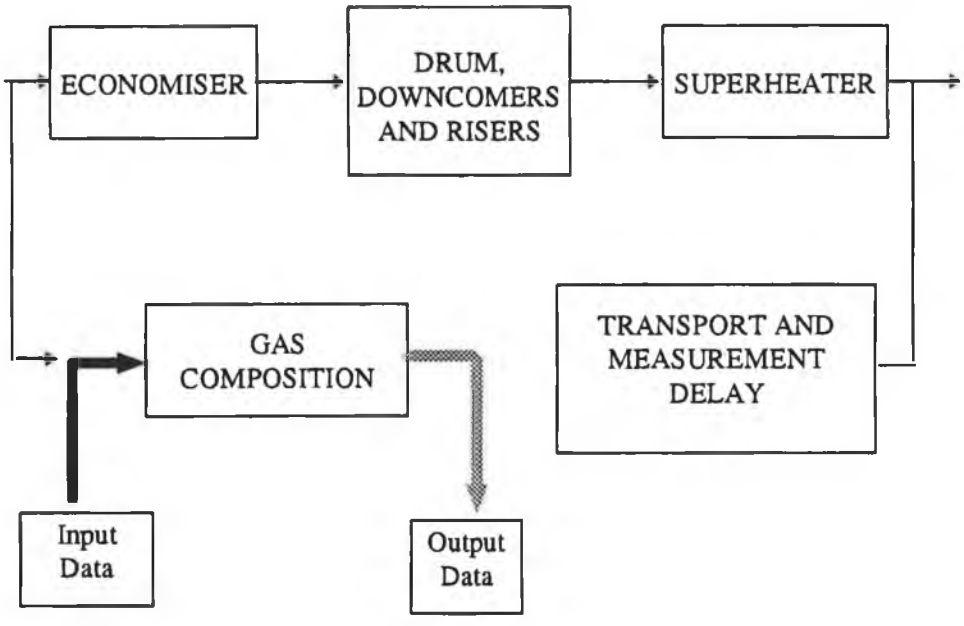


Fig. 6.6 Training Data Generation for Neural Network Model of Gas Composition

An initial attempt was made to train the neural network model of the superheater using data which was collected in the same way as training data for the neural network model of the entire boiler process . The boiler inputs were excited and the resulting superheater input and output signals were used as neural network input and output training data. This initial approach is illustrated in Fig. 6.7.

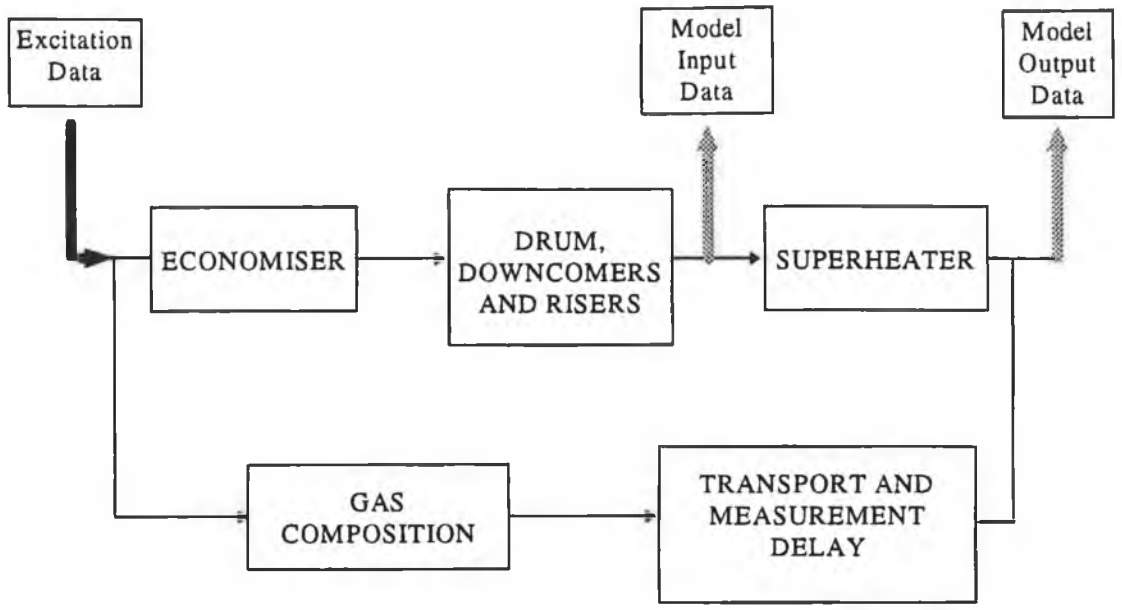


Fig. 6.7 Initial Training Data Generation for Neural Network Model of Superheater

It proved impossible to train the neural network model of the superheater using this data due to correlation between many of the variables in the input training set. For example, the neural network inputs for the superheater model include load, drum pressure and drum temperature. A change in load causes variations in both main steam pressure and main steam temperature. As a result the network is unable to correctly identify the relationship between the input and output training data.

A set of uncorrelated input training data is generated by isolating the superheater section of the first principles model and exciting the inputs of this section directly. It would not be physically possible to do this on a real boiler plant. The approach is shown in Fig 6.8.

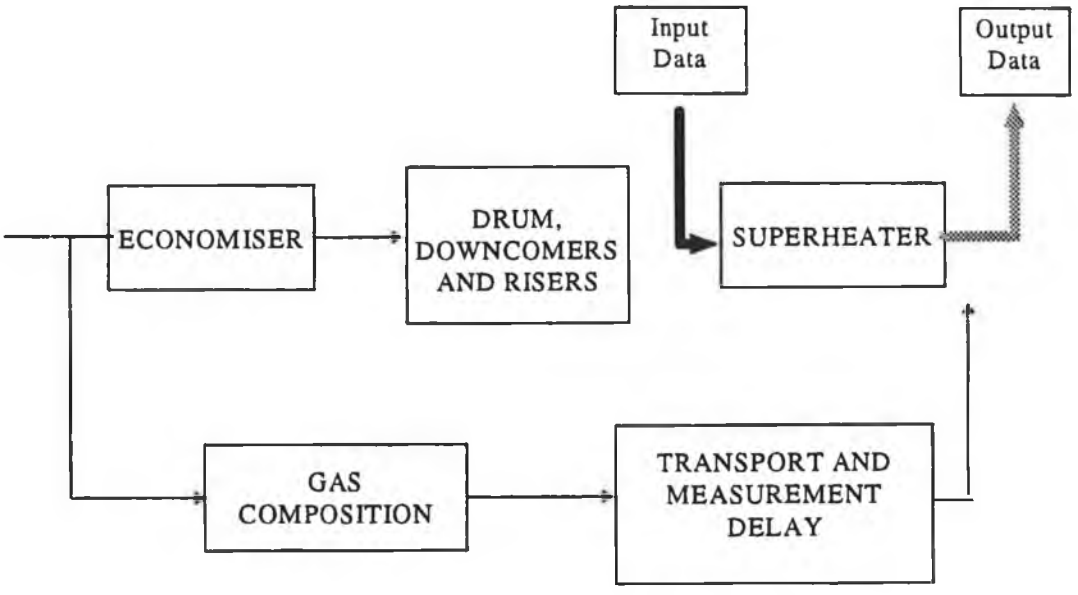


Fig. 6.8 Training Data Generation for Neural Network Model of Superheater

In each case the training and validation set is generated by applying a random square wave-type sequence of varying amplitude and frequency to the inputs of the appropriate section of the first-principles model. It is vital that the training data excites the first-principles model over its entire operating range in order to provide the network with examples of the plant behaviour at every operating point. However, it is not feasible to excite the open-loop first-principles model with large perturbations (which cover the full operating range of the model) as this leads to unstable operating conditions. Training data is generated by exciting the first-principles model with small perturbations around a steady-state operating point. In order to cover the complete operating range of the first-principles model, the training data is generated at several different operating points.

Fig. 6.9 shows a sample of the fuel mass flow rate training sequence, expressed as a percentage of fuel flow at full load. The maximum variation in each of the input signals is ±5% of its maximum value and the training data is collected at the following operating points: 10%, 20% 30%, 40%, 50%, 60%, 80%, 100%.

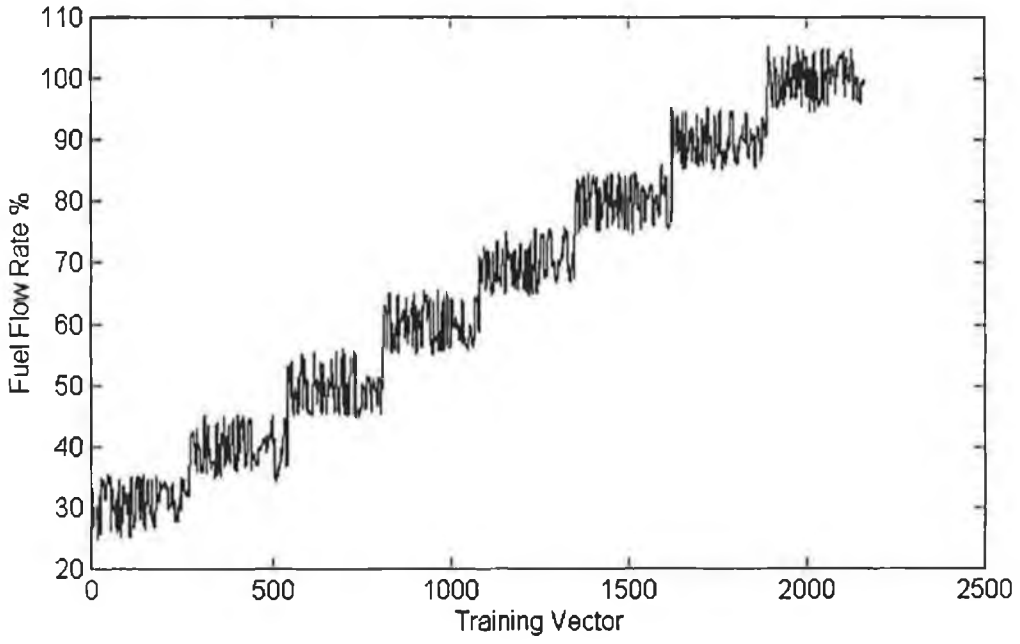


Fig. 6.9 Fuel Mass Flow Rate Training Data Set

The first-principles model is excited at each of its inputs by a square wave-type signal of variable magnitude and variable switching interval. The maximum magnitude of the square wave is, as stated previously dictated by stability considerations and is equal to ±5% of the maximum input value. The minimum and maximum values of the switching interval are dictated by the first-principles model dynamics. If the switching interval is too short, the output will not have time to

reach its new value and will remain close to the original steady-state value. Fig. 6.10a and 6.10b illustrate this problem:

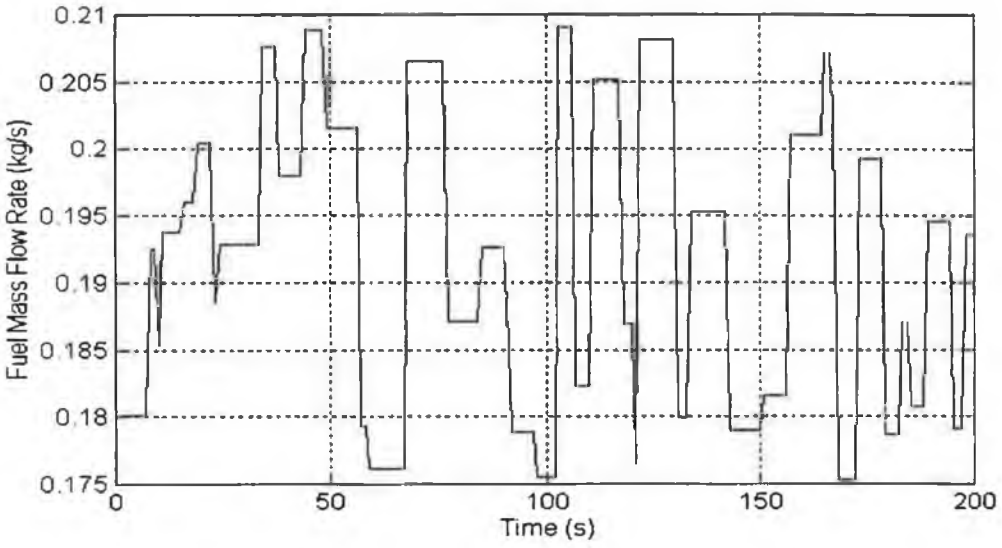


Fig. 6.10a Input Training Sequence (Fuel Mass Flow Rate) Generated Using Short Switching Interval

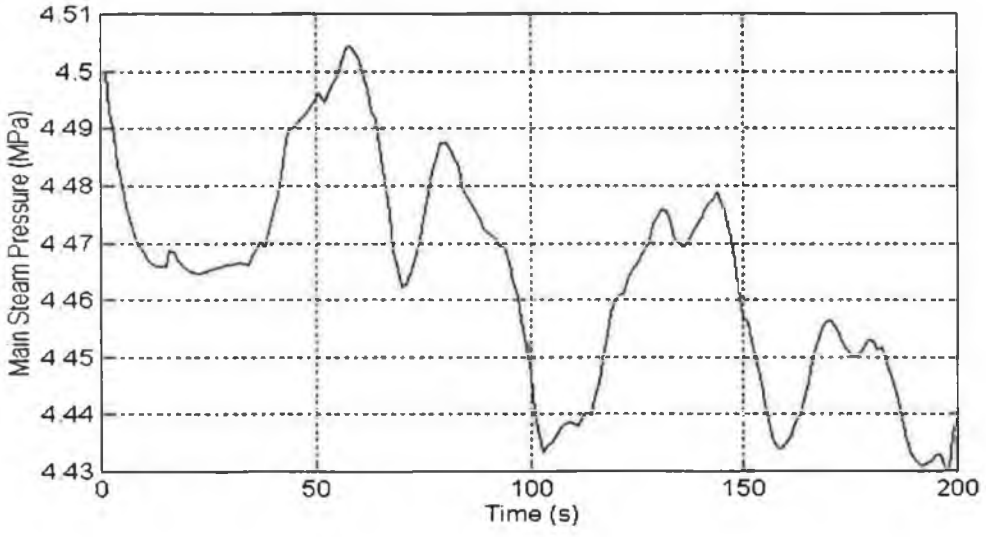


Fig. 6.10b Output Training Sequence (Main Steam Pressure) Generated Using Short Switching Interval.

It is clear from Fig. 6.10 that pressure remains within [4.43,4.51 MPa] despite large variations in fuel flow. It is extremely likely that main steam pressure will exceed this range during normal operation, thus introducing the neural network model to a situation which it has not "learned". Alternatively, if the switching interval is too long, the first-principles model will not be

sufficiently excited and the training data will not contain sufficient information about model behaviour. To prevent these problems, the minimum switching interval is chosen equal to the 10% response time of the first-principles model and the maximum switching interval equal to the 95% response time of the plant. Fig. 6.11 demonstrates how the input training sequence is generated.

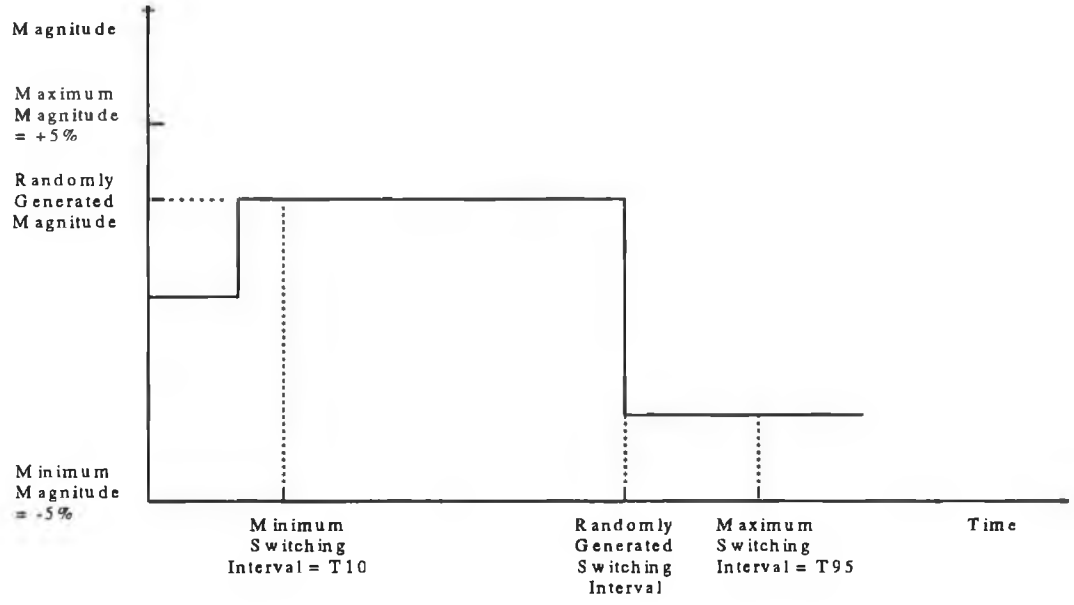


Fig. 6.11 Input Sequence Generation

Fig. 6.11 shows that the switching interval and magnitude of the input signal are selected randomly, between the maximum and minimum allowed values.

Given the stiff nature of the boiler system, there is a conflict between the 95% response time of the slow and fast subsystems. The maximum switching interval must be dictated by the slowest dynamics, for the reason outlined above. However, this has the consequence that the faster boiler dynamics are not excited for much of the training sequence. More value can be added to the training sequence by the inclusion of a random noise component with a smaller maximum amplitude and higher frequency dynamics. This is added to the square wave-type signal to excite the high frequency boiler dynamics.

The inclusion of this high frequency component greatly increases the amount of information which can be contained in a training sequence of given length. Fig. 6.8a shows a typical scaled square wave type input sequence - fuel mass flow rate. Fig. 6.8b shows this input sequence with the addition of random noise. Fig. 6.8c shows the resulting boiler output sequence - main steam pressure.

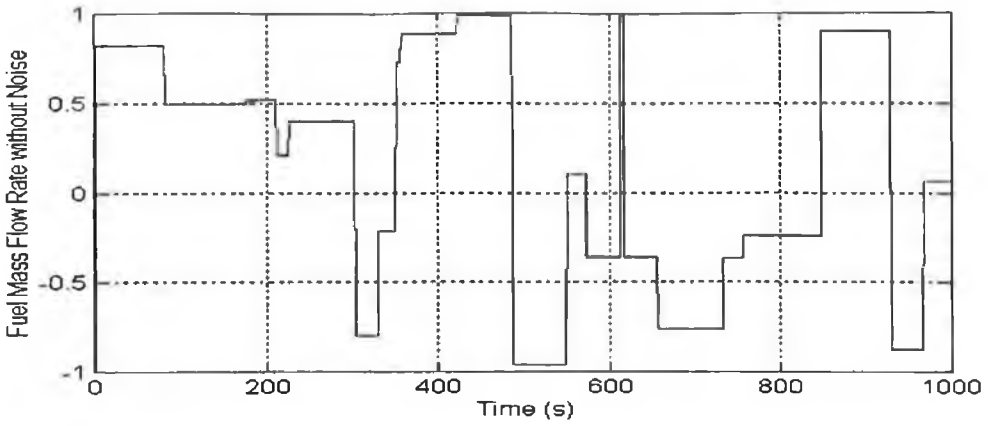


Fig. 6.8a Typical Input Training Sequence - No Noise

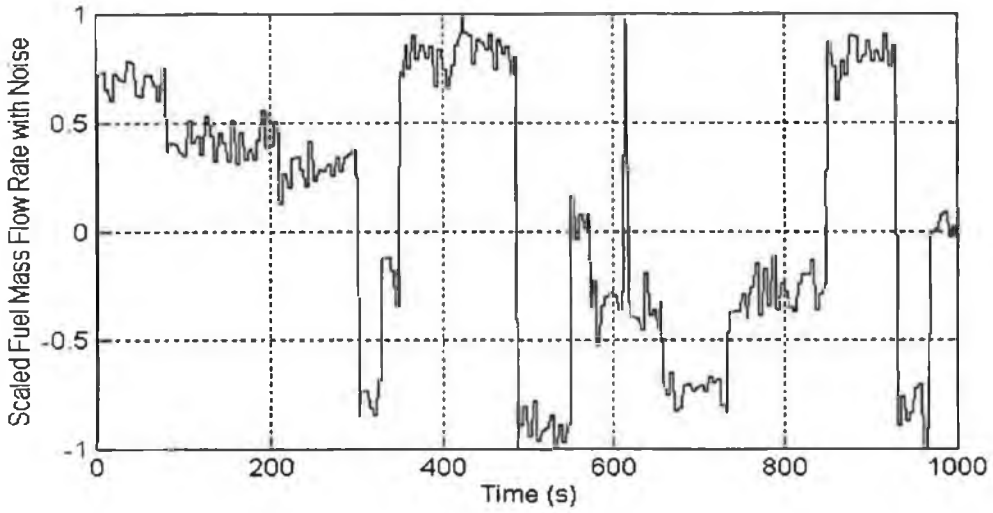


Fig. 6.8b Typical Input Training Sequence Including Noise

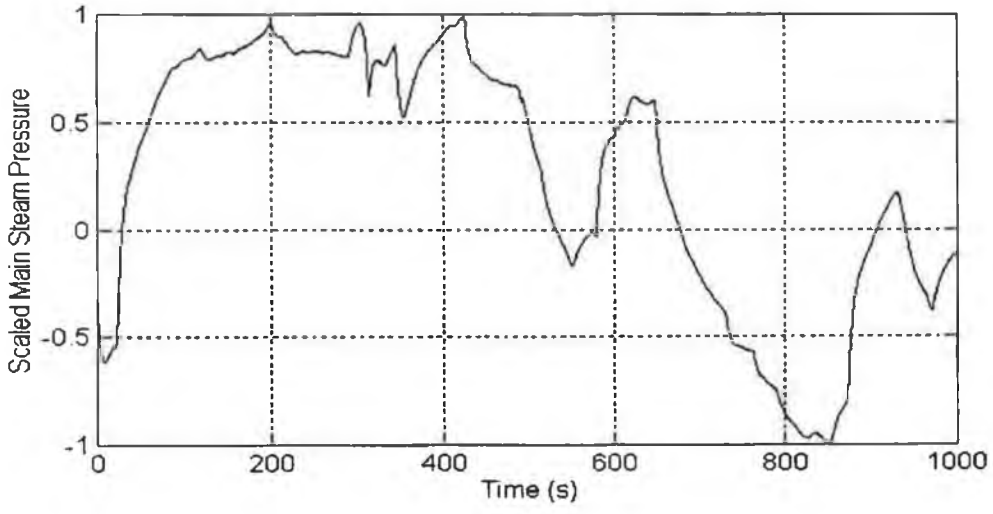


Fig. 6.8c Typical Output Training Sequence (Not Differenced)

The actual training set is created by rearranging the input and output variables in a matrix, as follows:

$$I=\begin{bmatrix} u(t) & u(t-1) & y(t) & y(t-1) \\ u(t-1) & u(t-2) & y(t-1) & y(t-2) \\ \cdots & \cdots & \cdots & \cdots \\ u(t-n+1) & u(t-n) & y(t-n+1) & y(t-n) \end{bmatrix},\quad O=\begin{bmatrix} \Delta y(t) \\ \Delta y(t-1) \\ \cdots \\ \Delta y(t-n+1) \end{bmatrix}$$

where I = input training data
O = output training sequence
$u(t)$ = row vector of boiler input data
$y(t)$ = row vector of boiler output data
$\Delta y(t) = y(t+1) - y(t)$
n = number of training vectors

(6.10)

It can be seen from Fig. 6.8c, that for a sampling period of 1s, that there is little variation between the value of an input or output variable between sampling instances. A reduced training set can be obtained by sub-sampling the original training set. Every 10th data vector of the collected data was used in the training set.

$$I=\begin{bmatrix} u(t) & u(t-1) & y(t) & y(t-1) \\ u(t-10) & u(t-11) & y(t-10) & y(t-11) \\ \cdots & \cdots & \cdots & \cdots \\ u(t-n+1) & u(t-n) & y(t-n+1) & y(t-n) \end{bmatrix},\quad O=\begin{bmatrix} \Delta y(t) \\ \Delta y(t-10) \\ \cdots \\ \Delta y(t-n+1) \end{bmatrix} \qquad (6.11)$$

The reduced training set contains nearly the same information as the original training set and allows for considerably shorter training times. The final training set contains 3000 input and output vectors. The validation set is not sub-sampled and contains 2800 input and output vectors, which are representative of the plant behaviour over the plant operating range.

6.4.2.3 Initialisation of Weights

The weights may be initialised either by a symmetric random number generator or by Nguygen-Widrow random generator, which takes into account the range of the training data variables to select suitable initial conditions. The two methods are compared for the full boiler model network. A two layer network, with 13 tan-sigmoid neurons in the first layer and 11 linear neurons in the second layer is used. The results, presented in Table. 6.1 show that better results are obtained using a symmetric random number generator.

Weight Initialisation	Drum Level Validation Error	Pressure Validation Error
Random	7.155e7	6.337e-3
Nyguen Widrow	1.095e8	1.562e-2

Table 6.1 Comparison of Weight Initialisation Methods

6.4.2.4 Training Period

Each iteration of the training algorithm generates new weights and biases, thus modifying the nonlinear relationship between the neural network inputs and outputs. Weight adaptation continues until the training algorithm can not find a new set of weights which reduces the training error criterion - further training after this point will not modify the weights or reduce the training error. It seems reasonable to select this final set of weights as the "best" choice for the implemented network. However, it is important to confirm that the validation error criterion continues to decrease during training. (The training error criterion, which is used to adapt the weights, is the sum squared error over the training set for a 1-step ahead prediction. The validation error criterion is the mean squared error over the validation set for a 10-step ahead prediction). It is possible that training may produce networks which have a continuously decreasing training error criterion but with an increasing or erratic validation error criterion. For example, it is possible to over-train a network, so that it learns the training data exactly, but is not capable of generalising to the validation set.

In all the investigated cases, there *is* a direct correlation between the training error criterion and the validation error criterion. A plot of training error *versus* training iterations and of validation error *versus* training iteration for main steam pressure, (Fig. 6.9) demonstrates the correlation - the validation error criterion decreases in parallel with the training error criterion.

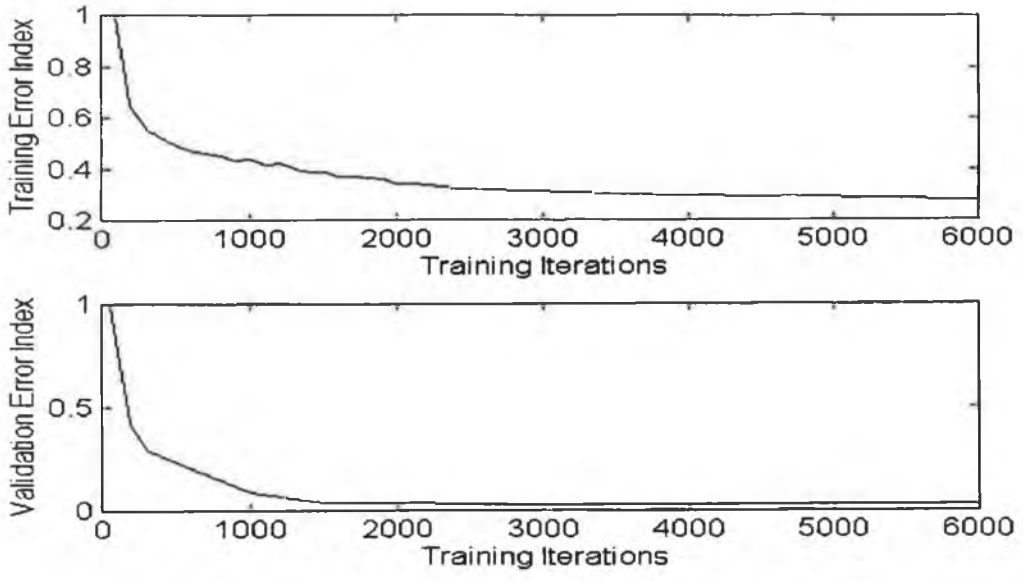


Fig. 6.9. Comparison of Validation Error Index *versus* Training Error Index

Given the correlation between validation error and training error, it is acceptable to continue training until the sum squared error over the training set has stopped decreasing. The number of iterations required to reach this point depends on the size of the network and the training data set. In order to perform a fair comparison between various model structures, the same number of training iterations are carried out for each network structure. For the full boiler model, 45,000 training iterations is the number chosen as in all cases the sum squared error has stabilised before this number of iterations. For the superheater model and the gas composition model, 30,000 training iterations are sufficient.

6.4.3 Neural Network Structure

6.4.3.5 Data Pre-processing

Data pre-processing is used to assist network training. It is performed both on the network input data and output data.

Input Data Pre-processing

- **Scaling**

  The network input structure uses measured boiler variables of widely varying range. For example, fuel mass flow rate may vary between 0 kg/s and 1 kg/s. Superheated steam pressure may vary between 4e6 Pa and 5e6 Pa. This input data must be scaled to prevent the large variations in superheated steam pressure dominating the smaller variations in fuel mass flow rate.

  To aid training, the range of the scaled input data should equal the input range of the first layer of neurons so that the full output range of the nonlinearity can be utilised. For example, for tan-sigmoid neurons which have an input range of -[1,1], the input data is scaled to a range of [-1,1]. For log-sigmoid neurons, which have an input range of [0,1], the input data should be scaled to between [0,1].

Output Data Pre-processing

- **Scaling**

  There are no hard and fast rules for target data scaling. It is sufficient that each target variable is scaled to the same order of magnitude, to prevent variations in any particular target variable dominating training.

- **Data Differencing**

  The neural network is to be trained to perform a single step ahead prediction of a dynamical

system. As regards training such a network, it is not advisable to simply provide the previous value of a given variable as a network input and the current value of that variable as a desired network output. If the variable changes slowly this is likely to result in a network which "learns" to pass the previous value of the variable to the network output where it is presented as the current value of the variable. This problem is overcome by differencing the network output data, so that the network learns to predict the change in a variable. The predicted change can then be added to the previous value of the variable to yield the current value of that variable.

6.4.3.6 Network Input Structure

The network input structure is of fundamental importance to the neural network performance - the network must be provided with all the information needed to predict the future boiler behaviour. The structure should also be parsimonious, however, to simplify network training.

Model of Main Steam Pressure and Drum Level

The first-principles boiler model developed in Chapter 4 has twenty states and six inputs. Theoretically, this implies that the neural network of the boiler should be provided with the current measurement of each of the twenty state variables and six inputs. In practice, it is not necessarily correct to assume that this input structure will result in the best network performance. Two types of modifications to this theoretical input structure are investigated:

1. A number of the boiler state variables, have limited effect on the response of the controlled variables. Economiser gas temperature is an example of one such variable. Removing less important variables from the input structure simplifies training, and should result in a better network. Each of the state variables of the first-principles boiler model was considered and the following variables were highlighted as being of minor importance.

   ◊ Economiser: Gas Temperature, Metal Temperature, Fluid Temperature

   The economiser preheats the feedwater entering the drum, using the heat which is remaining in the combustion gases before they leave the boiler. It is essentially an energy recovery unit and does not have a significant effect on the response of the controlled variables.

   ◊ Riser: Gas Temperature - Pass 1 and 2, Metal Temperatures, Pass 1 and 2, Superheater: Gas Temperature

   These state variables are not included because their dynamics are much faster than the

dynamics of the controlled variables - drum level and main steam pressure.

◊ Superheater: Steam Pressure and Temperature

These state variables are closely related to main steam temperature and pressure - main steam pressure and temperature are equal to lagged superheater steam pressure and temperature, respectively, as no heat is added to the steam after it leaves the superheater. Consequently, no significant additional information is provided to the neural network by the inclusion of these variables.

One of the controlled variables - main steam pressure - is already included in the remaining ten state variables. Drum level is also added to the input structure, in order to provide the network with an initial condition for the plant drum level. The network must predict the one-step-ahead change in the ten states variables and drum level so that the network inputs can be updated at the next prediction step.

2. A second possible modification to the theoretical input structure is the inclusion of previous boiler measurements. This is not necessary for the first-principles boiler model, however, it may provide additional information to the neural network about the dynamics of the first-principles model.

The validity of both suggested modifications is assessed by a comparison of different network structures. Five different types of input structures are compared. A two layer network with 13 tan-sigmoid neurons in the hidden layer and 11 linear neurons in the output layer was used for the comparison. The results of the comparison are summarised in Table 6.2

Test	Input Structure	Pressure Validation Error	Drum Level Validation Error
1.	6 inputs, 20 states, drum level measured at *t*.	9.333e10	12.888
2.	6 inputs, 20 states, drum level measured at *t, t-1*	1.728e8	1.569e-1
3.	6 inputs, 10 states, drum level measured at *t*	8.889e7	1.778e-1
4.	6 inputs, 10 states, drum level measured at *t, t*-1.	7.155e7	6.337e-3
5.	6 inputs, 10 states, drum level measured at *t.. t*-2	1.370e8	1.873e-1

Table. 6.2 Comparison of Input Structures

The first two tests compare networks which use the full set of state variables in their input structures. For Test 1, only the current values of the full set of state variables and inputs are provided. For Test 2, the current and previous values of the same variables are supplied. A significantly better performance is achieved in Test 2.

The last three tests compare networks which use the reduced set of state variables in their input structures. For Test 3, the current values only of the state variables and inputs are provided. For Test 4, the current and previous values of the same variables are used. For Test 5, the current and two previous values of the same variables are used. In all cases, the results are better than for Test 2. The best performance is achieved in Test 4, where the current and previous value of each of the inputs is applied.

These tests confirms that the inclusion of additional "minor" variables can have a negative affect on network performance. The best of the full-state networks (Test 2) produces a validation error which is more than twice the magnitude of the validation error for the comparable reduced-state network (Test 4).

The results show that inclusion of lagged input variables can improve network performance dramatically. A network which uses only current values of all the input and state variables (Test 3) has a drum level validation error twenty times greater than the validation error for a network which uses current *and* previous values of the same variables (Test 4). However, the results shows that the inclusion of additional lagged input variables may not necessarily enhance network performance. A network which uses three lagged input and output variables (Test 5) has a validation error approximately twice that of the network which uses two lagged input and output variables (Test 4). The results indicate that any benefit which might be gained by the inclusion of extra input variable is offset by a deterioration in network training.

The final choice of network input and output structure is:

NN INPUT VARIABLES		**NN OUTPUT VARIABLES**
Boiler Inputs	**State and Output Variables**	
Current and Previous: 1. Feedwater flow 2. Fuel flow 3. Attemperating flow 4. Air flow 5. Feedwater temperature 6. Load demand	**Current and Previous:** 1. Drum pressure 2. Drum water volume 3. Drum water temperature 4. Drum steam temperature 5. Downcomer water temperature 6. Steam quality 7. Superheater metal temperature 8. Main steam pressure 9. Main steam temperature 10. Riser fluid temperature 11. Drum level	1. Δ Drum pressure 2. Δ Drum water volume 3. Δ Drum water temperature 4. Δ Drum steam temperature 5. Δ Downcomer water temperature 6. Δ Steam quality 7. Δ Superheater metal temperature 8. Δ Main steam pressure 9. Δ Main steam temperature 10. Δ Riser fluid temperature 11. Δ Drum level

Table 6.3 Boiler Model Input and Output Variables

inue
r of
set.
r of
000
fore
000

nput

For
team
t the
mass

first
nple,
to a
data

arget
arget

mical

Model of Main Steam Temperature

All of the input variables and state variables for the superheater model are included in the input structure, there being no variables of obviously lesser significance. The current and previous values of each of these variables is included in the input structure, as this had been demonstrated to generate the best performance for the neural network model of the full boiler. The resulting list of input and output variables for the superheater model is:

NN INPUT VARIABLES		NN OUTPUT VARIABLES
Boiler Inputs	**State and Output Variables**	
Current and Previous: 1. Fuel flow rate 2. Attemperating flow rate 3. Air flow rate 4. Load demand 5. Drum pressure 6. Drum steam temperature	**Current and Previous:** 1. Superheater Gas Temperature 2. Superheater Metal Temperature 3. Superheater Steam Pressure 4. Superheater Steam Temperature 5. Main Steam Temperature 6. Main Steam Pressure	1. Δ Superheater Gas Temperature 2. Δ Superheater Metal Temperature 3. Δ Superheater Steam Pressure 4. Δ Superheater Steam Temperature 5. Δ Main Steam Temperature 6. Δ Main Steam Pressure

Table 6.4 Superheater Model Input and Output Variables

Model of Percentage O_2 in Stack Gases

As for the superheater model, all of the current and previous values of input and state variables of the first-principles gas composition model (excluding pure delay) are included in the input structure. The input and output structure is given in Table 6.5.

NN INPUT VARIABLES		NN OUTPUT VARIABLES
Boiler Inputs	**State and Output Variables**	
Current and Previous: 1. Fuel flow rate 2. Air flow rate	**Current and Previous:** 1. Percentage O_2 in stack gases, undelayed	1. Δ Percentage O_2 in stack gases, undelayed

Table 6.5 Gas Composition Model Input and Output Variables

6.4.3.7 Type of Nonlinearities

The neural network nonlinear characteristics are created by the nonlinear neurons in the network hidden layers. Two types of nonlinearities are compared - tan-sigmoid and log-sigmoid neurons. The tan-sigmoid function generates outputs between *-1* and *1* as the neuron's net input goes from negative to positive infinity. The log-sigmoid function generates outputs between *0* and *1* as the neuron's net input goes from negative to positive infinity. For the full boiler model, two networks of equal size but using different nonlinearities, are compared over several training runs to

determine which type of nonlinearity is must suitable for this task. In all cases, the network with the tan-sigmoid neurons achieve a much smaller prediction error over the validation set. Bi-polar tan-sigmoid neurons are used in all subsequent comparisons. The results of the comparisons are summarised in Table 6.6.

Full Boiler Model			
Network Structure	**Nonlinearity**	**Pressure Validation Error**	**Drum Level Validation Error**
2 Layers, 13 - 11	Tan-sigmoid	7.155e7	6.337e-3
	Log-sigmoid	1.946e8	5.318e-2

Table. 6.6 Comparison of Nonlinearities for Full Boiler Model

6.4.3.8 Network Size

The network size is specified by the number of hidden layers and by the number of neurons in each layer. The best combination of layers and neurons is determined by trial and error. A comparisons of model performance with respect to network size is carried out for each of the models. The results of the comparison for the full boiler model are summarised in Table 6.7.

Full Boiler Model			
Network Structure	**Number of Weights**	**Pressure Validation Error**	**Drum Level Validation Error**
3 Layers: 1 - 3 - 11	70	2.250e8	5.828e-1
3 Layers: 3 - 9 -11	228	4.182e8	1.612e-1
3 Layers: 5 -15 -11	410	1.952e8	4.150e-2
2 Layers: 10 - 11	450	8.491e7	8.379e-3
2 Layers: 13 -11	585	7.155e7	6.337e-3
2 Layers: 15 - 11	675	8.345e7	6.737e-3
2 Layers: 17 - 11	765	7.400e7	9.626e-3
2 Layers: 20-11	900	7.136e7	2.208e-2
2 Layers: 23 - 11	1035	6.266e7	8.198e-3
2 Layers: 27-11	1216	6.000e7	1.080e-2

Table. 6.7 Network performance *versus* network size for full boiler model

In all cases, the two-layer networks perform better than the three-layer networks. A closer examination of the results for the two layer networks reveals no obvious relationship between network size and network performance. In general, the pressure validation error decreases as network size increases. However, again with exceptions, the converse may be stated for the drum level validation error. One good compromise choice for network size is 2 layers with 13 neurons in the first layer and 11 neurons in the output layer. This network scores fourth as regards pressure validation error and first as regards drum level validation error. In addition, it is a relatively small network and thus has a low computational overhead.

The same comparison is carried out for the superheater model

Superheater Model		
Network Structure	**Number of Weights**	**Temperature Validation Error**
3 Layers: 1 - 3 - 6	47	3.0114
3 Layers: 3 - 6 - 6	132	2.3651
2 Layers: 5 - 6	160	2.0921
2 Layers: 10 - 6	320	1.5468
2 Layers: 15 - 6	480	1.2777

Table. 6.8 Network performance *versus* network size for superheater model

The best performance is achieved by a two layer network, with 5 neurons in the hidden layer. The comparison results for the gas composition model are:

Gas Composition Model		
Network Structure	**Number of Weights**	**O_2 Validation Error**
3 Layers: 1 - 3 - 1	12	2.512e-3
3 Layers: 3 - 6 - 1	42	4.567e-4
3 Layers: 5 - 10 - 1	90	3.589e-4
2 Layers: 5 - 1	35	4.295e-4
2 Layers: 10 - 1	70	4.299e-4
2 Layers: 15 - 1	105	4.287e-4

Table. 6.9 Network performance *versus* network size for gas composition model

The best performance has been achieved by a 3 layer network, with 5 neurons in the first hidden layer and 10 neurons in the second hidden layer.

6.5 Prediction Using Neural Network Model

The same procedure is used by both the neural network superheater and full boiler model to perform a single- or multi-step ahead prediction The procedure for a single-step ahead prediction is shown schematically in Fig. 6.10. At the start of the prediction, the neural network is provided with the measured values of the current and previous boiler states and boiler inputs. Using this data, the neural network predicts the change in each of the state variables during the current sampling period. The predicted value of each of the state variables at the next sampling interval is found by adding the predicted change to the current value of the state variable.

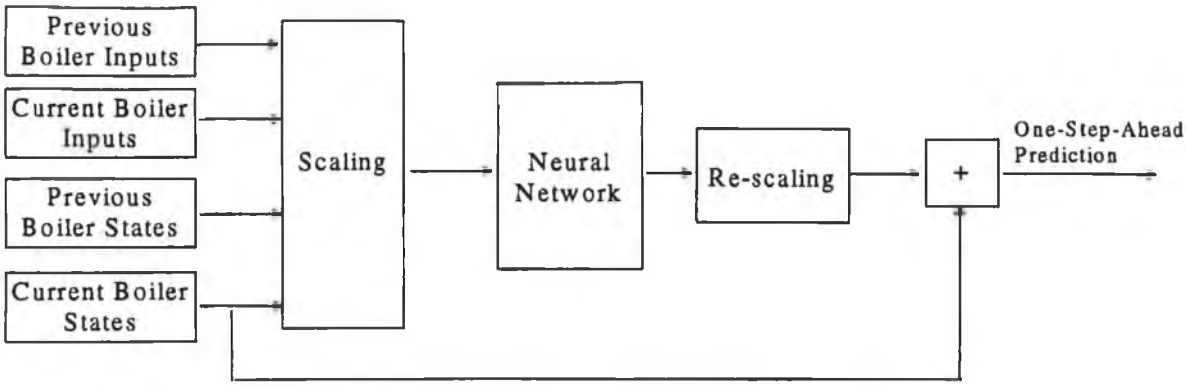


Fig. 6.10 Single-Step Prediction

The procedure for a multi-step prediction is similar in outline to that for a single step ahead prediction. At each prediction step the current output of the neural network is used to update the state variables in the neural network input structure. The procedure for updating the neural network input variables is shown in Fig. 6.11

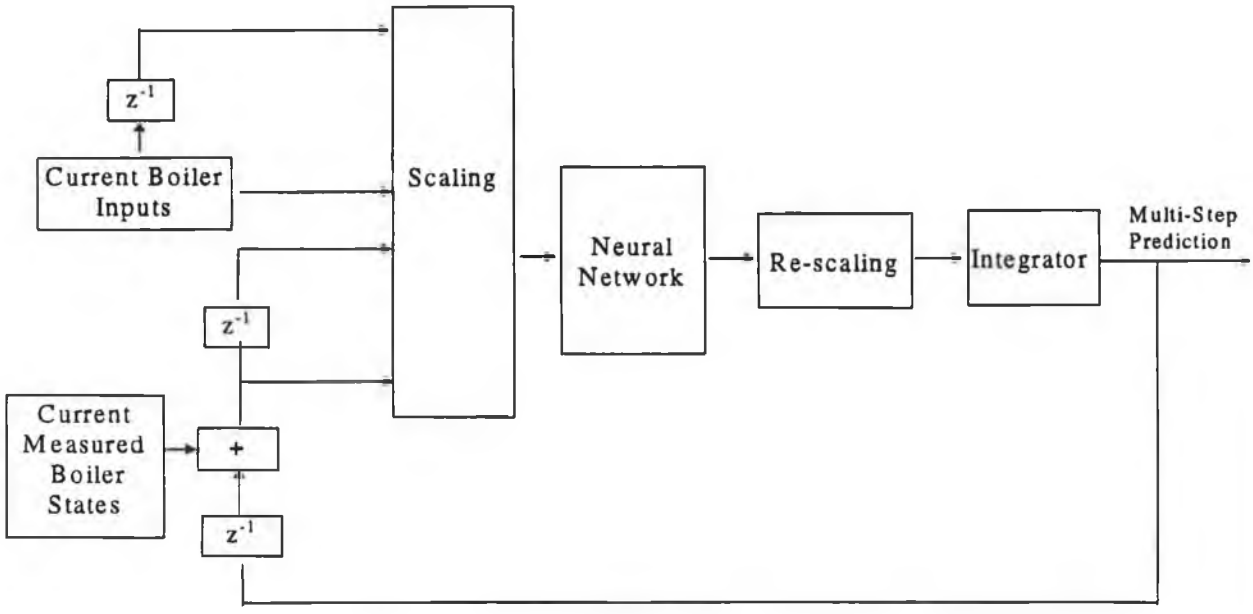


Fig. 6.11 Multi-Step Prediction

The procedure for performing a prediction is marginally different for the neural network gas composition model. One of the input variables of this model - percentage O_2 in the stack gases, undelayed - is not a physically measurable variable. It is simply a prediction of the measured value of percentage O_2 in the stack gases, where the length of the prediction horizon is equal to the length of the delay. It follows that it is not possible to initialise the neural network at the start of the prediction with a measured value of this variable. Instead, it is necessary to initialise the neural network model input with the single-step-ahead prediction of this variable which was

calculated during the last sampling interval. The procedure is presented in Fig. 6.12 for a single-step-ahead prediction.

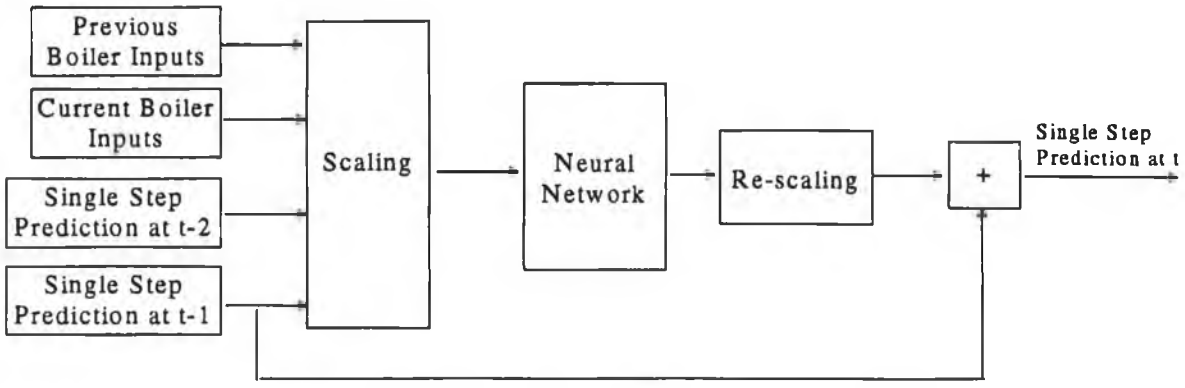


Fig. 6.12 Single Step Prediction Using Gas Composition Model

A multi-step prediction is also initialised in this manner, as shown in Fig. 6.13.

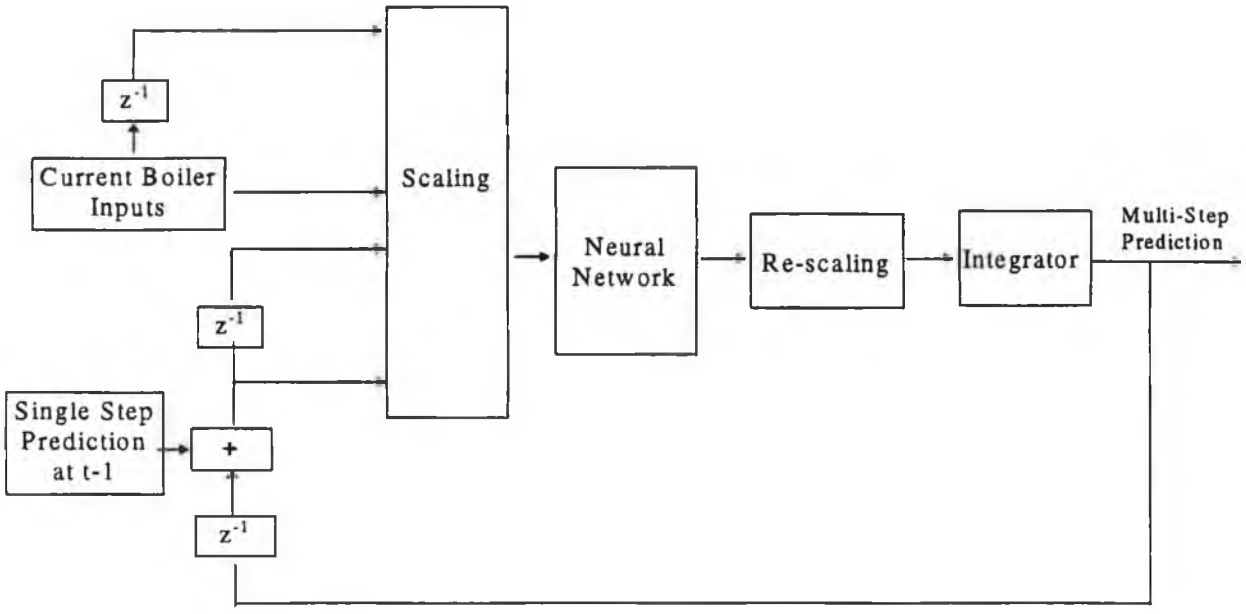


Fig. 6.13 Multi-Step Prediction for Gas Composition Model

6.6 Results

The ability of the neural network models to perform a multi-step ahead prediction of the first-principles model output is assessed by using the neural network models to perform a 10-step prediction of the first-principles model output. This predicted output is compared to the output generated by the first-principles model, which is of course advanced by 10 sampling periods. The network is initialised with the actual first-principles model inputs at the throughout the prediction but is initialised with the actual first-principles model states at the start of the prediction only.

A comparison is performed for each of the controlled outputs at two different operating points - 30% and 90%.

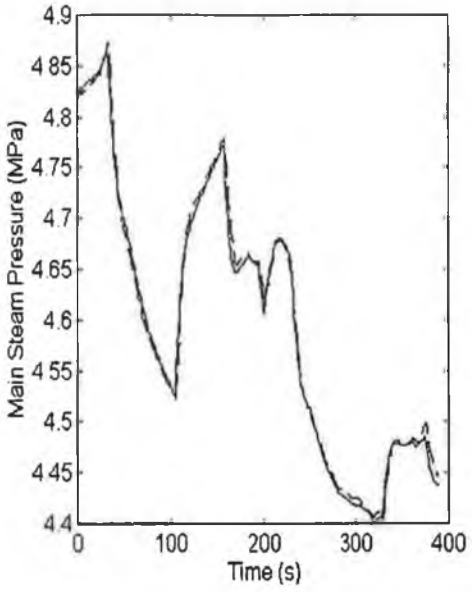


Fig. 6.14a Comparison of Neural Network Prediction of Main Steam Pressure *versus* First-Principles Boiler Model at 30% Load

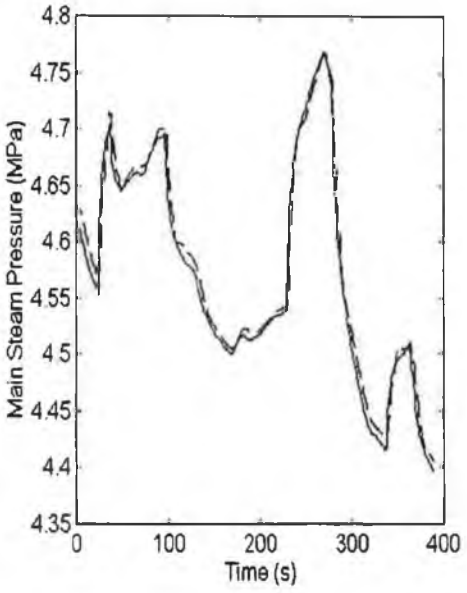


Fig. 6.14b Comparison of Neural Network Prediction of Main Steam Pressure *versus* First-Principles Boiler Model at 90% Load

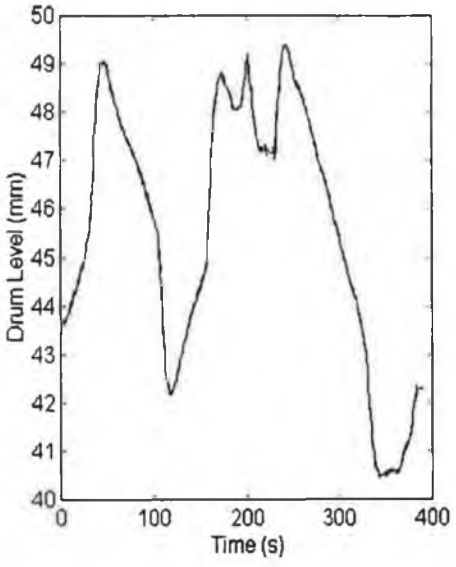


Fig. 6.15a Comparison of Neural Network Prediction of Drum Level *versus* First-Principles Boiler Model at 30% Load

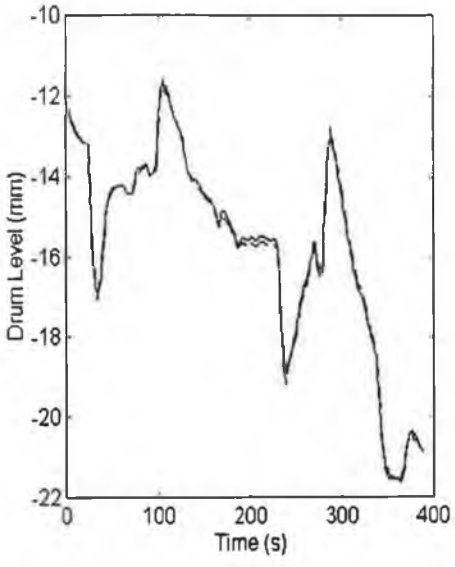


Fig. 6.15b Comparison of Neural Network Prediction of Drum Level *versus* First-Principles Boiler Model at 90% Load

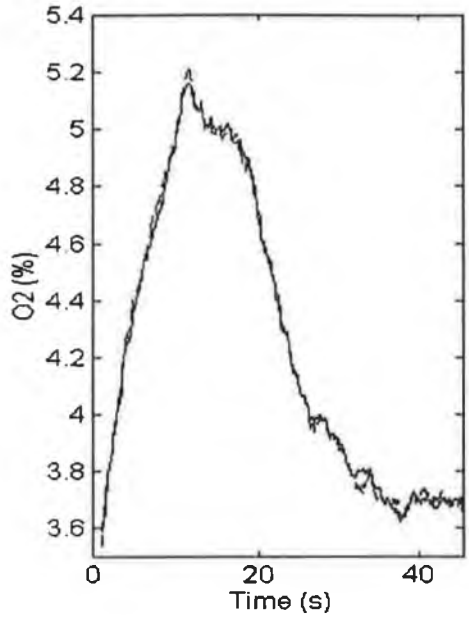


Fig. 6.12a Comparison of Neural Network Prediction of %O2 *versus* First-Principles Boiler Model at 30% Load

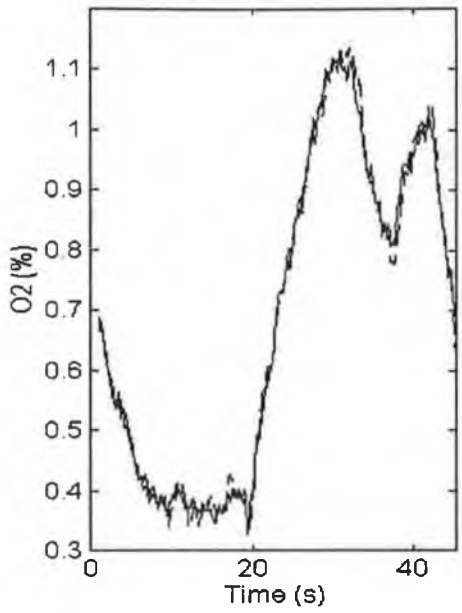


Fig. 6.12b Comparison of Neural Network Prediction of %O2 First-Principles Boiler Model at 90% Load

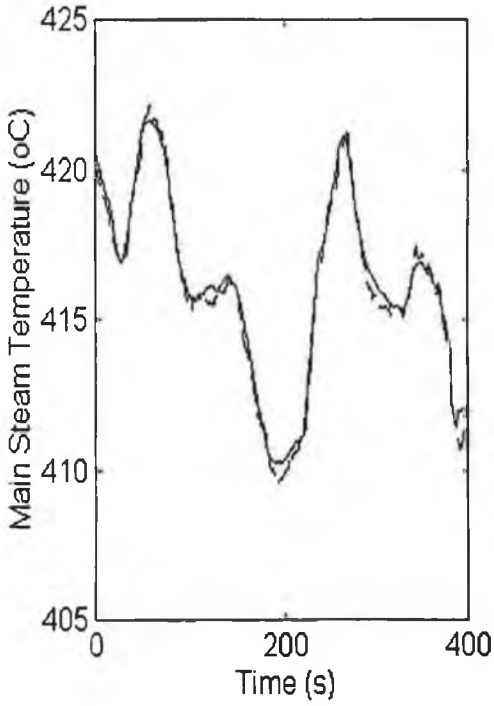


Fig. 6.13a Comparison of Neural Network Prediction of Main Steam Temperature *versus* First-Principles Boiler Model at 30% Load

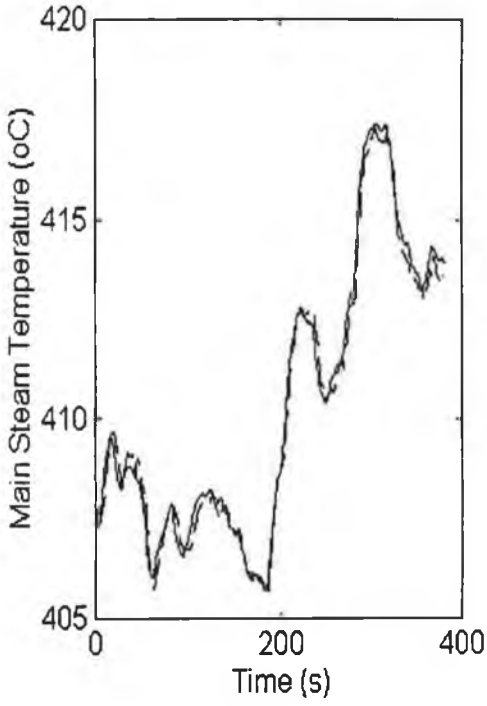


Fig. 6.13b Comparison of Neural Network Prediction of Main Steam Temperature *versus* First-Principles Boiler Model at 90% Load

It can be seen that the neural network of capable of predicting all four variables with a high degree of accuracy.

6.7 Conclusions

Neural network models have immense scope. It has been shown that they may be use to model any arbitrary nonlinear function (Narenda and Parthasarathy (1990)). In theory, neural network models can be generated quickly and do not require detailed knowledge of the system processes. However, a certain amount of system knowledge is necessary to construct a neural network e.g. which inputs are dynamical, an approximate estimate of the order of the system and the maximum system delay. A good understanding of the system is also vital in order to generate a training set which adequately covers the complete operating range of the plant.

Neural networks do not always train consistently. There are a large number of factors which affect network convergence e.g. initial weights used, size of adaptation gain, scaling of inputs, etc. It is impossible to test all possible combinations of these factors in order to derive the best possible model. In practice, this is not a significant problem, because there are many combinations of these factors which will generate a good neural network model.

The results show that neural networks are capable of modelling a boiler over its entire operation range. This was a significant challenge on account of the nonlinearity of the plant and its time varying dynamics. In addition, neural network models perform this task in a computationally efficient manner. This combination of performance and computational efficiency renders neural network models feasible as internal controller models.

7. Theory of Predictive Control

7.1 Introduction

Commercial viability demands that industry meet a growing number of stringent and often conflicting objectives. A boiler plant must optimise economic efficiency and plant life-time, while providing a quick response to varying steam demands. Typically, plant optimisation means operating the plant at the constraints. Good control reduces the variability of the controlled variables and allows the plant to be safely operated closer to the constraints. This is the objective of all control methodologies. However, better control could be achieved if the controller were provided with information about the plant constraints, thus allowing the controller to automatically select the best controller action which satisfies all the plant constraints. The controller may then operate the plant on the constraints, in order to optimise plant performance. This is one of the major strengths of predictive control. Predictive control is the only design methodology which provides a systematic means of handling constraints.

The second strength of predictive control is flexibility. Unlike many other model based control strategies, predictive control may use any form of plant model. The choice of model is dictated solely by the characteristics of the process and the controller performance criteria. Flexibility is also provided through the cost function, which can be adapted to suit the problem at hand.

There is a growing body of stability results for linear predictive control. This theory is also being extended to nonlinear plants. A comprehensive robustness theory has yet to emerge. However, predictive controllers can be adjusted for robustness more easily than classical feedback controllers (García *et al* (1989))

Predictive control has been demonstrated successfully on hundreds of diverse industrial applications e.g. distillation column (Richalet (1993b)), robot arm (Clarke (1988)), transonic wind tunnel (Soeterboek *et al* (1991)), anaesthesia (Linkens and Mahfouf (1994)).

The flexibility of predictive control can be assessed from its historical background. It emerged independently from three diverse sources:

- The first branch of predictive control appeared in the late 1970's and was motivated by industrial applications. Richalet *et al* (1978) were the first to use a long-range prediction horizon in their Model Algorithmic Control (MAC). This was quickly followed by Dynamic Matrix Control (DMC), developed by Cutler and Ramaker (1980). Both approaches use an explicit dynamic model of the plant. MAC uses a finite impulse response (FIR) model and

DMC uses a finite step response (FSR) model. Both approaches consider constraints on the plant and obtain numerical solutions. This type of control is usually referred to as Model Predictive Control (MPC).

- The second branch of control arose from the field of self-tuning control. It was first seen as Minimum Variance Control (MV) developed by Åström and Wittenmark (1973) and then as Generalised Minimum Variance (GMV) control, developed by Clarke and Gawthrop (1979). These algorithms were applied in self-tuning control strategies. They used short-range predictions and demonstrated poor robustness. A second generation of controllers which used long range predictions demonstrated much improved robustness. These include Extended Horizon Adaptive Control (EHAC) developed by Ydstie (1984) and Generalised Predictive Control developed by Clarke *et al* (1987a, 1987b). These controllers generally use SISO I/O models such as ARMAX (Auto-Regressive Moving Average Model with Exogenous Inputs) or ARIMAX (Integrated Auto-Regressive Moving Average Model with Exogenous Inputs) models which are in general use in self-tuning or adaptive algorithms. Controllers of this type will be referred to as Generalised Predictive Control (GPC).

- The third branch of predictive control developed as a variation of Linear Quadratic (LQ) or Linear Quadratic Gaussian (LQG) control. It has applied receding-horizon state-space theories to the problem of guaranteed stability. Controllers of this type use a terminal constraint or end-point weighting to achieve guaranteed stability under stated conditions. A stabilising receding horizon control with fixed terminal constraints was suggested by Kwon and Pearson (1975). Constrained Receding Horizon Predictive Control (CRHPC) was developed by Clarke and Scattolini (1991). This type of control will be referred to as Receding Horizon Control (RHC).

The *linear* predictive control methodology described in Chapter 8, termed Predictive Functional Control (PFC), was developed by Richalet *et al* (1978). It bears some similarities to GPC in that it is unconstrained and has an analytical solution. However, it also employs many of the useful techniques which were developed for MAC.

7.2 Predictive Control Strategy

Predictive controllers control the plant by specifying the desired plant output at a particular instance or instances in the future and then calculating the controller action which minimises the predicted error. The solution to the cost function minimisation problem specifies the controller output for each sampling period up until the last of the specified future time points. However, only the first set of controller outputs is actually applied. At the next sampling period, a new

controller solution is calculated, which takes into account any changes such as variations in the measured process output or setpoint. This receding horizon approach is illustrated for a three-step horizon in Fig. 7.1

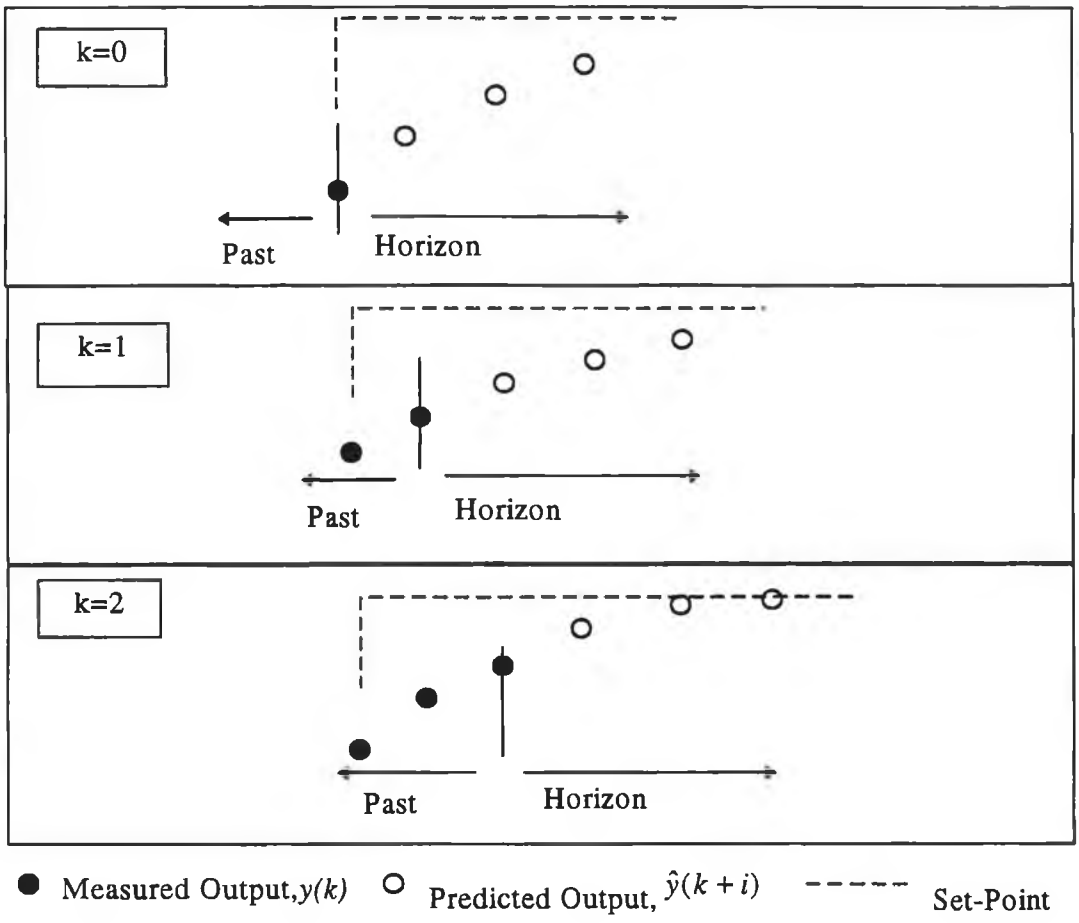


Fig. 7.1 Predictive Control - Receding Horizon Approach

The particulars of implementation differ for each variety of predictive control. However, all types of predictive control are composed of three basic components:

1. Internal Controller Model
2. Cost Function
3. Controller Solution

7.2.1 Internal Controller Models

In theory, the internal model used by a predictive controller may be of any type. The only absolute requirement being that the model is able to predict the plant behaviour with some degree of accuracy. In practice, the type of model used is dictated by the type of plant and by the controller specification. For example, fast processes require computationally efficient models with a small number of parameters; multivariable controllers require multivariable models.

The theory of predictive control has been developed largely for linear controllers which use linear models. Two commonly used linear models are described here.

Input/Outputs Models

The following SISO ARMAX model may be used to represent the process:

$$\hat{y}(k) = \frac{B(q^{-1})}{A(q^{-1})}u(k) + \frac{T(q^{-1})}{A(q^{-1})}\xi(k)$$

where $$A(q^{-1}) = \sum_{j=0}^{na} a_j q^{-j},\ a_0 = 1,\ B(q^{-1}) = \sum_{j=1}^{nb} b_j q^{-j} \qquad (7.1)$$

$$T(q^{-1}) = \sum_{j=1}^{nt} t_j q^{-j}$$

The model includes a disturbance polynomial $T(q^{-1})$, which has an important role to play in improving robustness.

The future plant output can be computed recursively using (7.1). This is computationally intensive - i computations are required for an i-step ahead predictor. The prediction can be calculated in a single step if the first and second terms on the right hand side of equation (7.1) are split into their individual past and future terms using the following Diophantine equations (for brevity, the q^{-1} dependence is not included)

$$\frac{B}{A} = G + \frac{H}{A} \qquad (7.2)$$

$$\frac{T}{A} = E + \frac{F}{A} \qquad (7.3)$$

Using the above identities, equation (7.1) is rearranged so that the prediction can be calculated in a single step:

$$\hat{y}(k+i) = Gu(k+i-1) + \frac{H}{A}u(k-1) + \frac{F}{T}[y(k) - \hat{y}(k)] + E\xi(k+i) \qquad (7.4)$$

The first and second terms of equation (7.4) are the predicted process output as calculated by the model. The third term corrects the prediction for modelling errors. It filters modelling errors in a way which can be defined by the disturbance polynomial $T(q^{-1})$. The fourth term is the prediction error due to future noise only. This term can be eliminated if the noise is assumed to be white noise with zero mean

Equation (7.4) can be viewed in an alternative manner for control law derivation. The first term is the response of the system to the current controller action and is described as the forced response term. The second and third terms is the response of the system to previous controller actions and is termed the free response term.

In GPC, an CARIMA (Controlled Auto-Regressive Integrated Moving Average Model) model is often used in preference to an ARMAX model.

$$\hat{y}(k) = \frac{B(q^{-1})}{A(q^{-1})}u(k) + \frac{T(q^{-1})}{\Delta A(q^{-1})}\xi(k) \qquad (7.5)$$

where $\Delta = (1 - q^{-1})$

The predictor output derived from the CARIMA model includes a term equal to the integral of previous modelling errors. If the modelling errors are constant (as is the case for constant disturbances and reference trajectories) the integrator output will increase or decrease until the predicted model output equals the true plant output. If the controller succeeds in placing the predictor output at the setpoint, it follows that the plant output will also be at the setpoint. In summary, a GPC controller using an CARIMA model can eliminate steady-state errors in the case of constant disturbances and constant reference trajectories.

The ARMAX model is used in EHAC and the ARIMAX or CARIMA model are used in GPC. Both models are computationally efficient and are suitable for use in adaptive controllers. Theoretically, they can be extended for use in MIMO controllers, but in practice this is not a straightforward task.

FIR/FSR models are obtained from equations (7.1) and (7.5) respectively by setting $A(q^{-1})$ equal to *1*. FIR/FSR models have the advantage of simplicity, but they can only be used to model stable processes without integrators. They are slow to compute as they have a large number of parameters They are suitable for modelling open-loop stable, slow processes.

State Space Models

The following discrete state space model could also be used to represent the process:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Gz(k) \\ \hat{y}(k) &= Cx(k) + v(k) \end{aligned} \qquad (7.6)$$

All modelling uncertainties and all deterministic disturbance signals are assumed to be combined into the system noise signal *z(k)*. The output is also assumed to be disturbed by measurement noise *v(k)*.

The output of the state space model after i steps is:

$$\hat{y}(k+i) = CA^i x(k) + \sum_{j=0}^{i-1} CA^{i-j-1} Bu(k+j) + \sum_{j=0}^{i-1} CA^{i-j-1} Gz(k+j) + v(k+i) \tag{7.7}$$

State space models are computationally efficient and suitable for modelling many types of process. They are inherently multivariable and very suitable for implementing MIMO controllers.

7.2.2 Cost Function

7.2.2.1 Statement of Cost Function

The cost function specifies the performance criteria which the controller should aim to achieve. Typically, it penalises the error between the process output and set-point as well as controller action. The position of the closed loop poles can be varied *via* the cost function definition. A typical quadratic cost function which takes into account tracking error and controller action is

$$J = \sum_{j=N_1}^{N_2} [\hat{y}(k+j) - w(k+j)] Q [\hat{y}(k+j) - w(k+j)]^T + \sum_{j=1}^{N_u} u(k+j-1) R u(k+j-1)^T$$

where

$\hat{y}(k+j)$ = predicted plant output
$w(k)$ = desired reference trajectory
N_1 = minimum prediction horizon
N_2 = maximum prediction horizon
N_u = control horizon
Q, R = weighting matrices

(7.8)

Quadratic or 2-norm criteria have the advantage that an analytical control law can be found for the unconstrained case. A disadvantage of the 2-norm criteria is that storage requirements can be considerable for a MIMO controller with a long prediction horizon. If a more compact controller is required the objective can be expressed using the 1-norm formulation and the control law calculated using Linear Programming (Taha,. (1987). The disadvantage of the 1-norm formulation is that it is not possible to derive an analytical, closed-form control law.

The ∞-norm of the error between the predicted constrained and the ideal unconstrained process output is used by Campo and Morari (1986). The ∞-norm has the effect of reducing peak excursions, whereas the 1- and 2-norms reduce the average deviation.

The cost function may be optimised subject to input and output constraints such as actuator position or rate limits or to state and output constraints such as the maximum operating temperatures in a furnace. Such constraints can be formulated as:

$$
\begin{aligned}
l_l &\le \hat{u}(k+i) \le l_u \\
n_l &\le \Delta\hat{u}(k+i) \le n_u \\
m_l &\le \hat{x}(k+i) \le m_u
\end{aligned}
$$

where

$$
\begin{aligned}
l_l(k) &= \text{lower actuator position constraint vector} \\
l_u(k) &= \text{upper actuator position constraint vector} \\
n_l(k) &= \text{lower actuator rate constraint vector} \\
n_u(k) &= \text{upper actuator rate constraint vector} \\
m_l(k) &= \text{lower state constraint vector} \\
m_u(k) &= \text{upper state constraint vector}
\end{aligned}
\qquad (7.9)
$$

7.2.2.2 Controller Tuning

Controller tuning may be affected through the cost-function definition or by augmenting the basic controller. There is considerable flexibility as to how the required controller performance may be specified. As a result, considerable diversity exists between the various predictive methodologies with respect to controller tuning. Table 7.1 lists some of the more important controller tuning parameters or controller tuning mechanisms. The first two parameters in the table, - minimum prediction horizon, N_1 and maximum prediction horizon, N_2 are common to all control strategies. The remaining parameters are placed on the same row as an equivalent tuning parameter. For example, the tuning parameter - control horizon, N_u, is considered to be equivalent to the tuning mechanism - control output structuring. This does not imply that these parameters are exactly equivalent, rather that these parameters are capable of having a similar effect on controller performance.

Controller Tuning Parameter	Alternative Tuning Parameter
1. Minimum Prediction Horizon, N_1	-
2. Maximum Prediction Horizon, N_2	-
3a. Control Horizon, N_u	3b. Control Output Structuring
4a. Output Error Weighting Matrix, Q	4b. Set-Point Pre-Filter
5a. Control Signal Output Weighting Matrix, R	5b. Controller Output Filter

Table 7.1 Predictive Control Tuning Parameters

1. Minimum Prediction Horizon - N_1,

The predictive controller minimises the error between the desired and the predicted plant output over a finite or infinite time span known as the prediction horizon. GPC includes every sampling instance between the minimum prediction horizon and the maximum prediction horizon. PFC reduces computational effort by using a subset of points over the prediction horizon known as coincidence points. The difference between these two approaches is illustrated in Fig. 7.2.

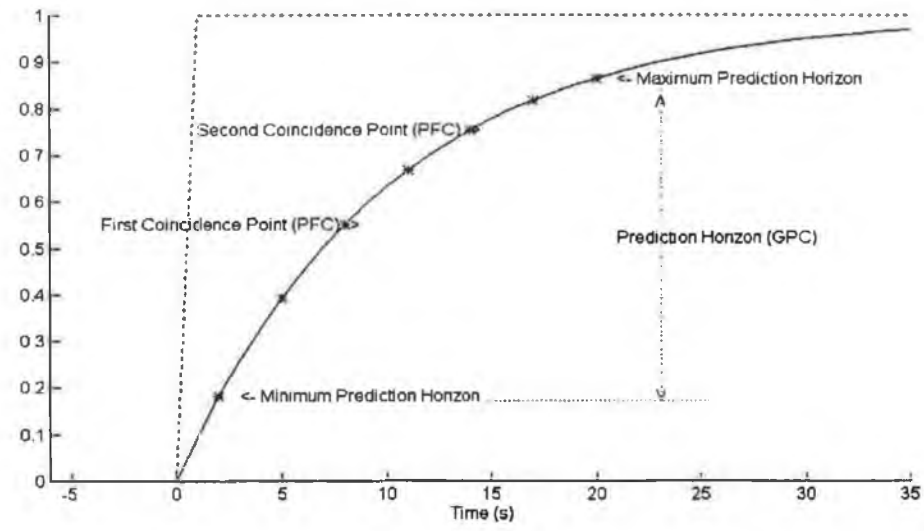


Fig. 7.2 Comparison of GPC and PFC Prediction Horizon

The GPC and PFC prediction horizon are exactly equivalent if the set of coincidence points chosen for the GPC controller includes every sampling instance on the prediction horizon between the minimum and maximum prediction horizon chosen by the PFC controller. In either case, the same flexibility is available with regard to controller tuning. The minimum prediction horizon specified by GPC is equivalent to the minimum coincidence point as specified by PFC.

The minimum prediction horizon, N_1, is generally chosen equal to $d+1$, where d is the system time delay index, such that

$$d\Delta T = \tau$$

where

d = time delay index (7.10)

ΔT = sampling period

τ = pure time delay

If the minimum prediction horizon of less than $d+1$ is chosen, tracking errors which cannot be influenced by control action (due to the existence of pure time delay in the system) are included in the cost function.

If N_1 is increased beyond $d+1$, the early tracking errors are not included in the cost function. In a minimum-phase system, this results in smoother, slower and more robust control. In a non-minimum phase system, the opposite can occur. If N_1 is increased to the point, where errors that occur during the transient nonminimum phase response are not included in the cost function, control action will be similar to that for a minimum phase system. As a result the tracking error during the transient non-minimum phase response will be greater but the overall system response will be faster (Soeterboek (1992)).

2. Maximum Prediction Horizon - N_2

The maximum prediction horizon, N_2 specifies the cut-off point for the inclusion of tracking errors in the cost function. It is reasonable to include tracking errors only up to the predicted settling time of the closed loop system. One rule of thumb suggests that using the 5% settling time of the closed loop process to select N_2 yields good control performance :

$$N_2 = \text{int}(T_{settle}) / \Delta T \tag{7.11}$$

Decreasing N_2 places greater emphasis on early tracking errors and decreases the system response time.

3a. Controller Horizon - N_u

Controller computations can be considerably reduced by specifying a controller horizon, N_u, beyond which the controller action is assumed to remain constant. Fig 7.3 shows the computed control action where N_u is equal to 3.

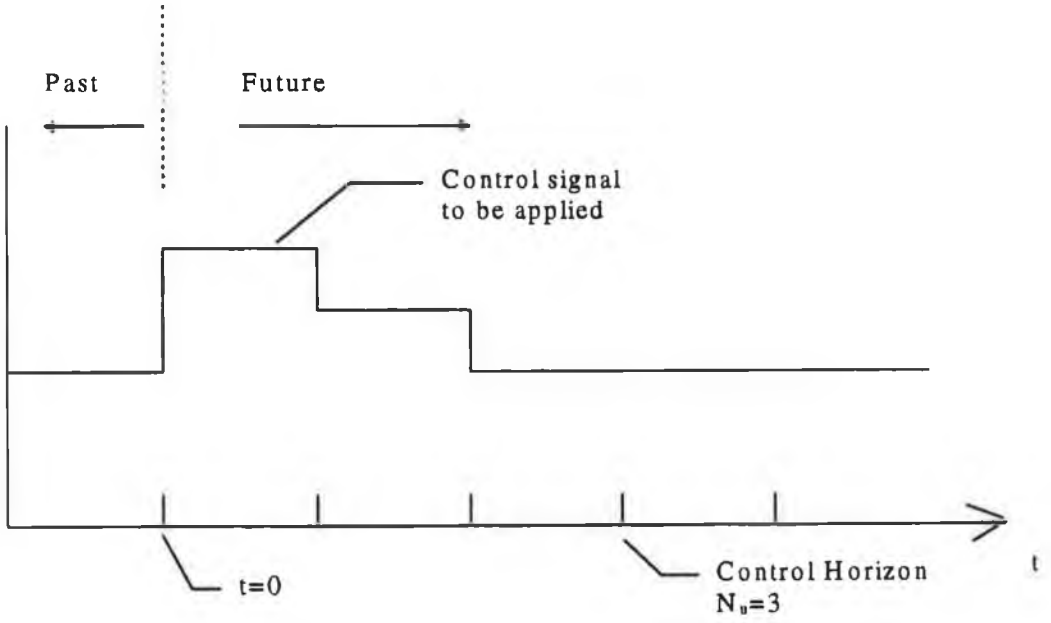


Fig. 7.3 Computed Control Signal for $N_u = 3$

It can be seen from Fig. 7.3 that the value of the computed control action is allowed to vary for the first three sampling periods and is thereafter forced to remain at the value computed for the third sampling period. In practice, only the control signal calculated for the first sampling period is applied to the plant, as the control signal is recalculated at each sampling period. In spite of this the value of N_u does has a significant role to play in terms of controller tuning. For example, if a control horizon of 1 is specified, the predictive controller calculates at each sampling instance the magnitude of the step which would bring the plant to the setpoint with the same dynamics as the open-loop plant. The mechanics of this are described in Section 8.5. If N_u is increased to 2, the control signal computed for the first sampling period may allow the plant to overshoot the setpoint, the control signal computed for the second sampling period should bring the plant to the setpoint. Given that only the first and larger control signal is repeated at each sampling interval, the dynamics of the closed loop system will be considerable faster than those achieved using a control horizon of 1.

In general, increasing N_u makes the controller more active and reduces robustness. If N_2 is large, increasing N_u from *1* to *na+1*, (where *na* is the order of plant) can in certain cases transform the controller from a mean-level to a dead-beat controller (Soeterboek (1992)). Consequently if the process is unstable, a value of *Nu* greater than *1* must be chosen, as mean-level controllers cannot control unstable processes.

3b. Control Output Structuring

In PFC the control horizon, N_u, is always assumed to be equal to the prediction horizon, H_2. If the prediction horizon is long, this greatly increases the complexity of the optimisation problem. At each sampling interval, the cost function must be optimised with respect to H_2-1 variables. In addition, as explained in the previous section, it would result in a very active control signal. The problem is solved by "structuring" the projected control signal i.e. the projected control signal must fit a specified function, known as a base function. The most simple base function is a step function. This is effectively equivalent to specifying a control horizon of 1. The projected control signal may change at the first sampling period only. Structuring the controller output has the same advantages as using a controller horizon. It reduces controller computations and can be used to tune the controller.

4a. Output Weighting Matrix - Q

Where the desired reference trajectory is the plant set-point, it is inevitable that large initial tracking errors will occur after a step change in the set-point. These large errors will give rise to very active controller action after a set-point change. The tracking error may be weighted so as

to place a low cost on the initial, inevitably large tracking errors and high cost on later tracking errors. Given the following cost function, which penalises tracking error only for a SISO system:

$$J = \left[\hat{y}(k+N_1) - w(k+N_1) \quad ..... \quad \hat{y}(k+N_2) - w(k+N_2)\right] Q \begin{bmatrix} \hat{y}(k+N_1) - w(k+N_1) \\ ..... \\ \hat{y}(k+N_2) - w(k+N_2) \end{bmatrix}$$

where (7.12)

$$Q = \begin{bmatrix} q_1 & ..... & 0 \\ .... & .... & .... \\ 0 & ..... & q_n \end{bmatrix}$$

The penalty on the early tracking errors can be reduced by defining q_i as follows:

$$q_i = \lambda^{H_2 - 1}$$

where (7.13)

$$\lambda \leq 1$$

In general, this results in smoother control as the controller does not attempt to reduce the initial large tracking errors. If a small or even non-existent penalty is specified for the first m tracking errors, this is equivalent to specifying a minimum prediction horizon, N_1, equal to $m+1$. As a result, this type of output weighting can have the same undesirable effect on non-minimum phase plants as a minimum prediction horizon which lies beyond the non-minimum transient - it may result in more rather than less active control.

4b. Set-Point Pre-filtering/Reference Trajectory - *w(k)*

An alternative means of reducing the impact of initial setpoint tracking errors is to employ a setpoint pre-filter. This is the approach adopted by PFC, IMC, (Morari and Zafiriou (1989) and more recently by GPC (Robinson and Clarke (1991). A simple cost function which penalises tracking errors only and includes a setpoint pre-filter is:

$$J = \sum_{j=N_1}^{N_2} [\hat{y}(k+j) - Qw(k+j)][\hat{y}(k+j) - Qw(k+j)]^T \tag{7.14}$$

A controller with pre-filter has the "two-degrees-of-freedom-structure", generally attributed to Horowitz (1963). It allows the controller to be independently designed for good disturbance rejection and setpoint tracking. The traditional one-degree-of -freedom-structure cannot offer this as both disturbances and reference signals affect the controller through the same transfer function. The one-degree-of-freedom-structure is shown in Fig. 7.4

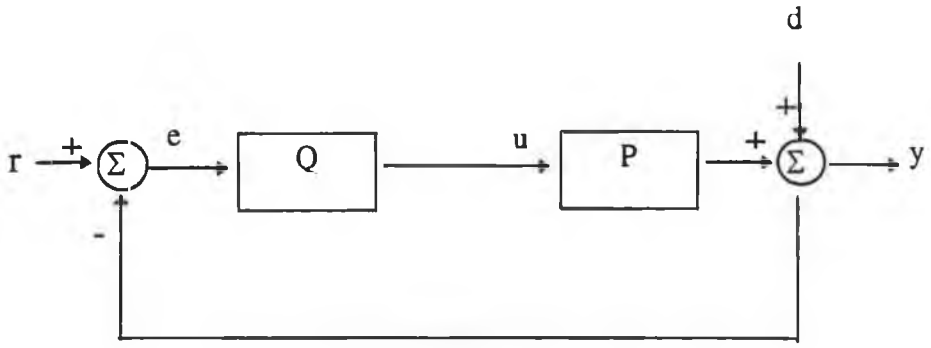


Fig. 7.4 One-Degree-of-Freedom-Structure

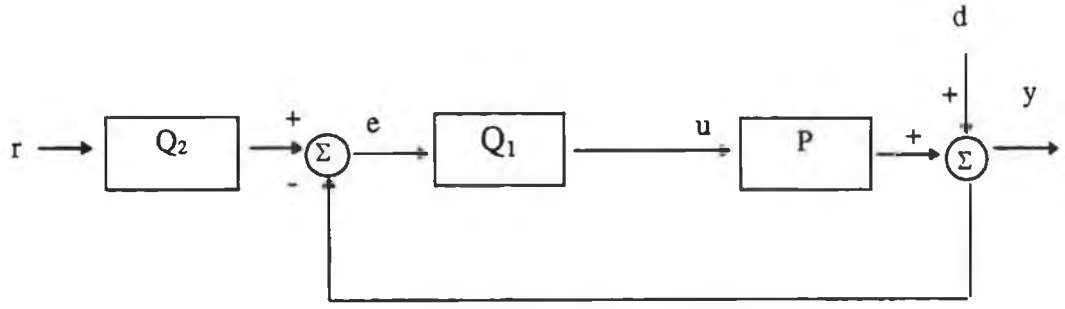
For this control structure the relationship between the controller input, e, disturbance signal, d, and reference signal, r, is equal to:

$$\frac{e}{d-r} = \frac{-1}{1+PQ}$$

where (7.15)

$$e = r - d$$

From this, it can be seen that r and d have the same effect on e (apart from sign). This does not present a problem if the reference signal, r and the disturbance signal, d behave in a similar manner. In that case, Q can be designed to yield a good response to a change in r or in d. If the behaviour of r and d is dissimilar however it may not be possible to design a controller, Q which can provide both satisfactory setpoint tracking and disturbance rejection. This problem can be solved by introducing additional controller blocks into the system as shown in Fig. 7.5



Fig, 7.5 Two-Degrees-Of-Freedom-Structure

The relationship between error and setpoint and disturbance inputs for this controller is:

$$e = \frac{-1}{1+PQ_1} d - \left(\frac{PQ_1Q_2}{1+PQ_1} - 1 \right) r \qquad (7.16)$$

It is now possible to design Q_1 for good disturbance rejection and to design Q_2 independently for good setpoint tracking.

In PFC, pre-filtering is extended to the plant output as well as the setpoint, as demonstrated in Fig, 7.6. This configuration, in effect, pre-filters the tracking error.

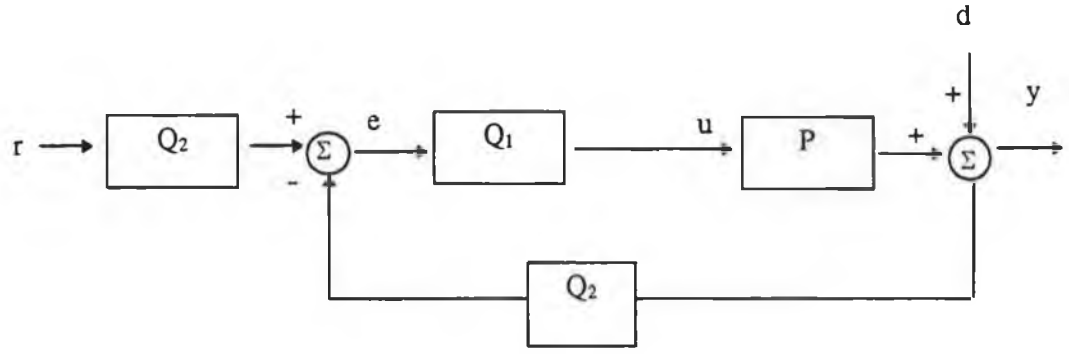


Fig. 7.6 PFC Error Prefiltering

Pre-filtering the error signal is equivalent to pre-filtering the setpoint sequence with a filter which is initialised at the current output value of the plant. This has a similar effect to using an output weighting matrix which places a low cost on the initial high tracking errors and a higher cost on future, small tracking errors. As a result system response can be controlled by changing the order and the dynamics of the pre-filter. PFC specifies the system response through the pre-filter dynamics only. This eliminates the use of non-intuitive output a weighting matrices.

Fig. 7.7 demonstrates setpoint pre-filtering following a setpoint change (indicated by the dashed line) at $t=0$. The first plot shows the projected, filtered setpoint sequence (indicated by a continuous line) at $t=1$ The controller uses this filtered sequence as the desired reference trajectory of the controlled variable. It calculates a control signal which minimises the error between the predicted plant output and this desired reference trajectory over the prediction horizon. The second plot shows that the calculated control signal was not exactly sufficient to place the actual plant output (indicated by '*') onto the desired reference trajectory at $t=2$. This discrepancy between desired and actual plant response may be due to unmodelled dynamics or actuator constraints. The desired reference trajectory is initialised with the actual value of the plant output at $t=2$. The controller again computes the control signal which minimises the error between the predicted and the plant output over the prediction horizon. This procedure is repeated at each sampling instance. The third plot shows the desired reference trajectory employed at each sampling instance up to $t=11$, and the actual measured value of the controlled variable at each sampling instance.

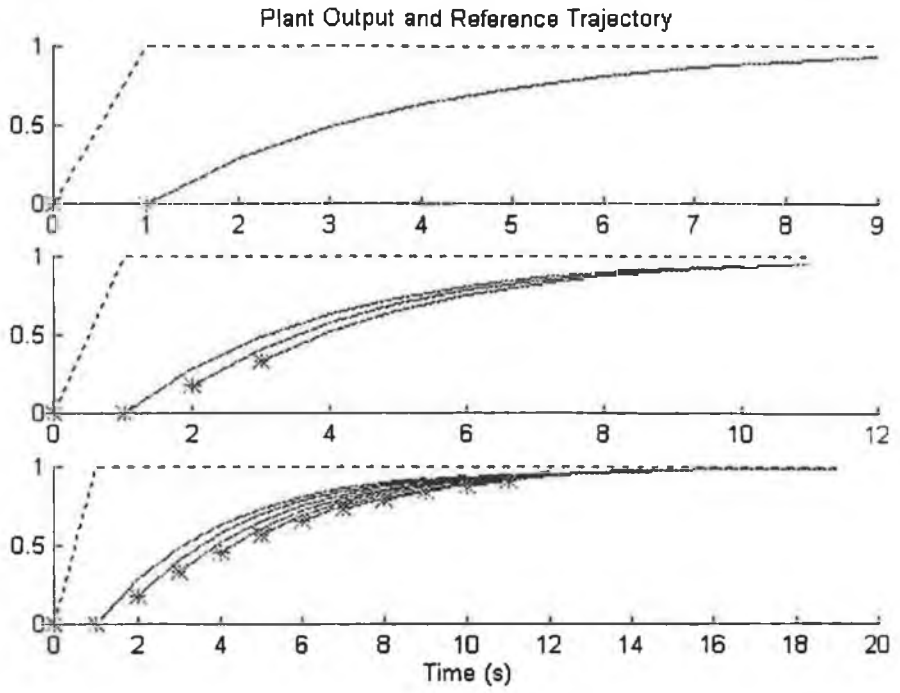


Fig. 7.7 Initialisation of Reference Trajectory at each Sampling Instance

5a. Controller Output Weighting Matrices - R

If the controller output weighting matrix R is set equal to zero, and setpoint pre-filtering is not employed, a minimum variance controller results. It can be shown (Soeterboek (1992) that the zeros of the open-loop process appear as closed-loop poles of a minimum variance controller. As a result, minimum variance control is not suitable for processes with badly situated zeros (most discretised processes). The position of the closed-loop poles can be changed by using a controller output weighting matrix, R which is greater than zero. This takes the variance of the controller output signal into account and reduces controller activity.

For a type 0 process, the steady state error is a function of ρ. To overcome this, many controllers (including GPC and DMC) weight the controller increments instead of the controller outputs i.e. use a weighting matrix R, equal to $\rho\Delta$. This, however, can result in a badly damped or unstable closed loop system. Essentially Δ has the filtering affect of weighting the high frequencies of $u(k)$ much more than the low frequencies. Soeterboek (1992) suggests using a filter which does not pass frequency zero, but passes all higher frequencies, one possibility is:

$$R = \frac{1-q^{-1}}{1-mq^{-1}} \qquad \text{where} \quad m \rightarrow 1 \tag{7.17}$$

The filter could also be designed to attenuate other specific frequencies such as the resonant frequencies of a mechanical system or high frequency measurement noise.

5b. Controller Output Filtering

In PFC, the controller output weighting matrix, R, is set equal to zero. High frequency control signals are eliminated by filtering the calculated control signal. This has a similar effect as the inclusion of a controller output weighting matrix, however, it is has the advantage of simplicity. It is easier to choose a filter time constant which is a physically meaningful parameter than it is to choose a controller output weighting matrix. In addition, a change in the filter time constant does not require a revaluation of the control law. This technique is also endorsed by Morari and Zafiriou (1989) for their Internal Model Control strategy.

The disadvantage of this approach is that the effect of the filter is not taken into account in the controller design procedure. The controller is designed and then augmented with the controller output filter. Inclusion of the filter could potentially destabilise the closed loop system.

7.2.3 Controller Solution

The controller action is derived by minimising the cost function. The minimisation problem can be solved in two different ways - analytically and numerically.

Analytical solutions can be found if the internal controller model is linear and the cost function is quadratic. Linear least squares can be used to derive an analytical controller solution for GPC or PFC. The control problem is solved using a Ricatti equation for RHC. Linear Programming techniques are used for controllers with a 1-norm or ∞-norm cost index.

If the internal controller model is nonlinear *or* if the cost function includes constraints, the cost function must be minimised using a numerical optimisation algorithm.

7.3 Stability

Terminal equality constraints of the form $x(N_2) = 0$ can be used to guarantee system stability. Stability results was first obtained for a SISO discrete time system with a non-singular transition matrix (Kwon and Pearson (1975). More recently it has been shown for MIMO systems, even if the transition matrix is singular (Chisci and Mosca (1994)).

There have been several recent developments in the design of stabilising predictive controllers for systems with input and output constraints. It has been shown that a predictive controller can globally stabilise a linear time system with constraints provided that a feasible solution exists (Rawlings and Muske (1993)).

7.4 Robustness

Robustness implies the ability of a controller to maintain system performance in the presence of unmodelled dynamics. The more robust the controller, the less sensitive is controller performance to mismatches between the plant and the internal controller model. The robustness of a predictive controller is dictated by the controller design. A predictive controller is not necessarily more robust than a classic feedback controller. (Garcia *et al* (1989) referring to MPC).

There are a number of mechanisms however for improving the robustness of a predictive controller. For example, robustness may be included explicitly at the controller design stage, through the use of a 'robust' cost function statement which penalises the sensitivity of controller performance to model mismatch. It can also be included by augmenting the controller with a set-point pre-filter or control signal filtering. It has been shown that both the set-point pre-filter (Robinson and Clarke (1989) and the control signal filter (Morari and Zafiriou (1989)) have a direct effect on robustness.

7.5 Conclusions

Predictive control is a flexible and potentially effective control theory . The general objective of a predictive control strategy is to ensure that the future plant response matches a pre-specified desired plant response. This general objective is translated into a specific optimisation problem by defining an internal controller model and a cost function.

Much of the strength of predictive control lies in the flexibility of its controller definition. A predictive controller is largely defined by its internal model and cost function, both of which may be customised to suit many types of control problems. For example, there are no restrictions on the type of internal model which can be used by a predictive controller. As a result the internal controller model can represent any type of plant including nonlinear, multivariable or time-delay plants. The cost function definition may also be adapted to suit a particular control problem For example, the cost function may be defined as a 1-norm, 2-norm or ∞ - norm. The cost function may include terms such as controller action in addition to tracking error. The cost function may be defined as subject to any type of input or output constraint. The flexibility inherent in the generic definition of a predictive controller is reflected in the diversity of existing predictive control methodologies.

8. Development of a Fuzzified Linear Predictive Controller

8.1 Introduction

This chapter describes the development of an analytical linear predictive control strategy for a boiler process. The boiler process is a complex, multivariable process which includes significant time delay. The predictive control methodology is suitable for this application for a number of reasons. Firstly, it is a model based strategy which can take advantage of modelled knowledge about the expected future behaviour of the boiler process. Secondly, it can be implemented as an MIMO controller if necessary, in order to take into account modelled knowledge about the interaction among the controlled variables. Thirdly, it can easily be adapted to control systems with time delay.

An analytical linear design strategy does have limitations however. A single linear controller which has been specified and tuned to operate at a particular operating point cannot perform equally well at other operating points if the system is nonlinear. For this work, a number of linear controllers have been used to provide control over the full operating range of the plant. Fuzzy logic is used to combine the different controller output signals, thus ensuring a smooth transition between the various linear controllers. The resulting controller will be referred to as a fuzzified linear controller to indicate that the controller has been extended using fuzzy logic. The fuzzified linear controller is in fact a nonlinear controller, but the term "nonlinear controller" will be reserved for a nonlinear controller which is based upon a single nonlinear model of the plant.

A further limitation of an analytical linear approach is that it is not possible for the controller solution to take account of hard nonlinearities such as actuator constraints. These could be taken into account if a numerical optimisation algorithm is used to obtain the control solution. Instead, the fuzzified linear controller offers the benefits of an analytical control solution which can be calculated quickly using previously determined controller coefficients.

8.2 Overview of Control Strategy

The boiler process considered in this study has four controlled variables. These are to be controlled using four manipulated variables. Each of the controlled variables is affected to some extent by a variation in any of the manipulated variables. However, each controlled variable is predominantly affected by a variation in one particular manipulated variable. Table 8.1 lists the controlled variables and their associated manipulated variables for this boiler:

Controller Variables	Associated Manipulated Variables
Main Steam Pressure	Fuel Mass Flow Rate
Drum Level	Feedwater Mass Flow Rate
Main Steam Temperature	Attemperating Water Mass Flow Rate
Percentage of O_2 in Stack Gases	Air Mass Flow Rate

Table 8.1 Controller Input and Output Variables

The overall control strategy adopts a combined multi-loop and single-loop approach. There is little interaction between the percentage of O_2 in the stack gases and the three fluid-side variables (pressure, temperature and drum level). Consequently the percentage of O_2 in the stack gases is controlled by a single-loop predictive controller, with a sampling rate of 0.1s, using air flow as the manipulated variable.

On the fluid-side, there are significant interactions between the three controlled variables and initially it seemed likely that a full multivariable approach could be of benefit here. In practice, this approach was not practical as the superheater steam temperature dynamics are considerably faster than the superheater steam pressure and drum level dynamics. As a result, it is not possible to select a sampling period which is suitable for the control of all three variables. Consequently, steam temperature is controlled by a fast single-loop predictive controller with a sampling period of 0.1s, using attemperation as the manipulated variable.

Steam pressure and drum level are controlled by a slower multivariable controller, with a sampling period of 1s, using fuel flow and feedwater flow as the manipulating variables.

The overall control strategy for the fuzzified linear predictive controller is represented schematically in Fig. 8.1

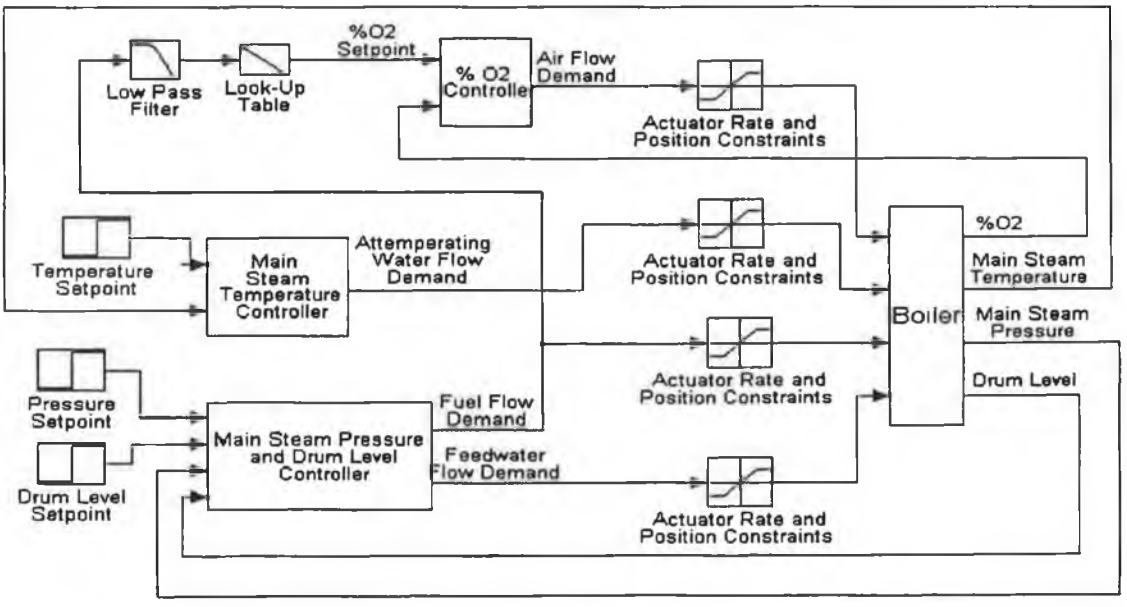


Fig. 8.1. Schematic of Fuzzified linear Predictive Control Strategy

It can be seen from Fig. 8.1 that the setpoint of the percentage O_2 controller is not fixed, but is dictated by the fuel mass flow rate. At all times, sufficient air must be available for stoichiometric combustion plus an amount of excess air for safety reasons. The excess air is regulated by controlling the percentage of O_2 in the stack gases. At lower loads, a considerable delay exists between combustion and the measurement of the percentage of O_2 in the resulting stack gases. To ensure that sufficient excess air is available for combustion, the setpoint of the percentage of O_2 in the stack gases is held at approximately 10%. At higher loads, the delay between the combustion chamber and the measurement of the resulting percentage O_2 in the stack gases is shorter. As a result, the setpoint of the percentage O_2 in the stack gases can be safely reduced to approximately 1%. This reduces the flow of excess air through the furnace, thus improving boiler efficiency. Fig. 8.2 shows the relationship between excess air and combustion losses. If the air available for combustion is less than the stoichiometric air requirement, some of the fuel is unburned resulting both in low efficiency as well as the production of carbon monoxide gases and the possibility of unburned fuel exploding at the top of the stack. Combustion losses are at a minimum when a small amount of excess air is available. If the excess air is increased beyond this optimum point, however, boiler efficiency is reduced because the excess air is heated by combustion but serves no useful purpose.

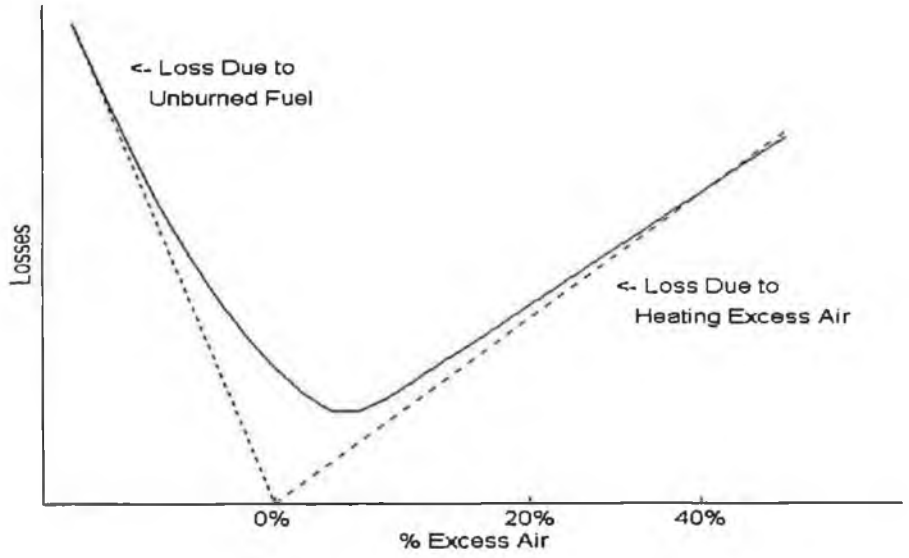


Fig. 8.2 Relationship Between Excess Air and Combustion Losses

The setpoint of the percentage O_2 controller should not be set exactly equal to the optimal value for efficient control as small deviations from the setpoint could result in sub-stoichiometric control. The size of the safety margin necessary between the percentage O_2 setpoint and the optimal value of percentage O_2 is largely dependent on the performance of the O_2 controller. Good control reduces the standard deviation of the control signal and allows the percentage O_2 setpoint to be shifted closer to the optimum value, without increasing the risk of the sub-

stoichiometric combustion. This reduces the excess air in the furnace and improves combustion efficiency. Fig 8.3 illustrates this point It compares the two controllers on the basis of the standard deviation of their respective control signals about their setpoints. The controllers are required to keep the controlled variable above a value of 5 at all times. Clearly, the control signal of the 'Controller A' has a much smaller standard deviation than the control signal of 'Controller B'. In consequence of this the setpoint of 'Controller A' may be set to a value of 10, without danger of the controlled signal falling below 5. In contrast the setpoint of 'Controller B' must be set to at least 15 in order to ensure that the controlled signal does not fall below 5 (Richalet *et al* (1978)).

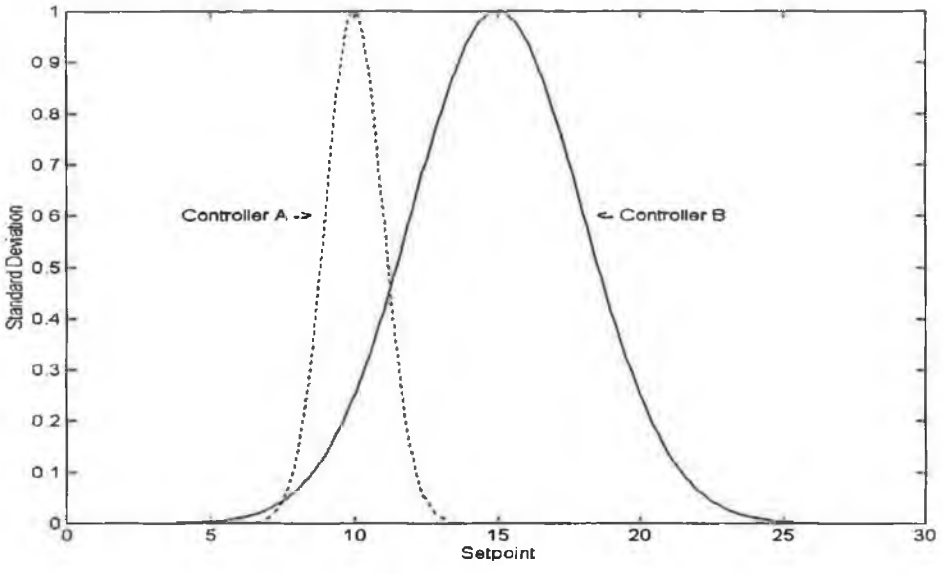


Fig. 8.3 Standard Deviation of Control Signals of 'Controller A' and 'Controller B' About Their Setpoints

In this case, the setpoint signal is derived from the fuel flow signal which is used as a measure of the plant operating point. The fuel flow rate signal is first low-pass filtered to prevent transient variations in fuel flow rate being transmitted to the air flow controller *via* the percentage O_2 setpoint signal. The relationship between boiler operating point and percentage O_2 setpoint is presented in Table 8.2.

Operating Point (%)	**O_2 Setpoint (%)**
10%	8.75
30%	5.75
50%	3.00
70%	1.60
90%	1.11

Table 8.2 Relationship Between Operating Point and Percentage O_2 Setpoint

Each controller is multivariable in the sense that each internal controller model is provided with appropriate disturbance inputs. For example, the steam temperature controller calculates the attemperating water flow rate signal, but it is also provided with the current fuel mass flow rate and load demand signal as these directly effect steam temperature. Similarly, the percentage O_2 controller is provided with fuel flow as a disturbance input. The complete list of disturbance inputs used by each controller is given in Table 8.3.

Main Steam Pressure and Drum Level Controller	Main Steam Temperature Controller	%O_2 Controller
1. Nr. of Open Valves (load input) 2. Attemperating Flow Rate 3. Air Flow Rate 4. Feedwater Temperature	1. Fuel Flow Rate 2. Air Flow Rate 3. Nr of Open Valves (load input) 4. Drum Pressure 5. Drum Temperature 6. Combustion Gas Temperature	1. Fuel Flow Rate

Table 8.3 Disturbance Inputs of the Fuzzified linear Predictive Controllers

8.3 Linear Predictive Controller Development

This section describes the development of a single linear controller which is based upon a linear model of the plant. The combination of a number of these linear controllers to produce the fuzzified linear controller is described in Section 8.4. A predictive controller attempts to achieve its objective by finding the controller action which minimises an appropriate cost function based on predicted future behaviour of the process. The cost function must consider the error between the predicted plant output and a reference trajectory over a finite or even infinite time span, known as the prediction horizon. The solution to the minimisation problem specifies the controller output for each sampling period in the prediction horizon. However, only the controller output for the first sampling period is actually applied. At the next sampling period, a new controller solution is calculated, which takes into account any new information such a disturbance or change in setpoint.

A linear predictive controller includes the following three fundamental components:

- Linearised Model
- Quadratic Cost Function
- Analytical Controller Solution

This following derivation uses the predictive control methodology, known as Predictive Functional Control (Richalet (1993a)). In Chapter 7, the features of Predictive Functional Control were identified and compared to those of other predictive control methodologies.

8.3.1 Linearised Model

Each of the three controllers requires a model of a different section of the plant. The controller of the percentage of O_2 in the stack requires a model of the combustion-side of the process in order to predict the percentage of O_2 in the stack. The multivariable steam pressure and drum level controller needs a model of the complete process in order to predict steam pressure and drum level. The steam temperature controller requires a model of the superheater in order to predict main steam temperature.

These three linearised models are obtained by linearising the nonlinear first-principles boiler model, described in Chapter 4. The linearisation process, which is described in Chapter 5, generates a state-space description of the plant:

$$x(n+1) = Ax(n) + Bu(n)$$
$$y_{m_i}(n) = C_i x(n)$$

where

$x(n)$ = vector of state variables of model
$u(n)$ = vector of manipulated variables
C_i = ith row of C
$y_{m_i}(n)$ = one step - ahead prediction for ith controlled variable (8.1)

A multi-step-ahead prediction requires the value of future controller outputs. These future control outputs are unknown as the controller output is recalculated as each sampling period. Therefore it is assumed that the future control output remains constant. Adopting this assumption, the predicted model output is:

$$y_{m_i}(n+H_j) = C_i A^{H_j} x(n) + C_i B \sum_{p=1}^{H_j} A^{p-1} u(n)$$

where (8.2)

H_j = jth coincidence point

8.3.2 Quadratic Cost Function

The cost function is specified as the sum of the squares of the errors between the change in the desired reference trajectory and the change in the predicted model output at specified points in the future called coincidence points.

$$J = \sum_{i=1}^{m} \sum_{j=1}^{k(i)} E_{i,j}^2$$

where

J = cost function value
m = number of controlled variables
$k(i)$ = number of coincidence points for ith controlled variable
$E_{i,j}$ = error at jth coincidence point for ith controlled variable (8.3)

The error between the desired change in model output and the predicted change in model output at some coincidence point H_j, can be expressed mathematically as:

$$E_{i,j} = (y_{m_i}(n+H_j) - y_{m_i}(n)) \; -(R_i(n+H_j) - R_i(n))$$

where

$E_{i,j}$ = error at j[th] coincidence point for i[th] output
y_{m_i} = i[th] model output
H_j = j[th] coincidence point
R_i = reference trajectory for i[th] output

(8.4)

The desired reference trajectory may simply be equated to the set-point signal. Smoother control may be obtained, however, by low-pass pre-filtering the set-point signal. In this case the setpoint signal is pre-filtered by specifying the desired reference trajectory as an exponential curve between the current process output and the set-point. It follows that the desired *error* trajectory between the process output and the setpoint is an exponentially decreasing curve. The error trajectory is described mathematically as:

$$\varepsilon(n+H_j) = \lambda_i^{H_j}\varepsilon_i(n)$$

where

$$\varepsilon_i(n) = S_i - y_{p_i}(n)$$

S_i = Setpoint for i[th] process output
y_{p_i} = i[th] process output (controlled variable)
$0 \le \lambda_i \le 1$

(8.5)

The physical relevance of λ_i can be seen by writing it in terms of the 63% rise-time of the desired exponential curve.

$$\lambda_i = \exp\left(\frac{-T_s}{\tau_i}\right)$$

where

T_s = sampling period
τ_i = reference trajectory time constant or 63% rise - time

(8.6)

The desired change in reference trajectory can now be rewritten as:

$$R_i(n+H_j) - R_i(n) = \varepsilon_i(n) - \varepsilon_i(n+H_j) = \varepsilon_i(n)(1-\lambda_i^{H_j})] \tag{8.7}$$

This expression is incorporated into (8.4), yielding:

$$E_{i,j} = (y_{m_i}(n+H_j) - y_{m_i}(n) - \varepsilon_i(n)(1-\lambda_i^{H_j}) \tag{8.8}$$

The equations for model and predictor output ((8.1) and (8.2) respectively) are substituted into equation (8.8), yielding:

$$E_{i,j} = C_i(A^{H_j} - 1)x(n) + C_i B \sum_{p=1}^{H_j} A^{p-1} u(n) - (1 - \lambda_i^{H_j})\varepsilon_i(n) \tag{8.9}$$

For brevity, the error can be rewritten as:

$$E_{i,j} = L_{i,j}x(n) + M_{i,j}u(n) + N_{i,j}\varepsilon_i(n)$$

where

$$\begin{aligned} L_{i,j} &= C_i(A^{H_j} - 1) \\ M_{i,j} &= C_i B \sum_{i=1}^{k} A^{i-1} \\ N_{i,j} &= -(1 - \lambda_i^{H_j}) \end{aligned} \tag{8.10}$$

The complete set of equations for all the model outputs and all the coincidence points is rewritten in matrix format, (* denotes vector dot product multiplication)

$$E = Lx(n) + Mu(n) + N * \varepsilon(n) \tag{8.11}$$

The cost function, restated in terms of (8.11) is:

$$\begin{aligned} J &= E^T E \\ &= (Lx(n) + Mu(n) + N.\varepsilon(n))^{\mathrm{T}} (Lx(n) + Mu(n) + N * \varepsilon(n)) \end{aligned} \tag{8.12}$$

This cost function is simple but still allows great freedom in controller tuning. The system response can be changed in two different ways. Firstly, the rate of change of the desired reference trajectory may be varied *via* the reference trajectory time constant. Secondly, the position of the coincidence points may be changed - choosing coincidence points in the near future results in a more active control action. An attractive feature of this method is that both of these tuning parameters are physically meaningful. In addition, by stating the cost function incrementally, it is shown in the next section that the effect of steady-state errors is automatically compensated for by the controller.

8.3.3 Derivation of Control Law

The model inputs may include disturbance inputs as well as manipulated inputs If this is the case the vector of model inputs is partitioned into two parts: u_c is the set of manipulated variables and

u_d is the set of plant disturbance inputs. The cost function becomes:

$$\begin{aligned} J &= E^T E \\ &= (Lx(n) + \begin{bmatrix} M_c & M_d \end{bmatrix} \begin{bmatrix} u_c(n) \\ u_d(n) \end{bmatrix} + N.\varepsilon(n))^{\mathrm{T}} (Lx(n) + \begin{bmatrix} M_c & M_d \end{bmatrix} \begin{bmatrix} u_c(n) \\ u_d(n) \end{bmatrix} + N * \varepsilon(n)) \end{aligned} \tag{8.13}$$

The control solution is found using the method of least squares. The partial derivative of J is minimised with respect to the manipulated inputs, $u_c(n)$, to yield the controller solution.

$$\begin{aligned} &\frac{\delta E}{\delta u_C} = 2u_c(n)^T M_c^T M_c + 2x(n)^T L^T M_c + 2\varepsilon^T N^T M_c + 2u_d(n)^T M_d^T M_c = 0 \\ &\Rightarrow u_c(n) = -(M_c^T M_c)^{-1} M_c^T (Lx(n) + N * \varepsilon(n) + M_d u_d(n)) \end{aligned} \tag{8.14}$$

It can be shown that the control law includes integral action by considering the control law for a SISO system and a single step-ahead prediction horizon. Given a SISO system, equation (8.14) can be reduced to:

$$u_c(n) = -M_c^{-1}(Lx(n) + N * \varepsilon(n)) \tag{8.15}$$

If the plant output is equal to the setpoint the controller action is equal to:

$$u_c(n) = -M_c^{-1} Lx(n) \tag{8.16}$$

Replacing M_c and L with their full expressions, yields:

$$u_c(n) = -(CB\sum_{i=1}^{k} A^{i-1})^{-1}(C(A^H - 1))x(n) \tag{8.17}$$

If a single step prediction horizon is employed, the control law may be reduced to:

$$\begin{aligned} u_c(n) &= -(CB)^{-1}(C(A-1))x(n) \\ &= B^{-1}(I - A)x(n) \end{aligned} \tag{8.18}$$

This is also the value of control action required to place the internal controller model into steady state, as can be seen by rearranging equation (8.18). The rearranged equation is equal to the equation for a discrete linear system in steady-state.

$$x(n) = Ax(n) + Bu_c(n) \tag{8.19}$$

If the internal controller model is not in steady-state equilibrium, the controller output will not be constant. It follows that the plant can only attain steady state equilibrium if the plant output is equal to the setpoint.

8.3.4 Application to Systems With Time Delay

Control of percentage O_2 in the stack gases is complicated by the presence of a pure time delay between controller action and percentage O_2 measurement. The control law must be adapted to take the effects of this time delay into account. In Predictive Functional Control, the approach used is to consider the process as two separate entities - a pure time delay and a process with no time delay. This notional representation of a time delay system is depicted graphically in Fig. 8.4

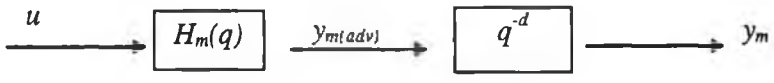


Fig. 8.4 Model of Time Delay System

The output of the model $H_m(q)$ has been termed $y_{m(adv)}$ as it is equivalent to the actual model output, y_m, advanced by d sampling periods. The relationship between y_m and $y_{m(adv)}$ is expressed mathematically as follows:

$$y_{m(adv)}(n) = y_m(n+d) \tag{8.20}$$

For control purposes it is assumed that the process variable corresponding to $y_{m(adv)}$ is the controlled variable i.e. that the controlled system excludes the pure measurement delay. In reality, there is no measurable process variable, $y_{p(adv)}$, which corresponds to $y_{m(adv)}$. However, the process model is used to estimate the value of this notional variable - $y_{p(adv)}$

$$y_{p(adv)}(n) = y_p(n) - y_m(n) + y_{m(adv)}(n) \tag{8. 21}$$

The reference trajectory is initialised by this advanced process output instead of by the actual process output. The desired reference trajectory is now defined as follows:

$$\varepsilon(n+H_j) = \lambda_i^{H_j}\varepsilon_i(n)$$

where

$$\varepsilon_i(n) = S_i - y_{p(adv)_i}(n)$$

S_i = Setpoint for i^{th} process output

$y_{p(adv)_i}$ = i^{th} advanced process output (controlled variable)

$0 \le \lambda_i \le 1$ (8.22)

This nominal separation of a process into a non-delay system and a pure time delay was suggested by Smith (1958) and forms the basis for a feedback controller of time delay systems which is referred to as the Smith regulator. A feedback structure for a system with delay, as shown in Fig. 8.5:

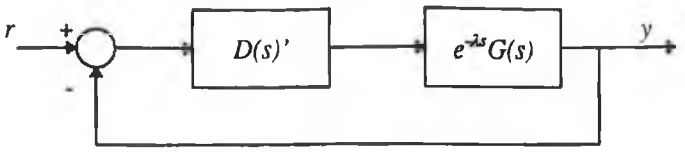


Fig. 8.5 Feedback Structure for Time Delay System

The transfer function for this system is:

$$\frac{Y(s)}{R(s)} = \frac{D(s)' e^{-\lambda s} G(s)}{1 + D(s)' e^{-\lambda s} G(s)} \tag{8.23}$$

Smith suggested that if the plant delay could be taken outside the feedback loop, a modified compensator $D(s)$ could be designed to control $G(s)$ alone i.e. the plant excluding pure time delay. The transfer function of the system would then be:

$$\frac{Y(s)}{R(s)} = \frac{D(s)G(s)}{1 + G(s)D(s)} e^{-\lambda s} \tag{8.24}$$

By equating equation (8.23) and (8.24) the transfer function of the modified compensator is found to be:

$$D(s) = \frac{D(s)'}{1 + D(s)'[-G(s) + G(s)e^{-\lambda s}]} \tag{8.25}$$

The structure of this compensator is shown in Fig. 8.6

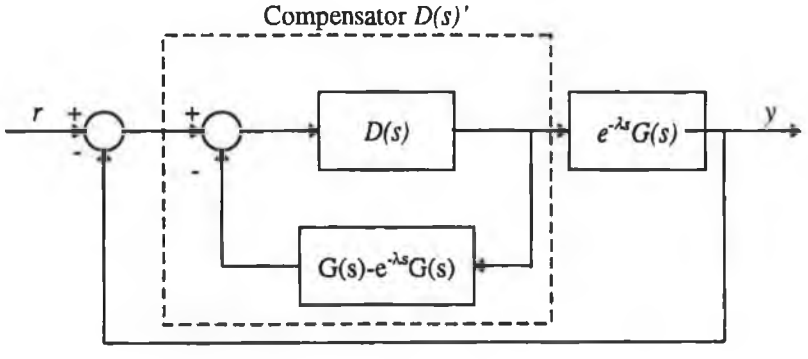


Fig. 8.6 Smith Regulator for Time Delay System

It can be seen from Fig. 8.6 that the output of the plant without the delay is fed back to the controller. This is exactly equivalent to the structure of the linear predictive controller for time delay systems.

8.4 Fuzzy Interpolation Between Linear Controllers

The previous section describes the development of a linear predictive controller which is based upon a linear model of the plant. The linear model represents the plant well around a particular operating point, but it does not necessarily represent the plant well at other operating points. Likewise the controller tuning parameters have been chosen to give the best possible results at a particular operating point and may give poor results at other operating points. In effect, it is not possible to achieve good controller performance over the full operating range of a nonlinear plant using a single linear controller. This problem is generally overcome using either adaptive or gain scheduling techniques.

Adaptive controllers continuously recalculate the parameters of the linear model using an estimation technique such as recursive least squares. This method has three disadvantages. Firstly, on-line adaptation of the model parameters can result in instability or unpredictable behaviour (Rohrs *et al* (1985)). Secondly, it is only suitable for models which have a fairly small number of parameters for computational reasons. Thirdly, it involves recalculation of the controller parameters at each sampling period.

For gain scheduling, model parameters are first obtained off-line for a number of different operating points spread over the full operating range of the plant (e.g. 10%, 50%, 90%). A simple gain scheduling controller switches between these models as the operating point changes from the operating region around one model to the operating region around the next model. This method may result in poor control near the boundary of an operating region or in a "bumpy" changeover from one region to the next. One popular means of overcoming these problems is to estimate the model parameters at intermediate operating points by interpolating between the parameters of known linear models. The controller parameters may then be calculated on-line from the interpolated model parameters. A second popular method interpolates between the parameters of two linear controllers to derive the parameters of a controller at an intermediate operating point.

The first of these methods assumes that interpolation will result in a model or controller which is capable of representing the plant at the intermediate operating point. A comparison of the parameters of several linearised models of the boiler indicates that the model parameters do not increase or decrease monotonically between operating points. The transition matrices of the 10%, 50% and 90% linear models are compared on an element by element basis. The comparison showed that 11 of the 400 elements neither increase nor decrease monotonically with operating point. It follows that simple linear interpolation between the parameters of two linearised models

will not necessarily generate accurate model parameters at an intermediate operating point. It also follows that simple linear interpolation between the parameters of two linear controllers will not necessarily generate appropriate controller parameters at an intermediate operating point.

An alternative control strategy which uses fuzzy logic was developed for this work. Three linear controllers, based on linearised models of the boiler at the 10%, 50% and 90% operating points are left to run concurrently. The actual controller output is equal to a weighted sum of the output of each of the three controllers. The weights are dependent on the current operating point of the boiler and on pre-specified fuzzy sets, shown below in Fig. 8.7.

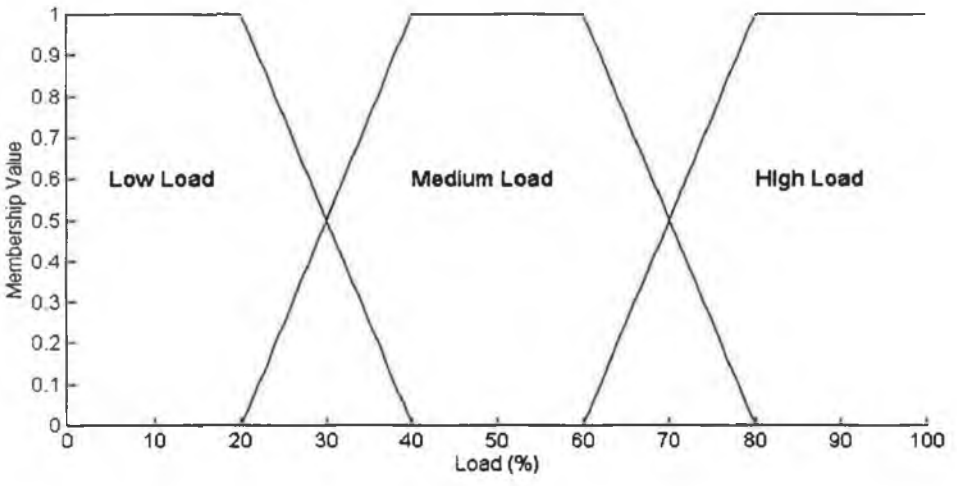


Fig. 8.7 Fuzzy Sets for Operating Point

The actual controller output is calculated as follows:

$$u_{out} = u_{10} mv_{low}(op) + u_{50} mv_{med}(op) + u_{90} mv_{high}(op)$$

where

u_{out} = actual (fuzzified) controller output
u_{10} = output of 10% linear controller
u_{50} = output of 50% linear controller
u_{90} = output of 90% linear controller
mv_{low} = membership value for low load fuzzy set
mv_{med} = membership value for medium load fuzzy set
mv_{high} = membership value for high load fuzzy set
op = current operating point

(8.26)

1. Number and Range of Fuzzy Sets

   The number of fuzzy sets required to cover the plant operating range is equal to the number of linear controllers required to ensure good control over the full operating range of the boiler.

The performance of each linear controller is largely dependent on the accuracy of its internal controller model, however. This implies that the number of linear controllers to be used is dictated by the number of linearised models required to represent the boiler process over the entire operating range. It was shown in Section 5.6.2 that three linearised models of the boiler process obtained at 10%, 50% and 90% are sufficient to represent the plant at all operating points, provided that the output of these linearised models are interpolated at intermediate operating points. For example, neither the 50% linearised model nor the 90% linearised model can accurately represent the boiler process at the 70% operating point. However, by interpolating the output of these two models, it *is* possible to accurately predict the plant behaviour at the 70% operating point. Based on the results of Section 5.6.2, it was decided to use three fuzzy sets - a low load set, a medium load set and a high load set. At low loads, the output of the 10% linearised model is predominant, at medium loads the output of the 50% linearised model is predominant and at high loads the output of the 90% linearised model is predominant.

2. Shape of Fuzzy Sets

   The shape of the fuzzy sets dictates the mechanism by which overall control of the boiler process moves between the three linear controllers. In this case, it was decided to use a "flat-topped" shape. This shape takes into account that good models are available at the 10%, 50% and 90% operating points by using only the output of the 10%, 50% and 90% controllers respectively for a small operating range (±10%) around these operating points. Between 0% and 20% the actual controller output is equal to the output of the 10% linear controller only. Between 40% and 60% the actual controller output is equal to the output of the 50% linear controller only. After 80% the controller output is equal to the output of the 90% linear controller only. At operating points outside the ±10% range of the linear controller, the fuzzy sets allow control to move in a smooth way between the three different linear controllers as necessary. Between 20% and 40%, the output of the 10% and 50% linear controllers are combined to generate the actual controller output. Between 60% and 80% the output of the 50% and 90% linear controllers are combined to generate the actual controller output.

Using this strategy it is not necessary to assume that the parameters of a linearised model change smoothly with operating point. Likewise, this strategy does not require interpolation between either the model or controller parameters which would be computationally costly given the large number of parameters which would be involved in this case. For example, the state transition matrix of the linearised model has 400 elements.

8.5 Controller Tuning

The desired response of each of the controlled variables may be specified via two tuning parameters:

1 Time Constant of the Reference Trajectory
2 Position of Coincidence Point(s)

A minimum of eight parameters must be specified for the linear controller at each operating point i.e. the reference trajectory time response and at least one coincidence point for each of the four controlled variables. In all, a minimum of 24 tuning parameters must be specified (given 3 operating points, 4 controlled variables and 2 tuning parameters per controlled variable.)

It may not be possible to use the same set of tuning parameters at the 10%, 50% and 90% operating point due to the variation in boiler dynamics with operating point. The method used to select the tuning parameters at the 10%, 50% and 90% operating points is described in separate sections.

8.5.1 Selection of Tuning Parameters for 90% Linear Controller

The time constant of the controlled variables could be used as a basis for choosing the controller tuning parameters. However, due to the presence of an integrator, it is not possible to determine the time constant of the controlled variables from their step responses. An alternative procedure was devised to find the dominant time constant of each of the controlled variables, given in a change in its associated manipulated variable. (The coupling between the manipulated and controlled variables is shown in Table 8.1).

1. The linearised models are converted into Jordan form.
2. Any integrators are removed from the resulting Jordan form model.
3. The new model is reduced to a single state, linear model using balanced truncation model reduction. The time constant of the first order reduced model of each controlled variable is presented in Table. 8.4

Controlled Variable-Model Output	Manipulated Variable - Model Input	Model Time Constant (s)
Main Steam Pressure	Fuel Flow Rate	361
Drum Level	Feedwater Flow Rate	361
Main Steam Temperature	Attemperating Water Flow Rate	5.6
$\%O_2$ in Stack Gases	Air Flow Rate	1.8

Table 8.4 Time Constant of First Order Reduced Models

4. Time constant of reference trajectory is set equal to half the time constant of the model.
5. Controller performance is tested by simulation.
6. Initial choice of controller tuning parameters is improved by trial and error, to yield the default set of controller tuning parameters which are presented in Table 8.5.

Controlled Variable	Time Constant of Reference Trajectory (s)	Position of Coincidence Point (s)
Main Steam Pressure	24	12
Main Steam Temperature	30	15
Drum Level	0.6	0.3
$\%O_2$ in Stack Gases	3.6	1.8

Table 8.5 Default Set of Tuning Parameters

In the case of each of the controlled variables, it can be seen that the position of the coincidence point is equal to half the value of the time constant of the reference trajectory. It can also be noted that it is possible to increase or decrease either the position of the coincidence point or the reference trajectory time constant by a third and that the result is an integer.

The effect of the controller tuning parameters on system response is investigated via a series of step response tests using the following combinations of the controller tuning parameters.

Test 1 Use default parameters

Test 2 Increase default time constant only by 33%

Test 3 Increase default coincidence point only by 33%

Test 4 Increase both default time constant and coincidence point by 33%

Test 5 Decrease both default time constant and coincidence point by 33%

A change in any one tuning parameter has some effect on all four controlled variables, owing to the multivariable nature of the boiler process. For example, a change in the setpoint of main steam pressure causes the other controlled variables to deviate from their setpoints. To illustrate the nature of this interaction, the response of all the controlled and manipulated variables to a step change in the setpoint of main steam pressure is shown in Fig. 8.8. The controlled variables are shown in the left hand column and the manipulated variables in the right hand column.

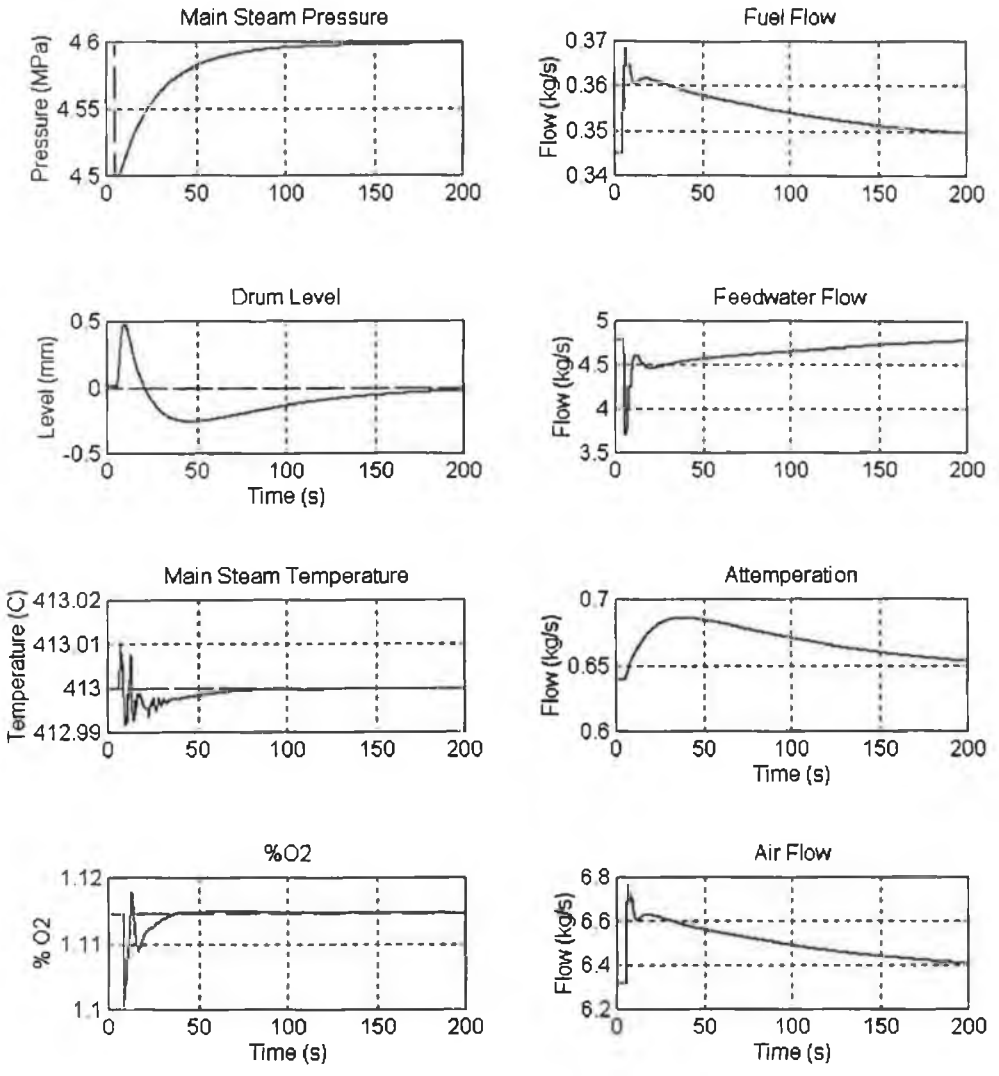


Fig. 8.8 System Response to a Step Change in Main Steam Pressure Setpoint

It can be seen from Fig. 8.8 that the fuel flow rate must be increased to order to increase main steam pressure from its original level of 4.5 MPa to the new setpoint level of 4.6 MPa. This increase in fuel flow causes the density of the water in the drum and risers to reduce and as a result causes drum water level to increase. Feedwater flow is initially reduced from 4.75 kg/s to 3.75 kg/s in order to prevent a transient increase in drum water level and is then increased to cope with the increased rate of evaporation. Attemperation is increased as the increased fuel flow causes the temperature of the main steam temperature to rise. Air flow must also be varied in order to maintain the correct level of percentage O_2 in the stack gases.

The magnitude of the disturbance to drum water level, main steam temperature and percentage O_2 in the stack gases is dependent on the magnitude of the main steam pressure setpoint change and the desired response time of main steam pressure. In order to obtain good overall control

performance for the entire system, it is necessary to consider the interactions between the controlled variables. However, an initial tuning can first be carried out separately for each of the controlled variables. After this initial tuning is completed, the overall system response is examined and further tuning may be carried out which takes into account the interaction between the controlled variables.

The tuning tests are shown for each of the controlled variables in turn. For clarity, the effect of the controller tuning parameters for each controlled variable are shown for that controlled variable only. Likewise, only the response of the manipulated variable which has the greatest effect on the controlled variable is included in the set of results for that controlled variable.

In order to perform a quantitative analysis of the tuning results, the following three indices are defined.

1. The 63% response time (time required for the controlled variable to reach 63% of its final value) has been chosen as an index of system response time. This index has been chosen because it allows the actual 63% response time of the plant to be directly compared with the desired 63% response time of the plant. The desired 63% response time of the plant is equal to the first controller tuning parameter i.e. time constant of reference trajectory.

2. The second index is a measure of controller action. It is equal to the maximum overshoot or undershoot of the manipulated variable measured in relation to the final steady state value of the manipulated variable.

$$OS_{max} = \max\left(\left|u - u_{ss}\right|\right)$$

where u = manipulated variable (8.27)

u_{ss} = Final steady state value of manipulated variable

3. The third index is also a measure of controller energy. It is defined as the sum of the square of the deviation of the manipulated variable from its mean value, calculated over a defined time-span. This index is referred to as the Integral Squared Control Action *(ISU)*.

$$ISU = \sum_{k=k_{min}}^{k=k_{max}} u(k) - \bar{u})^2$$

where (8.28)

$$\bar{u} = \frac{\sum_{k=k_{min}}^{k=k_{max}} u(k)}{k_{max} - k_{min} + 1}$$

Main Steam Pressure

Fig. 8.9 shows the response of main steam pressure to a step change in set-point for each of the five sets of tuning parameters. The corresponding control action for fuel flow is shown in Fig. 8.10.

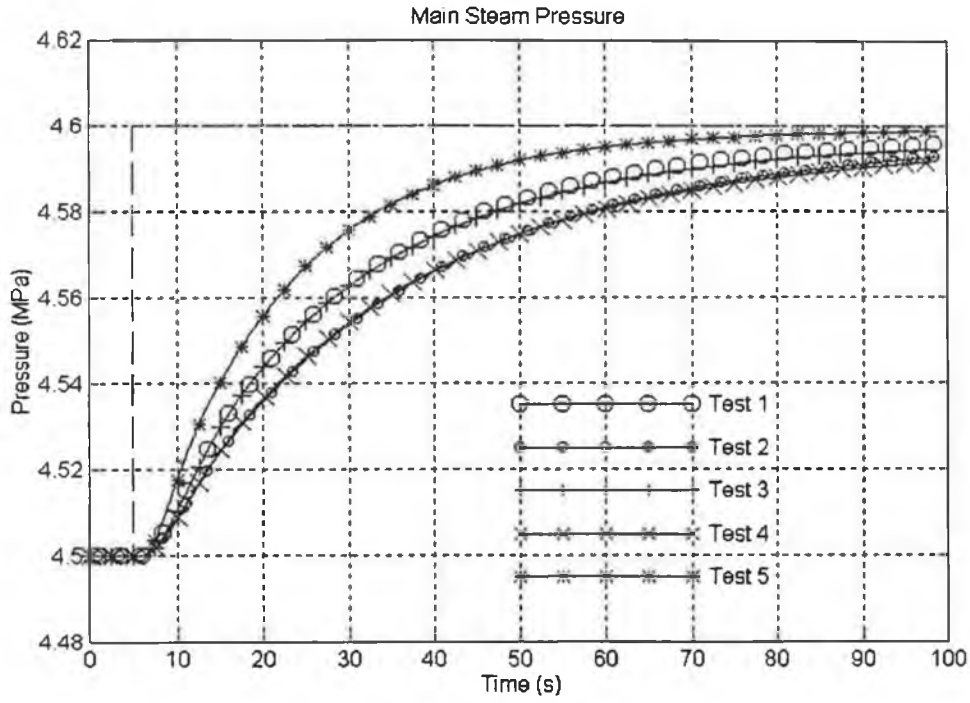


Fig. 8.9 Response of Main Steam Pressure to Step Change in Setpoint

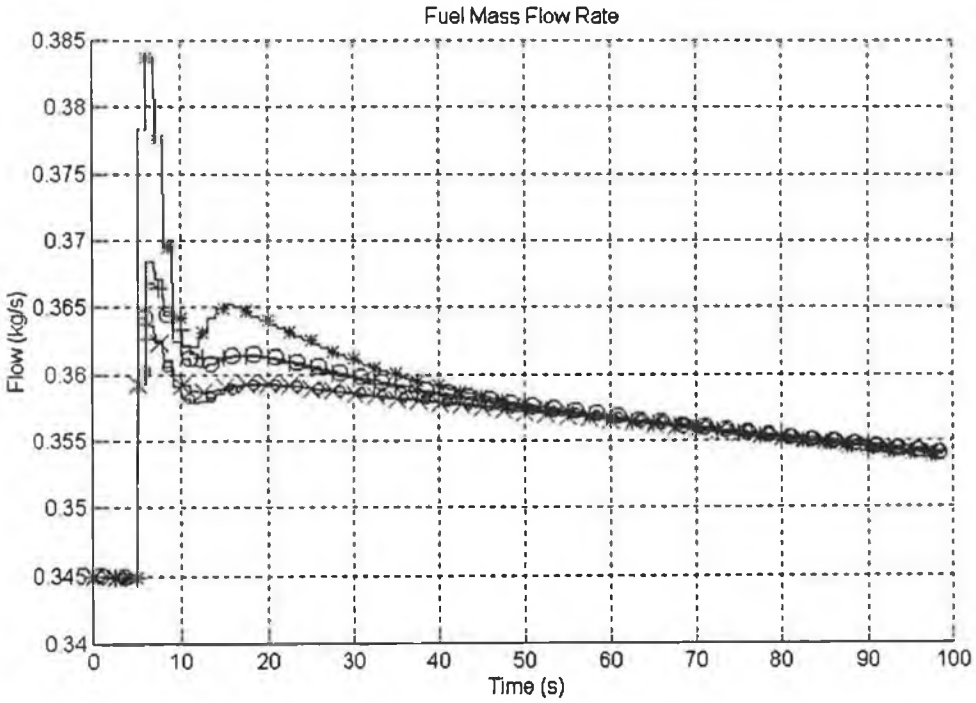


Fig. 8.10 Response of Fuel Flow to Step Change in Setpoint

It can be seen from Fig. 8.9 and Fig. 8.10 that the system responses may be divided into three groups. The fastest group contains the response for Test 5, which as might be expected, specified the shortest time response. The next fastest group contains the responses for Tests 1 and 3, both of which specified the same response time. The third and slowest group contains the results for Tests 2 and 4, both of which specified the slowest response time. This grouping of the system responses suggests that the position of the coincidence point has little effect on the system response.

Table 8.6 presents the measured indices for each set of controller tuning parameters. Shading is used to indicate a non-default parameter value.

Tuning for Main Steam Pressure Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Fuel Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	24	12	25.28	1.88e-2	3.1e-2
2	32	12	32.29	1.38e-2	2.0e-2
3	24	16	25.36	1.72e-2	3.0e-2
4	32	16	32.00	1.28e-2	2.0e-2
5	16	8	18.05	3.43e-2	5.7e-2

Table 8.6 Effect of Main Steam Pressure Controller Tuning Parameters

Test 1 This test demonstrates that it is possible to specify the plant response time quite accurately using the tuning parameters. There is a close correlation between the reference trajectory time constant (24s) and the 63% response time of the controlled variable (25.28s).

Test 2 The reference trajectory time constant is increased to 32s, causing the fuel flow overshoot to decrease by 26%, *ISU* to decrease by 36% and the plant response time to increase to 32.29s. Again a very close correlation exists between the desired 63% response time and the actual 63% response time.

Test 3 The coincidence point position is changed from 12s to 16s. This should increase the plant response time and decrease controller action. In fact, the increase in plant response time is less than 0.1s even though fuel flow overshoot decreases by 8% and the *ISU*

index decreases by 3.2%. This indicates that increasing the coincidence point can decrease high frequency controller action.

Test 4 The reference trajectory time constant is increased to 32s and the coincidence point is moved to 16s. A comparison with Test 2 is of interest here. It is to be expected that the plant response time would be slower than that achieved in Test 2. However, plant response time is very slightly faster (0.29s) and more importantly fuel flow overshoot is 7% smaller than for Test 2. Again it appears that increasing the coincidence point can decrease high frequency controller activity without compromising plant response time.

Test 5 The reference trajectory time response is decreased to 16s and the coincidence point is placed at 8s. The plant response time decreases to 18s. The actual plant response time is 2s longer than the desired plant response time, indicating that the plant response is now nearly as fast as possible. This is confirmed by a large increase in fuel flow overshoot (83% increase) and *ISU* index (82% increase). .

The results demonstrate that it is possible to specify a desired response time for main steam pressure with a very good degree of accuracy. The degree of accuracy decreases as the desired response time decreases and the minimum response time possible is approached. The results also demonstrate that the reference trajectory time constant has a much more dominant effect on system response than the position of the coincidence point. The position of the coincidence point can be used to increase or decrease high frequency controller activity.

The choice of controller tuning parameters is dependent on the performance specification for the controlled system. If a fast response is required, the parameters used in Test 5 could be adopted. Otherwise, the parameters used in Test 3 offer a good compromise between controller action and system response.

Drum Level

The response of drum level to a step change in setpoint is presented in Fig. 8.11 The corresponding controller action - feedwater flow rate - is presented in Fig. 8.12.

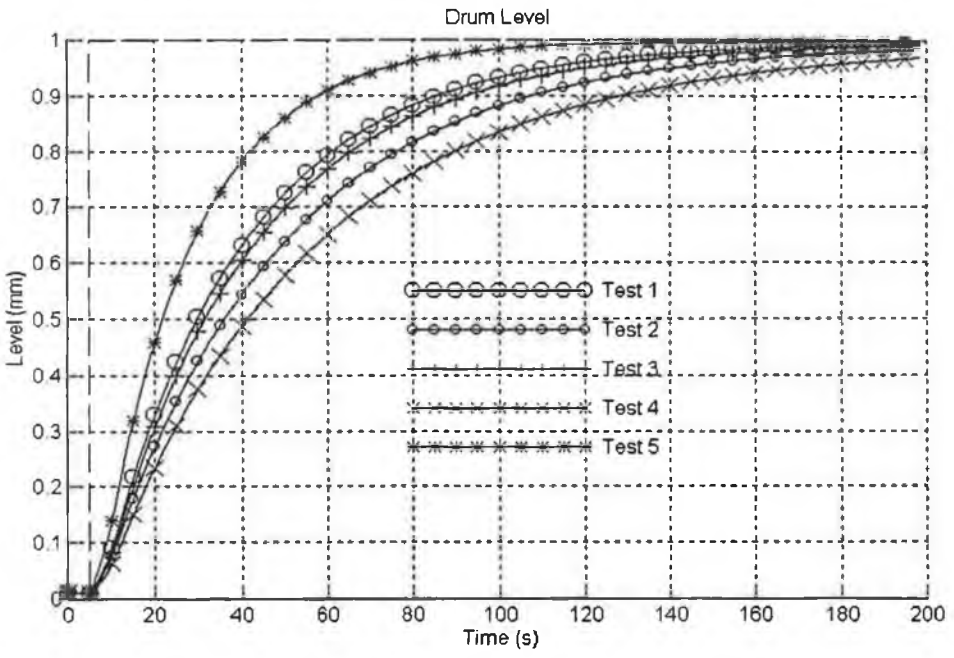


Fig. 8.11 Response of Drum Level to Step Change in Setpoint

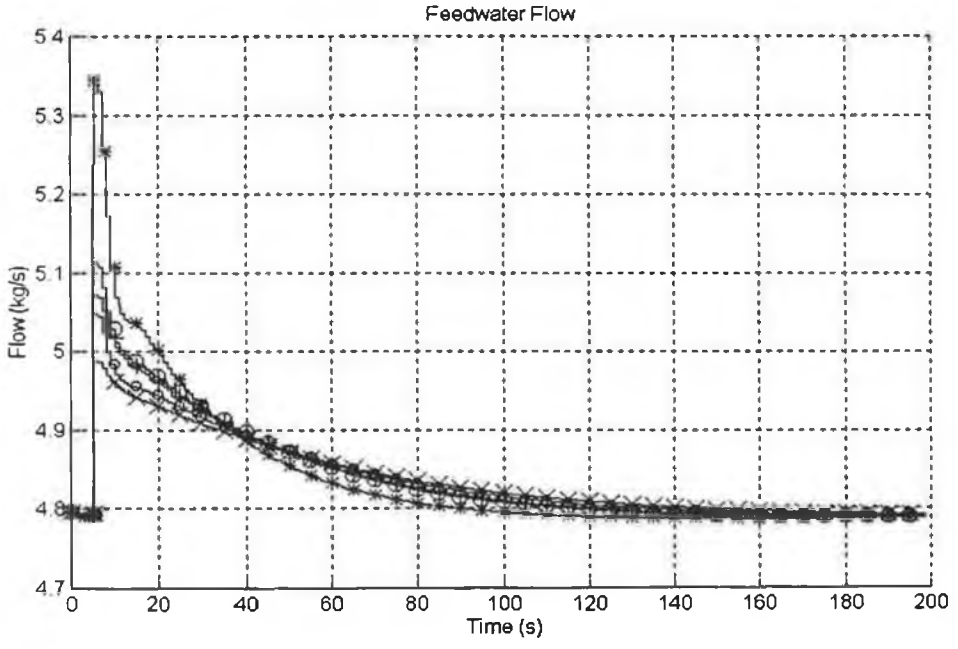


Fig. 8.12 Response of Feedwater Flow to Step Change in Setpoint

The responses for each of the five tests can be seen distinctly in Fig. 8.11 i.e. they are not grouped. This suggests that the position of the coincidence point has a greater effect in tuning drum level control than it has in tuning main steam pressure controller. The relationship between the controller tuning parameters and system response is presented in Table. 8.7.

Tuning for Drum Level Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Feedwater Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	30	15	35.00	0.324	9.2
2	40	15	44.14	0.257	6.3
3	30	20	37.60	0.283	8.1
4	40	20	51.93	0.195	4.7
5	20	10	23.42	0.557	17.1

Table 8.7 Effect of Drum Level Controller Tuning Parameters

Test 1 A reasonable correlation exists between the reference trajectory time constant (30s) and the 63% response time of the controlled variable (35s).

Test 2 The reference trajectory time constant is increased to 40s, causing the plant response time to increase to 44s, the *ISU* index to decrease by 32% and the feedwater flow overshoot to decrease by 20%.

Test 3 The coincidence point position is changed from 15s to 20s. As a result, the feedwater flow overshoot and the *ISU* index both decreases by 12% and the drum level response time increases by approximately 2.5s.

Test 4 The reference trajectory time constant is increased to 40s and the coincidence point is moved to 20s. The plant response time (52s) now becomes considerably slower than the reference trajectory time constant and the feedwater flow overshoot decreases by 79%. This suggests that the position of the coincidence point may have an important effect on system response in certain situations. One possible reason for this is that the accuracy of the prediction decreases as the predictive horizon decreases. A 20 step ahead prediction horizon may be too long for accurate control.

Test 5 The reference trajectory time response is decreased to 20s and the coincidence point is placed at 10s. This causes the feedwater flow overshoot to increase by 72%, the *ISU* index to increase by 85% and the plant response time to decrease to 23s.

As for pressure, the relationship between reference trajectory time constant and plant time response is quite close for drum level. Reference trajectory time constant is also the dominant tuning parameter but coincidence point position can be seen to have an effect, particularly if the desired time response is slow and the prediction horizon is very long.

The parameters used in Test 5 generate the fastest system response and result in the greatest controller activity. The parameter set used in Test 3 is more attractive - system response is slightly slower but controller action has been reduced considerably.

Main Steam Temperature

The system response of main steam temperature and the corresponding attemperating flow rate are shown in Fig. 8.13 and Fig. 8.14 respectively, for each combination of tuning parameter.

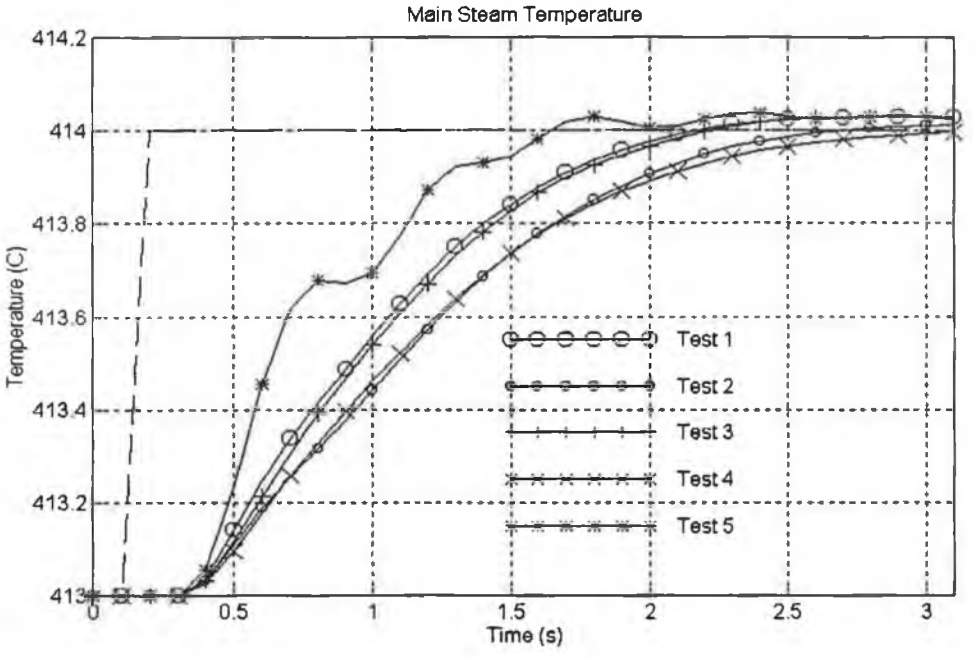


Fig. 8.13 Response of Main Steam Temperature to Step Change in Setpoint

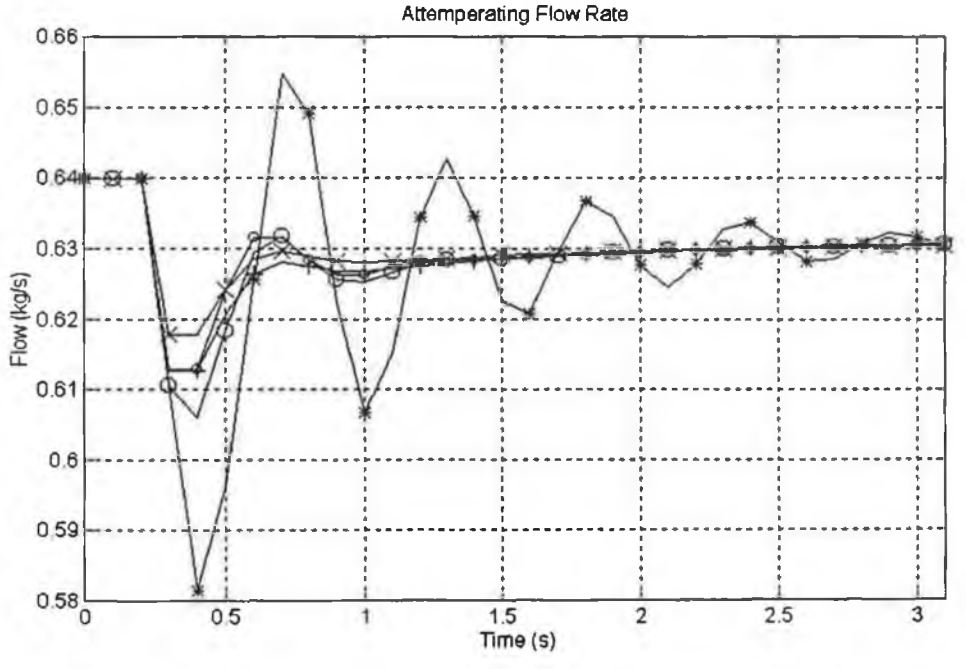


Fig. 8.14 Response of Attemperating Flow Rate to Step Change in Setpoint

As for main steam pressure, the five system responses can be separated into three separate groups - fast, medium, slow. The fastest response is noticeable for being extremely oscillatory. The performance indices for each set of controller tuning parameters are presented in Table 8.8.

Tuning for Main Steam Temperature Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Attemperating Flow Undershoot (kg/s)	ISU $(kg/s)^2$
1	0.6	0.3	1.00	2.80e-2	2.6e-3
2	0.8	0.3	1.20	2.11e-2	2.0e-3
3	0.6	0.4	1.04	2.13e-2	2.2e-3
4	0.8	0.4	1.19	1.61e-2	1.6e-3
5	0.4	0.2	0.62	5.25e-2	7.8e-3

Table 8.8 Effect of Main Steam Temperature Controller Tuning Parameters

Test 1 A response time of 1.0s is obtained using a desired response time of 0.6s.

Test 2 The desired response time is increased by 0.2s, causing the actual response time to increase by 0.2s, attemperating flow undershoot to decrease by 24% and the *ISU* index to decrease by 30%..

Test 3 The coincidence point is increased by 0.2s. This has very little effect on the actual response time but attemperating flow undershoot decreases by 23%.

Test 4 The default desired response time and coincidence point position are increased by 33%. A comparison with Test 2 shows a very similar response time, but the attemperating flow undershoot has decreased by 23%.

Test 5 Both the default desired response time and coincidence point position are decreased by 33%. Again the actual response time is approximately 0.2s greater than the desired response time. Attemperating flow undershoot increases by 87% and *ISU* index increases by 200%.

The results confirm that system response is dictated by the time constant of the reference trajectory. Likewise high frequency controller action has been decreased by increasing the position of the coincidence point.

Explanation of Oscillatory Control Action in Test 5

Selection of the best set of tuning parameters is immediately simplified by eliminating the parameters used in Test 5 which result in an oscillatory controller action. The reason for this oscillatory controller action can be understood by carrying out a one-step-ahead prediction of main steam temperature while using a square-wave type attemperation input. All other boiler inputs are held constant. The frequency of the square wave is first set equal to 0.5 Hz. The resulting plant response and one-step-ahead prediction are plotted in Fig. 8.15.

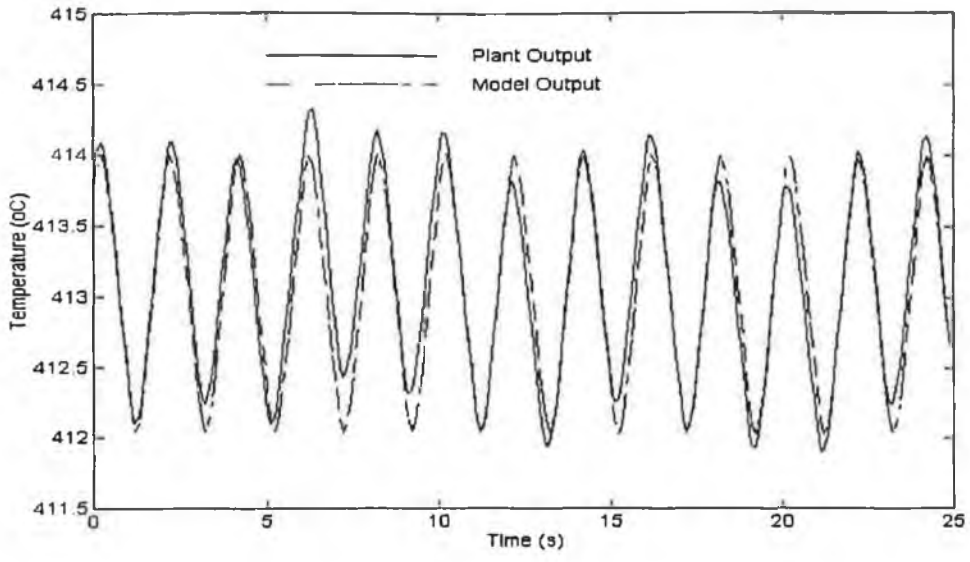


Fig. 8.15 Comparison and Plant and Model Output Using 0.5 Hz Attemperation Input Signal

It can be seen from Fig. 8.15 that the model output is almost exactly equal to the plant output, confirming that the model is capable of predicting the plant output reasonably accurately given this type of input signal.

The frequency of the square wave is increased by a factor of 10 to 5 Hz. The resulting plant response and the one-step prediction are plotted in Fig. 8.16.

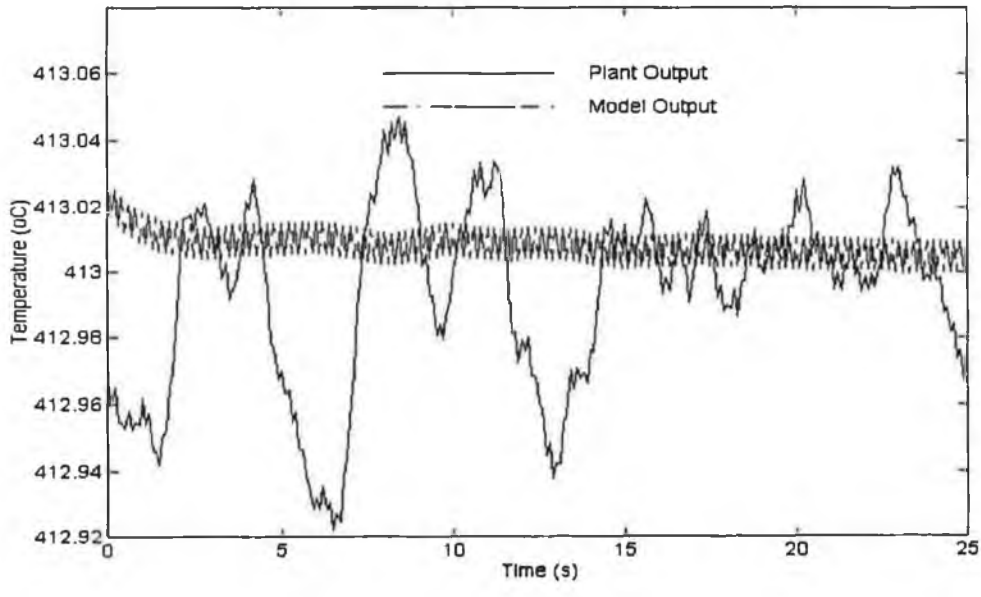


Fig. 8.16 Comparison and Plant and Model Output Using 5 Hz Attemperation Input Signal

It can now be seen that the accuracy of the model has deteriorated and that the model greatly underestimates the plant output. This explains the appearance of oscillations in the controller output when the desired response time of the plant is decreased to 0.4s At each sampling interval,

the predictive controller determines the control action required to move the predicted plant output to the reference trajectory. As the desired plant response time decreases, the frequency of the controller action increases. However for high frequency control action, the internal controller model underestimates the plant response and consequently the proposed control action causes the plant output to overshoot the reference trajectory.

It is likely that the deterioration in model performance when the frequency of the control signal increases could be prevented by specifying a shorter sampling interval for the controller. This hypothesis can be tested by repeating the previous test but using the continuous linearised superheater model to predict the plant output instead of the discretised linearised superheater model. The plant response and the one-step-ahead prediction obtained using a continuous linearised model of the superheater are plotted in Fig. 8.17.

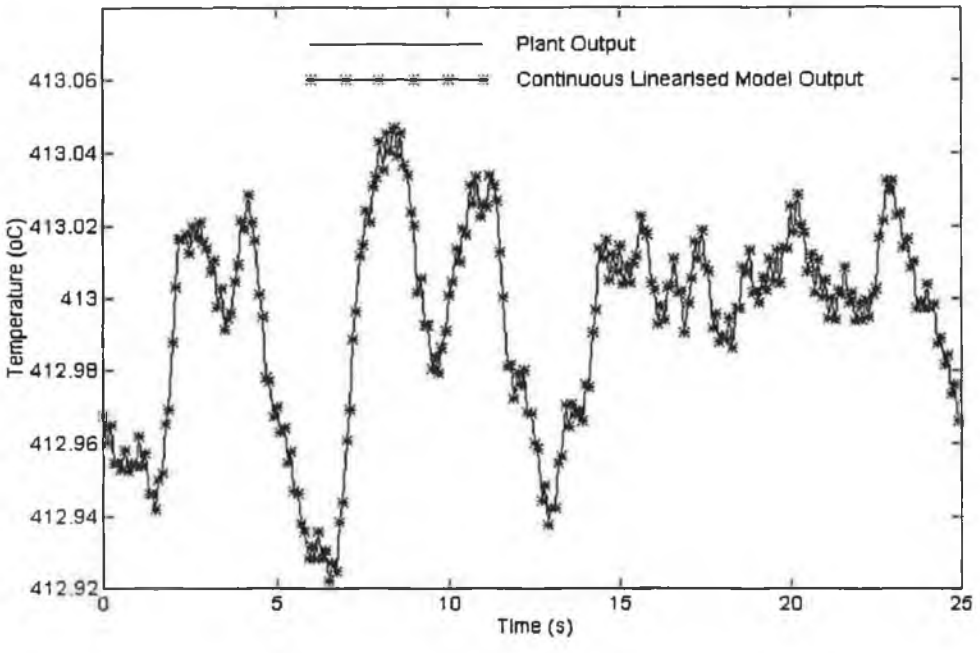


Fig. 8.17 Comparison and Plant and Continuous Model Output Using 5 Hz Attemperation Input Signal

As expected the continuous linear model can predict the plant response far more accurately than the discretised linear model. It follows that the performance of the discretised linear model could be improved by reducing the sampling period. This is not felt to be necessary in this instance however as the specified sampling period of 0.1s is adequate for a wide range of controller tuning parameters.

The parameter set of Test 3 is attractive as it offers a good compromise between control action and system response. There is little benefit to be gained in accepting the slower response obtained in Test 2 and Test 4 as controller action has not decreased significantly.

Percentage O_2 in the Stack Gases

The step response of the measured percentage O_2 in the stack gases is shown in Fig. 8.18 for each combination of the controller tuning parameters. The corresponding air flow rate signal is shown in Fig. 8.19.

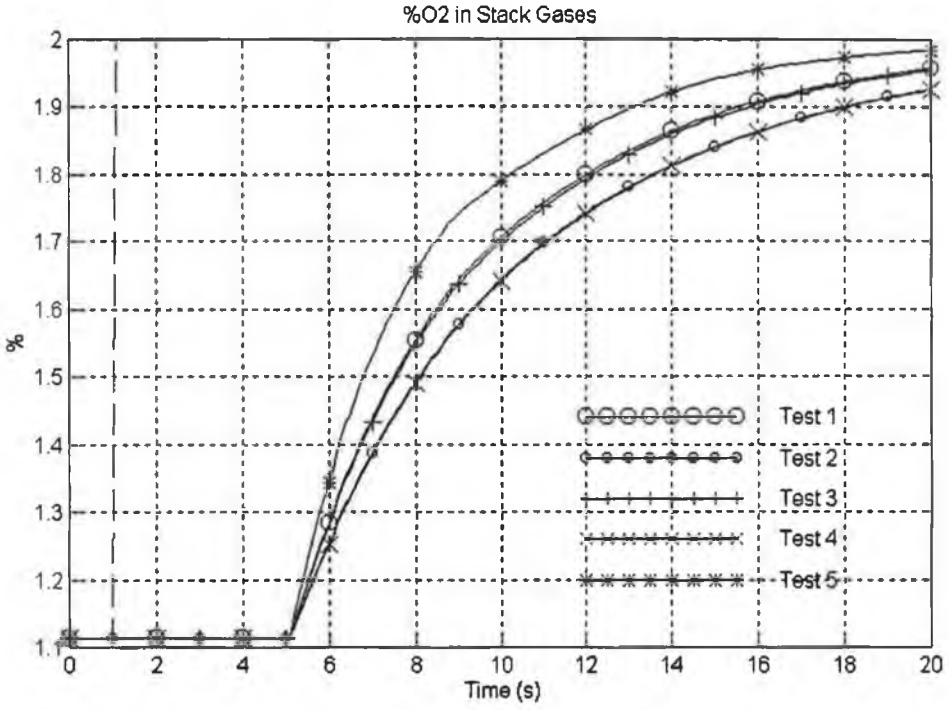


Fig. 8.18 Response of Percentage O_2 in Stack Gases to Step Change in Setpoint

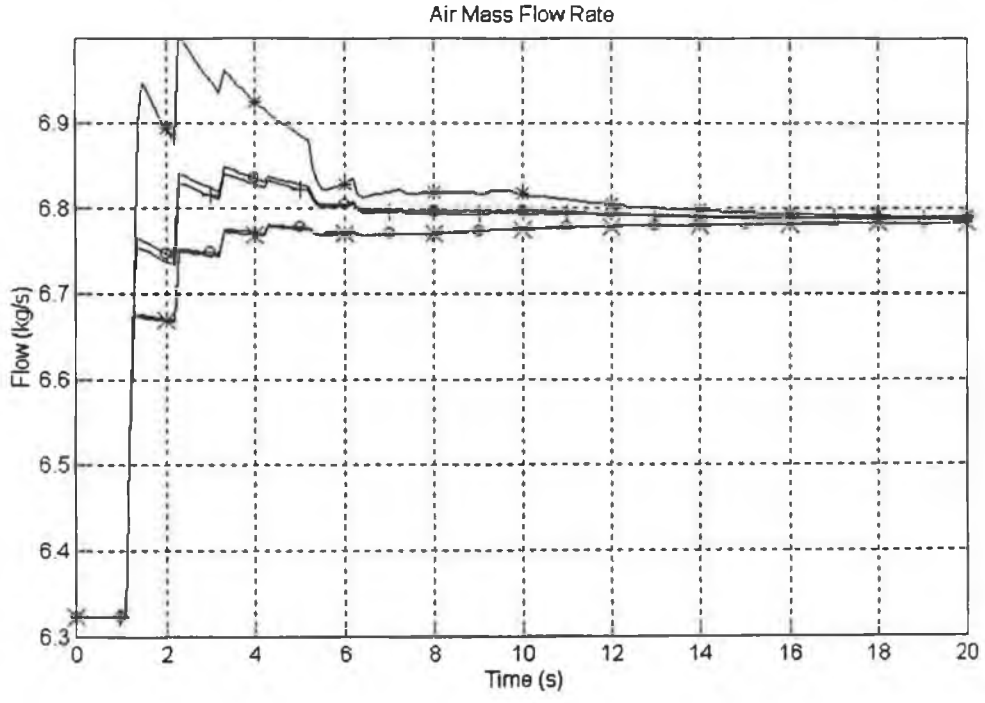


Fig. 8.19 Response of Air Flow Rate to Step Change in Setpoint

As for main steam pressure and main steam temperature, the five different system responses are separable into three distinct groups of fast, medium and slow responses. The performance indices for each set of controller tuning parameters are presented in Table 8.9

Tuning for Percentage O_2 Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Feedwater Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	3.6	1.8	8.43	9.27e-2	8.2e-1
2	4.8	1.8	9.49	2.92e-2	7.4e-1
3	3.6	2.4	8.54	8.45e-2	7.9e-1
4	4.8	2.4	9.54	2.91-2	7.4e-1
5	2.4	1.2	7.17	2.45e-1	17.7e-1

Table 8.9 Effect of Percentage O_2 Controller Tuning Parameters

Test 1 A response time of 8.43s is obtained using a desired response time of 3.6s. The large discrepancy between desired and actual response time is due to a delay of approximately 4s in percentage O_2 measurement. The response time excluding delay is approximately 4.43s.

Test 2 The desired response time is increased by 1.2s, causing the actual response time to increase by approximately 1s and air flow overshoot to decrease by 68%.

Test 3 The coincidence point is increased by 1.2s. This has practically no effect on the actual response time but it causes air flow overshoot to decrease by 8%.

Test 4 The default desired response time and coincidence point position are increased by 33%. A comparison with Test 2 shows practically no change in either response time or controller action.

Test 5 Both the default desired response time and coincidence point position are decreased by 33%. The response time decreases by 1.26s, the air flow overshoot increases by 164% and the *ISU* index by 115%.

The results for percentage O_2 controller tuning follow the same pattern as the results for the other three controlled variables. The plant time response may be accurately specified *via* the reference trajectory time constant, whereas coincidence point position mainly effects the magnitude of the initial controller action.

In the first four tests the controller action approaches an open-loop response. The parameter set used in Test 5 is chosen because it yields a faster system response.

Analysis of Results

The tests confirm that the controller tuning parameters are an effective and accurate means of specifying the response time of the controlled variables. In particular, the response time of the controlled variable is closely related to the response time of the reference trajectory. Specifying a slower reference trajectory response time reduces controller action and consequently increases the system response time.

Controller action is also dictated by the position of the coincidence point. Placing a coincidence point at the very start of the reference trajectory increases the controller action which occurs after a setpoint change. This can be understood by considering predictive control of a first order process. The control action required immediately after a step change in setpoint is compared for two different sets of tuning parameters. The time constant of the reference trajectory is the same for both sets of tuning parameters but coincidence point position is different.

The process and the two sets of controller tuning parameters are defined as follows:

Process: Time Constant, τ_p =11s
Steady State Gain, k_p=1

Controller 1: Time Constant of Reference Trajectory, τ_r=10s
Position of Coincidence Point, H_1=1s
Sampling Period, TS=1s

Controller 2: Time Constant of Reference Trajectory, τ_r=10s
Position of Coincidence Point, H_2=15s
Sampling Period, TS=1s

In both cases, the controller objective is to make the predicted plant output coincide with the reference trajectory at the coincidence point, by specifying a control action which is assumed to be constant at future sampling periods.

If the initial process output is equal to 0, the desired value of the process output at the coincidence point is equal to:

$$y_{r_1} = S(1-e^{-\frac{H_1 TS}{\tau_r}}) \qquad (8.29)$$

where

y_{r_1} = Value of reference trajectory at t = H_1

S = Setpoint

The predicted value of the process at the first coincidence point is calculated by the internal controller model to be:

$$y_{p_1} = u_1 k_p (1 - e^{-\frac{H_1 TS}{\tau_r}})$$

where (8.30)

y_{p_1} = Predicted value of process at t = H_1

u_1 = Control action

The control action necessary to ensure that the predicted value of the process and the desired value of the process are equal at the coincidence point. is determined by equating equation (8.29) and equation (8.30). The resulting control action is:

$$u_1 = \frac{S(1 - e^{-\frac{H_1 TS}{\tau_r}})}{k_p (1 - e^{-\frac{H_1 TS}{\tau_p}})} \tag{8.31}$$

Using the defined process and controller parameters and a setpoint of 1, the required controlled action is equal to 1.095. The second controller uses the same method to calculate control action and determines that a control action of 1.044 is required. Moving the coincidence point position from 1s to 15s has reduced initial controller action by 4.7%.

The effect of coincidence point on controller action is presented graphically in Fig. 8.20. The desired reference trajectory of the process is shown for a step change in setpoint. Also shown are two predicted process trajectories. The first predicted trajectory coincides with the reference trajectory at the near coincidence point. The second predicted trajectory coincides with the reference trajectory at the far coincidence point.

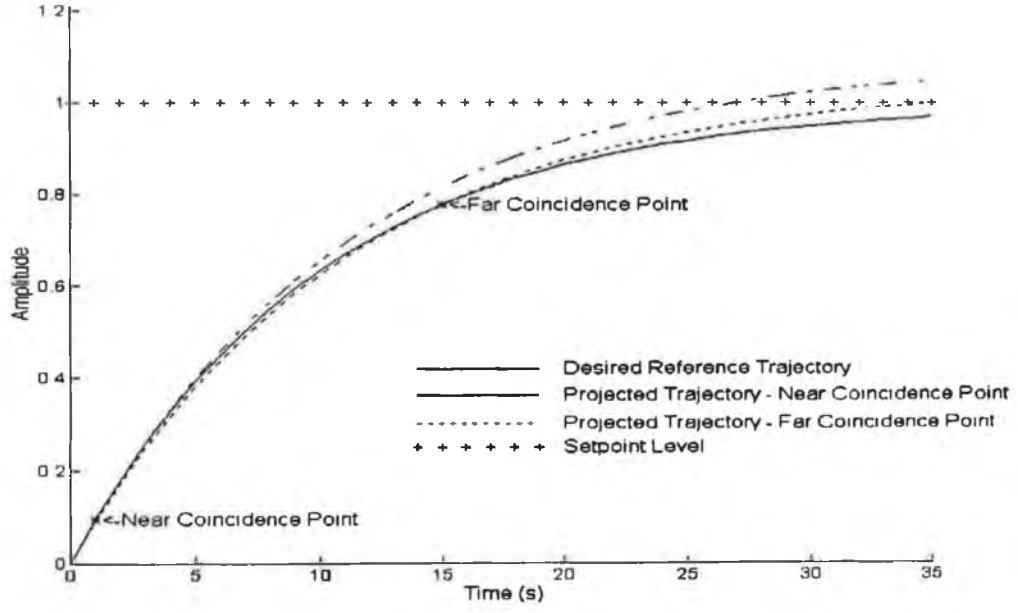


Fig. 8.20 Predicted Trajectories for Two Different Coincidence Points and Reference Trajectory

It can be seen that the step response passing through the near coincidence point has a greater steady state amplitude than the step response passing through the far coincidence point. It follows that initial control action decreases as the position of the coincidence point moves further into the future.

The effect of coincidence point on initial controller action is affected to some extent by the time constant of the reference trajectory. The position of the coincidence point becomes more critical as the time constant of the reference trajectory decreases. This is illustrated by changing the time constant of the reference trajectory for both of the controllers described above. The process and controllers are now defined as:

Process: Time Constant, τ_p =11s
Steady State Gain, k_p=1

Controller 1: Time Constant of Reference Trajectory, τ_r=5s
Position of Coincidence Point, H_1=1s
Sampling Period, TS=1s

Controller 2: Time Constant of Reference Trajectory, τ_r=5s
Position of Coincidence Point, H_2=15s
Sampling Period, TS=1s

In this case, the required control action calculated by the first controller after a step-change in setpoint is 2.086. The corresponding control action calculated by the second controller is 1.277. Moving the coincidence point position from 1s to 15s has now reduced initial controller action by 39%.

An examination of the test results highlighted particular parameter settings which can achieve a good compromise between system response and controller action. This "improved" set of tuning parameters is listed in Table 8.10.

Controlled Variable	Test Nr.	Time Constant of Reference Trajectory (s)	Position of Coincidence Point (s)
Main Steam Pressure (for slow response)	3	24	16
Main Steam Pressure (for fast response)	5	16	8
Main Steam Temperature	3	0.6	0.4
Drum Level	3	30	20
% O_2 in Stack Gases	5	2.4	1.2

Table 8.10 Improved Set of Tuning Parameters

Controller Tuning for Load Disturbances

The control strategy is required to provide a good response to load disturbances (i.e. variations in steam demand) as well as to setpoint changes. Consequently, the final choice of controller tuning parameters must yield a good system response in the case of a set-point change or in the case of a load disturbance. Both the default and improved set of tuning parameters have been shown to provide good response in the case of set-point changes. The relationship between controller tuning parameters and system response in the case of a disturbance is considered for two different scenarios:

1. System response in the case of modelled disturbances

   The controller calculates the required control action necessary to place the predicted model output on the reference trajectory. If the model is exact, the plant output returns to the setpoint along the desired reference trajectory specified by the controller.

2. System response in the case of unmodelled disturbances

   The controller again calculates the control action necessary to place the predicted model output on the reference trajectory. However, unmodelled disturbances cause the controller to underestimate or overestimate the control action required to place the plant output on the reference trajectory. As a result the plant does not return to the setpoint along the desired reference trajectory. The greater the effect of the unmodelled disturbances on the system output, the smaller the effect of the controlled tuning parameters on the system response. If the internal controller model is extremely inaccurate, it may be preferable to use a control strategy which is not model-based.

The internal controller models for each of the three controllers include important disturbance models including load disturbance models. As a result, it is to be expected that the system response in the case of a step load disturbance should be similar to the system response in the case of a setpoint change. The controller performance for step load disturbances is investigated for the following combinations of controller tuning parameters:

Set 1. Default set of parameter settings
Set 2. Improved set of parameter settings with main steam pressure tuned for slow response
Set 3. Improved set of parameter settings with main steam pressure tuned for fast response

The effect of these three sets of tuning parameters on the system response is investigated by applying a step increase in plant load from 90% to 95% and simulating the system response for

each of the three parameters sets. For each parameter set the response is reasonable and quite similar. The system response using Set 2 is shown in Fig. 8.21. The controlled variables are arranged in the left hand column and the manipulated variables in the right hand column.

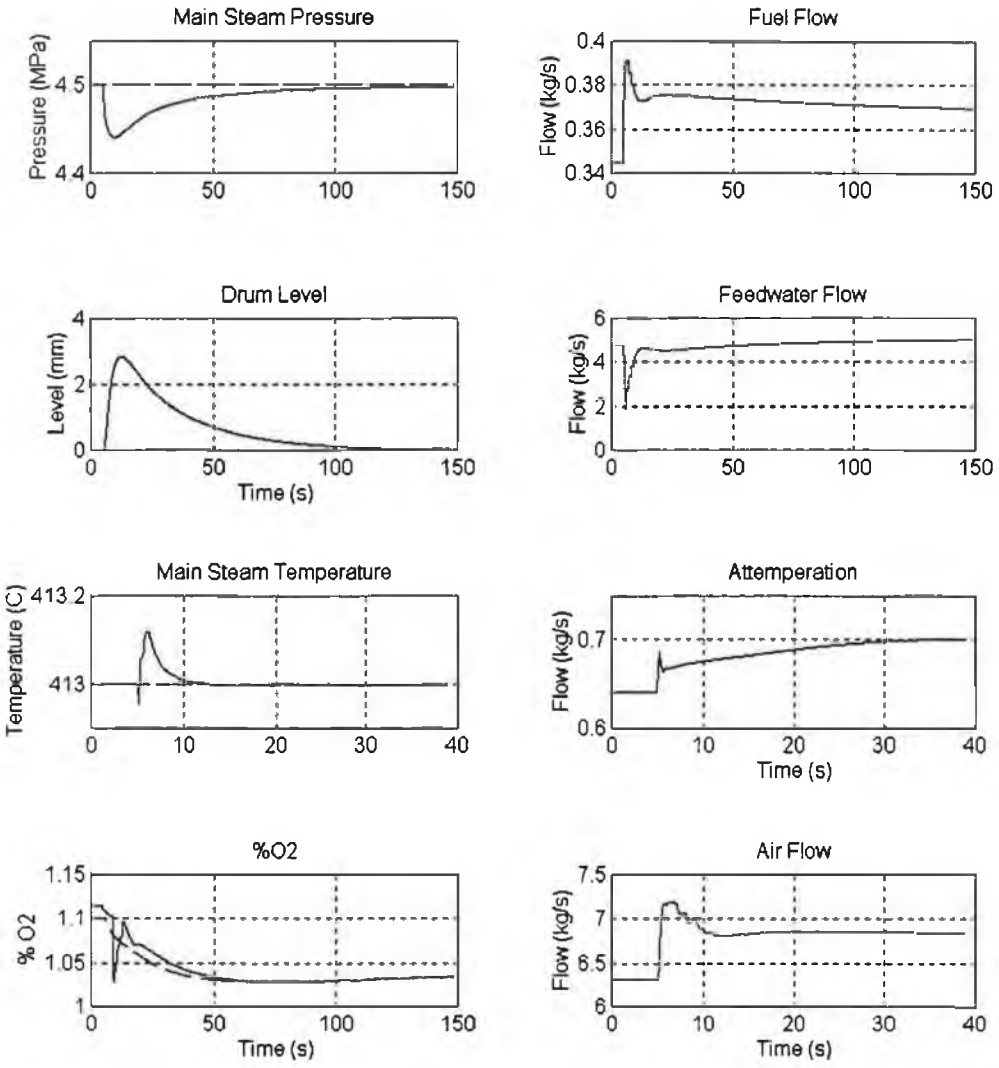


Fig. 8.21 System Response to a 5% Step Change in Load at 90% Load

In the case of main steam pressure, main steam temperature and drum level, it can seen from Fig. 8.21 that the controlled variable returns to the setpoint along an approximately exponential reference trajectory. (The system response of percentage O_2 in the stack gases is complicated by the variation in setpoint which also occurs after the load disturbance). The performance indices of main steam pressure are presented in Tables 8.11 for each of the parameter settings. The 63% response time is calculated from the time that main steam pressure reaches its maximum deviation from the setpoint.

Main Steam Pressure Tuning for Load Disturbances					
Controller Tuning Parameters			Performance Indices		
Tuning Parameter Set	**Time Constant of Reference Trajectory (s)**	**Position of Coincidence Point (s)**	**Actual 63% Response Time (s)**	**Fuel Flow Overshoot (kg/s)**	**ISU $(kg/s)^2$**
Default	24	12	23.57	3.62e-2	9.8e-3
Improved (Fast Response)	16	8	17.62	5.35e-2	14.4e-3
Improved (Slow Response)	24	16	22.99	2.28e-2	10.0e-3

Table 8.11 Effect of Main Steam Pressure Tuning Parameters For Load Disturbances

1. The 63% response time using the default parameters is 23.57s. This is very close to the desired 63% response time of 24s and very similar to the 63% response time of 25.28s that was obtained using the default parameters for a step change in setpoint.
2. A 63% response time of 17.62s is obtained when a desired response time of 16s and a coincidence point of 8s is specified. Controller action increases significantly, fuel flow overshoot increases by 48%.
3. The default time constant is used but the position of the coincidence point is increased by 8s. As might be expected, this causes the fuel flow overshoot and the *ISU* of the control signal to decrease. Despite this the 63% response time of main steam pressure is slightly shorter. A comparison of the main steam pressure response obtained using the default parameters and the improved parameters for a slow response shows that in fact a faster response is obtained using the default parameters. It is not possible to rigorously compare the 63% response time obtained for a load disturbance using two different tuning parameters because the maximum deviation from the setpoint varies as does the time at which main steam pressure reaches its maximum deviation from the setpoint. For a true comparison of the performance of the controller using different parameters settings, it would be preferable to measure the time required for main steam pressure to reach a prespecified value. In this case however, the 63% response time is of more interest however as it allows a direct comparison with the results obtained for setpoint step changes.

The results confirm that the controller tuning parameters may be used to specify plant response with reasonable accuracy, both in the case of a setpoint change or a load disturbance.

Interaction Between Controlled Variables

The interaction between the controlled variables has not been considered up to this point in the

selection of the controller tuning parameters. For example, the effect of the controller tuning parameters for one controlled variable has been considered with respect to the response of that controlled variable only. It is interesting however to note the effect that specifying a fast main steam pressure response has on the response of the other three controlled variables. This effect is illustrated in Table 8.12, which gives the maximum deviation of each of the controlled variables from the setpoint.

Tuning Parameters	**Maximum Deviation of Controlled Variable from Setpoint**			
Parameter Set	**Main Steam Pressure**	**Main Steam Temperature**	**Drum Level**	**% O_2 in Stack Gases**
1. Default	0.0565	2.72	0.0923	0.072
2. Improved (slow pressure)	0.0594	2.84	0.1195	0.071
3. Improved (fast pressure)	0.0509	3.02	0.1202	0.080

Table 8.12 Comparison of Maximum Deviation From Setpoint for 3 Parameter Sets

It can be seen from Table 8.12 that specifying a faster response for main steam pressure controller reduces the maximum deviation of main steam pressure from the set-point. However, it has the undesirable effect of increasing the maximum deviation from the setpoint of the other three controlled variables. This effect is particularly significant for drum level and main steam temperature, highlighting the strong interrelations between these three variables. It follows that the final choice of tuning parameters for main steam pressure must take into consideration the performance criteria for the other three controlled variables.

The multivariable main steam pressure and drum level controller attempts to satisfy the controller tuning specifications for both main steam pressure and drum level. It is unable to achieve drum level tuning specifications however due to feedwater actuator constraints. In this case, the second set of parameters is chosen (improved parameter set with main steam pressure tuned for slow response) as it generates a reasonable response for all of the controlled variables.

Use of a multivariable controller for main steam pressure and drum level simplifes the tuning and improves the control of these two variables. Multivariable control is a systematic means of balancing the controller objectives of two controlled variables against each other. The cost function in this case specifies that equal importance should be attached to the tracking errors of both main steam pressure and drum level. The same effect could be attained by carefully tuning both variables, but this would be less systematic and more time consuming.

8.5.2 Selection of Tuning Parameters for 50% Linear Controller

The set of controller tuning parameters adopted for the 90% linear controller is used as an initial choice of tuning parameters for the 50% linear controller. A good system response is obtained at 50% using this parameter set so this parameter set is also adopted for the 50% linear controller. The system response to a 5% step change in load at 50% using this set of parameters is shown in Fig. 8.21.

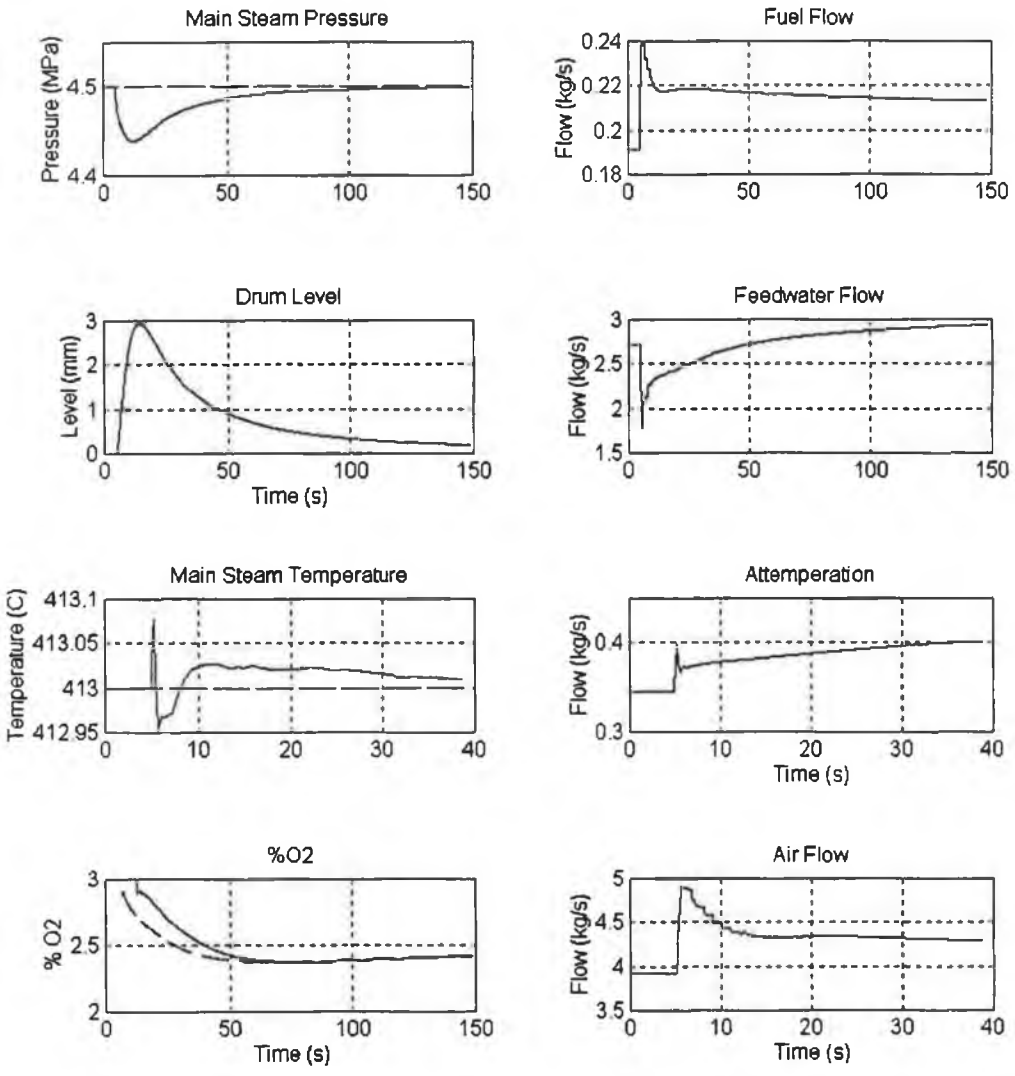


Fig. 8.22 System Response to a 5% Step Change in Load at 50% Load

8.5.3 Selection of Tuning Parameters for 10% Linear Controller

The tuning parameters for the 90% linear controller are also employed as the initial choice of tuning parameters for the 10% linear controller. The system response obtained using these tuning parameters is shown in Fig 8.23

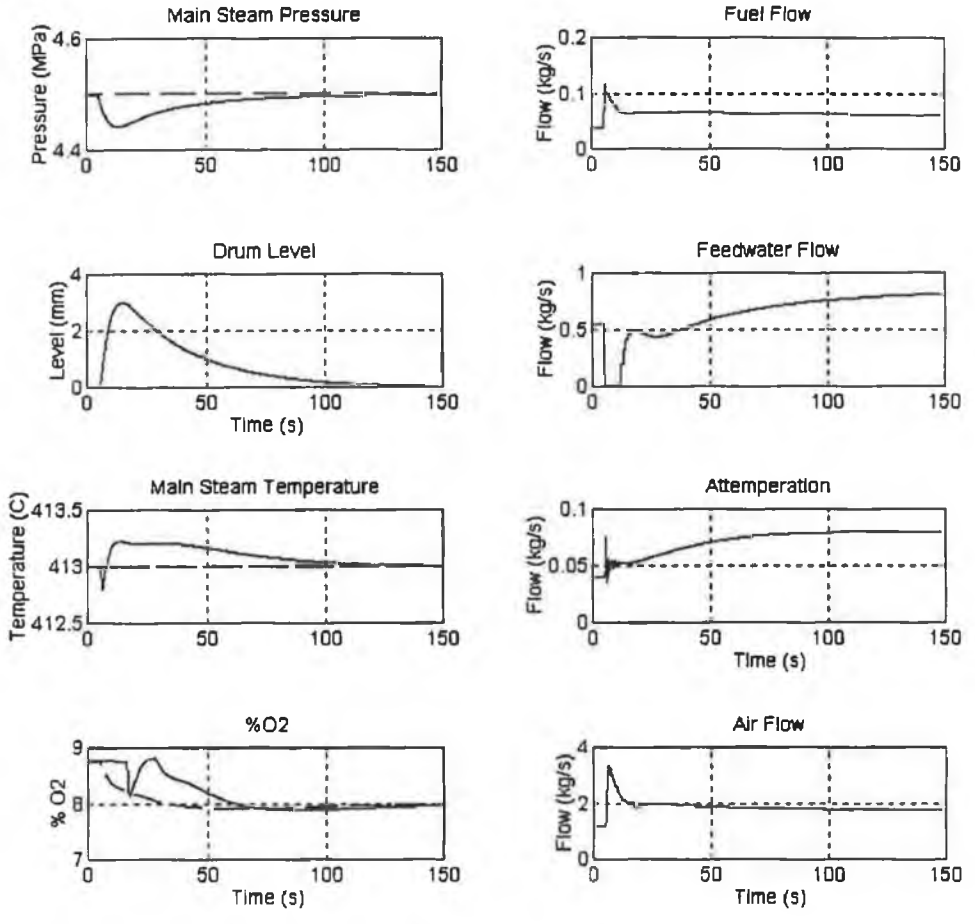


Fig. 8.23 System Response to a 5% Step Change in Load at 10% Load Using Initial Controller Parameter Tuning Set

Explanation for Relatively Poor O_2 control

The response of the percentage O_2 signal seems unsatisfactory on a first inspection. A closer examination however shows that the response would be considered good if the percentage O_2 setpoint was constant. The controller supplies a sufficient air flow to compensate for the increased fuel flow and to maintain percentage O_2 at its original setpoint. However, as the percentage O_2 setpoint has been reduced, the actual amount of air required to place percentage O_2 signal at the (new) setpoint has also been reduced. In this case the system response could not be substantially improved by using a different set of tuning parameters.

Explanation for Oscillations in Attemperation Signal

It can be also be seen from Fig. 8.23 that small oscillations occur in the attemperation signal shortly after the load disturbance occurs. It was shown in Section 8.5.1 that oscillations can also occur after a setpoint change if a very fast desired time response is specified. The oscillations of the control signal in this instance are not related to the controller tuning parameter specification however. They are the correct control signal response to variations in the air mass flow rate and fuel mass flow rate. Fig 8.24 shows the three control signals - attemperating flow rate, fuel flow rate and air flow rate during the time span in which the oscillations occur.

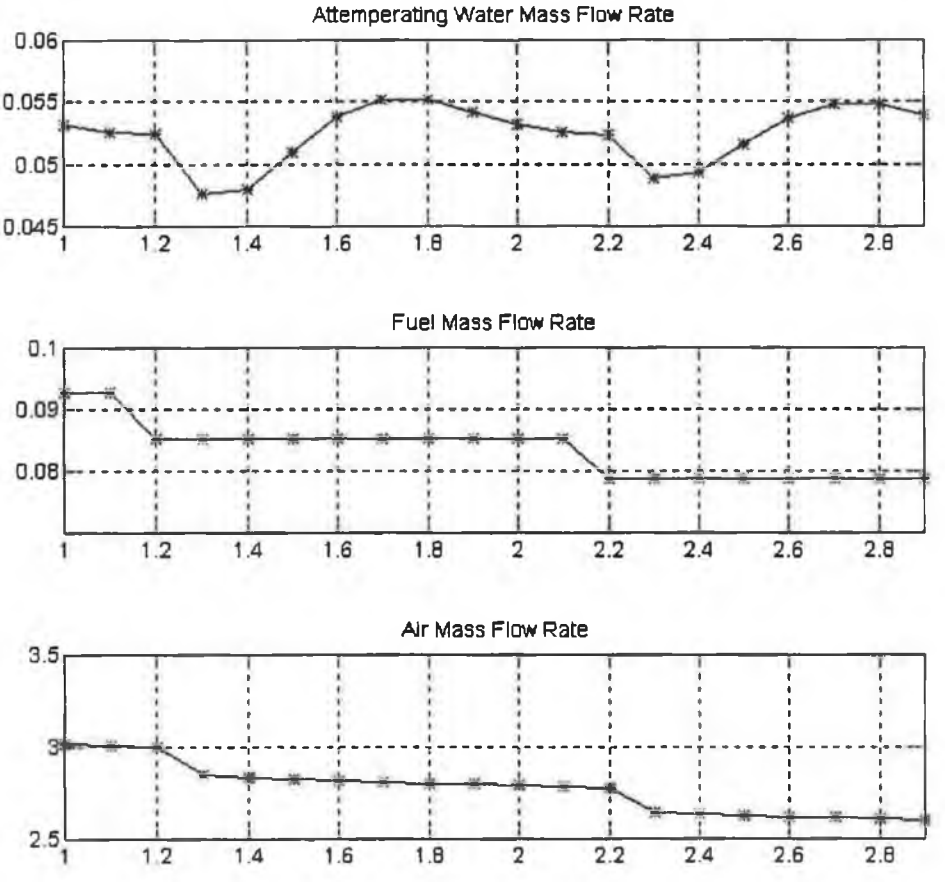


Fig. 8.24 Magnification of Behaviour of Three Control Signals After Load Disturbance

It can be seen in Fig. 8.24 that the attemperation oscillations are correlated to variations in the air mass flow rate and fuel flow rate signal. The correlation can be explained in terms of the effect of air mass flow rate and fuel flow rate on main steam temperature.

1. Each time the fuel mass flow rate signal decreases the attemperating flow rate signal increases. This is correct as a decrease in fuel flow rate will result in a decrease in main steam temperature.

2. The decrease in fuel mass flow rate prompts the percentage O_2 controller to decrease the air mass flow rate.
3. The decrease in air mass flow rate causes the main steam temperature controller to start increasing attemperation. Again, this is the correct control action as a decrease in the air mass flow rate will cause an increase in main steam temperature.

These oscillations are not as apparent at higher loads. This is explained by first comparing the effect of a load disturbance at the 10% operating point to the effect of an equal load disturbance at the 90% operating point. The disturbance causes greater variations in both air flow rate and fuel flow rate at the 10% operating point than at the 90% operating point. Fig. 8.25a and 8.25b shows the variation in the fuel flow rate and air flow rate control signal respectively after a 5% load disturbance at both the 10% and 90% operating point. The variation is shown around the initial steady state value of the control signal.

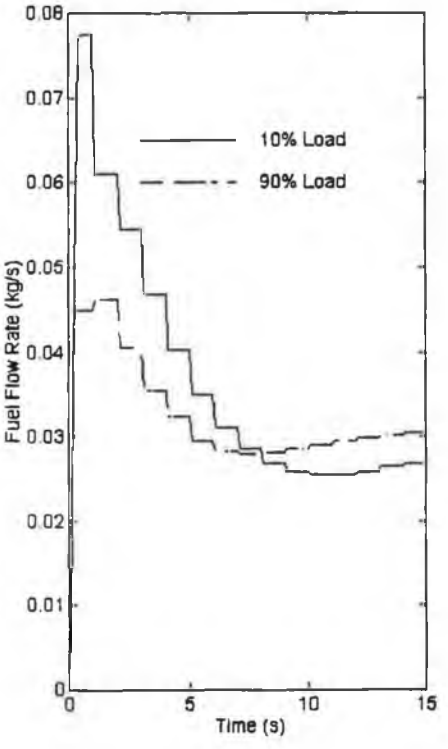


Fig. 8.25a Fuel Flow Rate Control Signal at 10% and 90% Load after 5% Disturbance

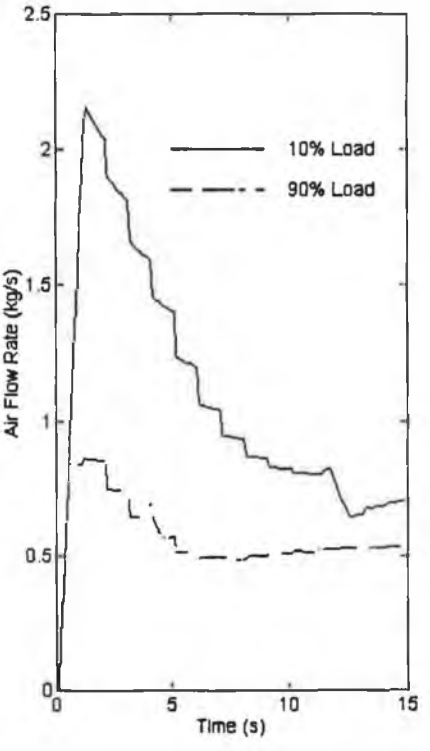


Fig. 8.25b Air Flow Rate Control Signal at 10% and 90% Load after 5% Disturbance

The increased air flow rate and fuel flow rate signals can be attributed to a general decrease in the speed of boiler dynamics as the operating point decreases. In particular, three factors contribute to the larger air flow rate and fuel flow rate signals at the 90% operating point.

1. Response time of main steam pressure to a step change in fuel flow increases as the operating point of the boiler decreases. This implies that greater control action is required to achieve the same system response time for main steam pressure at lower operating points.
2. The magnitude of the variations in the air flow rate signal is very dependent on the magnitude of the variations in the fuel flow control signal. A large increase in fuel flow rate must be matched by a corresponding increase in air flow rate.
3. Response time of percentage O_2 to a step change in air flow increases as the operating point of the boiler decreases. This implies that bigger air control action is required to achieve the same system response time at lower operating point.

It might be expected that the response time of main steam temperature to air flow rate or fuel flow rate would also increase with the decrease in operating point. This would allow the attemperating flow rate to respond more slowly to air or fuel disturbances at low operating points than at high operating points. This is not the case however, as the response of main steam temperature to variations in fuel mass flow rate or air flow rate is not as dependent on operating point as that of either main steam pressure or percentage O_2. This is due to the fact that main steam temperature is almost completely dictated by conditions in the superheater. The superheater is placed close to the furnace and variations in the fuel or air flow rate have an almost immediate effect on the superheater temperature. In addition the superheater has a much smaller heat and mass storage capacity than either the boiler or the furnace, so superheater response time can vary over a much smaller range than either boiler or furnace response time.

The magnitude of the oscillations could be reduced by specifying a slower response time for both main steam pressure and percentage O_2. This would reduce the variations in air flow rate and fuel flow rate after a disturbance, and thus reduce the required attemperation control action. This would reduce actuator activity for the attemperating control signal and prolong actuator life time. However, the benefits in terms of actuator wear are outweighed by a possible consequence of increasing system response time. At low loads the boiler reacts more slowly to a demand for increased steam (load). Consequently, it is particularly important at low loads to react as quickly as possible to an increased load demand, in order to prevent the boiler steam storage capacity being fully depleted. Increasing the desired time response of main steam pressure increases the depletion of the steam storage capacity. Given that the oscillations do not indicate possible system instability, it is preferable to use the original controller tuning parameters and accept the resulting oscillations.

8.6 Results

The performance of each of the three linear controllers has been demonstrated for a 5% step change in load. In order to test the performance of the fuzzified linear control strategy, a 5% step load disturbance is applied at 70% load. The actual controller output in this instance is equal to a weighted sum of the output of the 50% linear controller and the 90% linear controller. The system response is shown in Fig. 8.26.

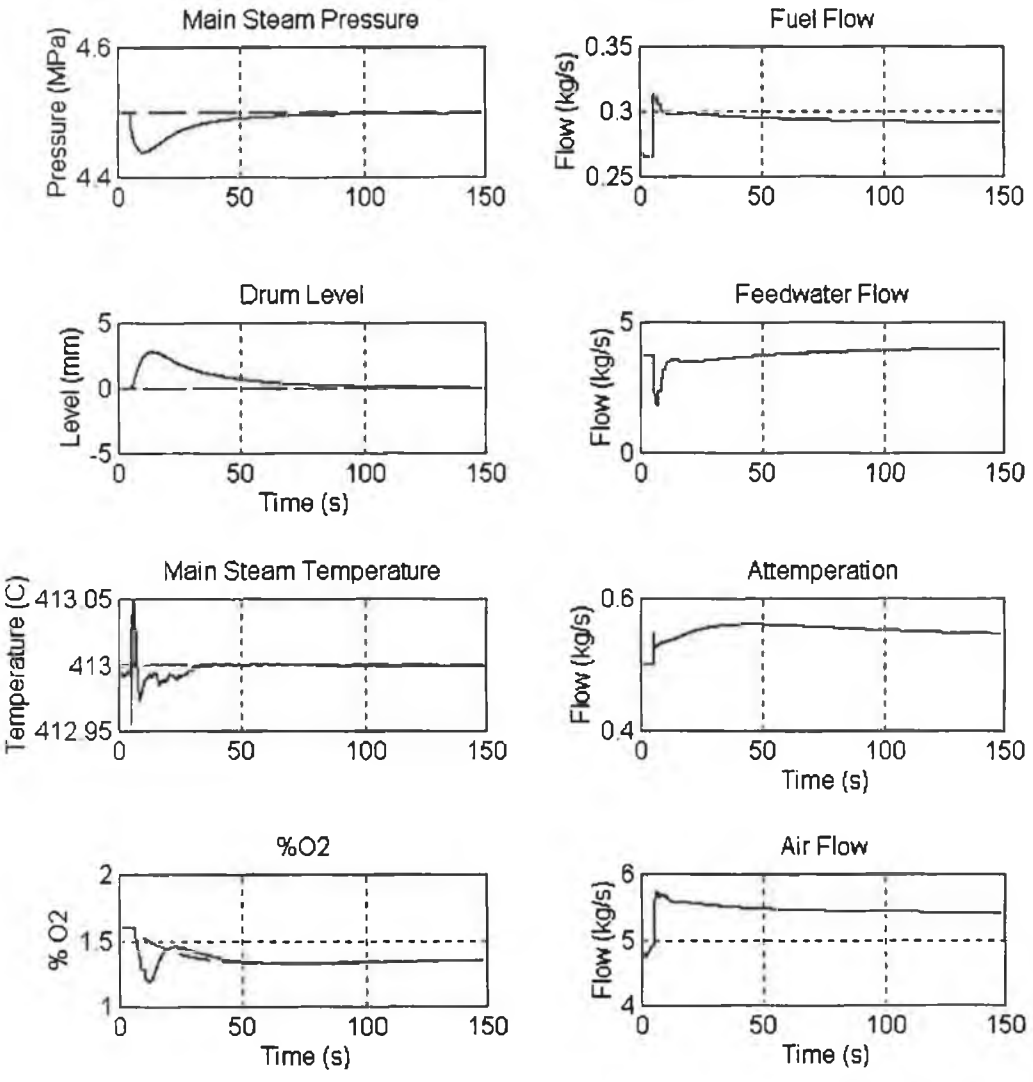


Fig. 8.26 System Response to a 5% Step Change in Load at 70% Load

The system response at 70% is similar to that obtained at either 50% or 90%, demonstrating the applicability of the fuzzified control strategy of combining the output of two linear controllers. An important feature of the predictive methodology is seen on a comparison of the feedwater flow control signal and the drum level signal. The predictive controller predicts that drum level will

start to decrease, due to the increased evaporation rate and pre-emptively increases feedwater flow rate before the drum level actually starts to fall. This action pre-empts drum level dropping below the set-point. Such action is only possible with a model-based predictive control strategy which has explicit knowledge of the nonminimum phase effects of the system.

8.7 Conclusions

Linear predictive controller is an effective control strategy. Simulation results show that predictive control can achieve very good system response. The control has been achieved in a number of ways:

- The predictive controller exploits knowledge about the expected response to the plant.
- The multivariable drum level and steam pressure controller exploits knowledge about the interactions between these two variables.
- The controller is well tuned as the tuning parameters are physically meaningful. Simulation results have confirmed that there is a close and intuitive relationship between the controller tuning parameters and the system response.
- The controller has been fuzzified to operate well over the full operating range of the plant.
- The controller has been adapted to control a delayed variable - percentage O_2 in the stack gases by incorporating a Smith regulator (Smith (1958)) into the standard control law.

In addition to controller performance, the strategy has important design features. Controllers coefficients are obtained directly and automatically from the plant model. The control solution is analytical and can be quickly calculated.

Control tuning has highlighted a possible source of problems. The internal controller model of the main steam temperature controller is inaccurate for high frequency input signals. The inaccuracy is traced to the discretisation of the continuous linearised model. This suggests that an inherently discrete model is preferable to a discretised model.

In summary it has been shown that good boiler control can be achieved using predictive control. This standard of control has been obtained at the cost of some design effort. However a very large payback can be expected in terms of fuel costs, plant life-time and steam quality.

9. Development of a Nonlinear Predictive Controller

9.1 Introduction

Currently, most model-based boiler control strategies rely on linearised models to represent the very nonlinear boiler process (McDonald and Kwatny (1973)), (Nakamura and Akaike (1981)), (Hogg and El-Rabaie (1991)), (Manayathara *et al* (1994)). In order to model the plant over its entire operating range, either a set of linearised models or alternatively, adaptive strategies which continuously update the parameters of a linear model must be used. Most of these strategies have a closed-form, linear controller solution. Such analytical controllers do not require extensive computer resources - the control law is compact, and can be computed quickly.

Linear strategies have limitations however. Linearised models may only be generated if the nonlinear function is continuous. This means that it is not possible to model actuators which exhibit hard nonlinearities such as position or speed saturation. Likewise, an analytical linear control strategy cannot optimise controller action if nonlinear constraints exist.

An alternative approach is to use a nonlinear model which has the potential to represent the plant over its entire operating range and a nonlinear numerical control law to optimise controller action The nonlinear plant model may be extended to model hard nonlinearities and the numerical control law can optimise the controller action in a constrained manner. Numerical optimisation algorithms can be computationally intensive, but since boiler dynamics are relatively slow, this does not pose a significant problem.

An optimisation algorithm searches over the error surface for the global minimum. In the case of a nonlinear optimisation problem, the search is complicated by the existence of local minima. If the optimisation algorithm fails to find the global minimum or even a good local minimum, the performance of the nonlinear controller may be inferior to that of a linear controller. Many conventional optimisation algorithms use gradient search methods which can become trapped in local minima. Genetic algorithms (Goldberg (1989)) are a relatively new optimisation technique which use stochastic techniques to search over a larger area of the error surface, and as a result, are less likely converge at a local minima. It is of interest to compare simulated controller performance using a conventional optimisation technique to that obtained using genetic algorithm optimisation. The comparison provides a measure of what improvement can be achieved in control performance by finding global as opposed to local minima.

9.2 Theory of Nonlinear Predictive Control

The control objective of a nonlinear predictive controller is the same as that of a linear predictive controller, namely, to find the controller action which minimises a specified cost function. The cost function for a predictive controller always includes some measure of the error between the predicted plant output and the set-point (or a desired reference trajectory to the set-point). Likewise, a nonlinear predictive controller comprises the same three components as a linear predictive controller - cost function, internal controller model and control law. The theory of nonlinear predictive control shall be considered for each of those three components:

9.2.1 Internal Controller Model

There is a growing interest in the implementation of nonlinear models for predictive control. Nonlinear models can represent a nonlinear plant at every operating point whereas linear models can approximate the behaviour of nonlinear plant at one operating point only. Sommer (1994) has developed predictive controllers based on the following generic nonlinear models - the generalised Hammerstein model, the Wiener model and a bilinear model. Sommer suggests that a generic nonlinear model may be the best method of modelling a system where there is no prior knowledge of the system parameters or structure. The Hammerstein model and the Wiener model are both composed of a nonlinear static component and a linear dynamic component. Sommer uses a block diagram representation which shows clearly the fundamental difference between the two models:

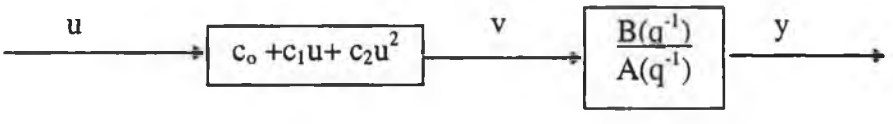


Fig. 9.1 Block Diagram of a Hammerstein Model

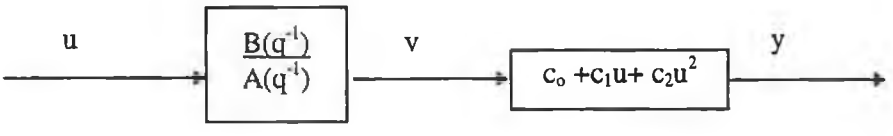


Fig. 9.2 Block Diagram of a Wiener Model

The nonlinear static component precedes the linear dynamic component for a Hammerstein model. The nonlinear static component follows the linear dynamic component for a Wiener model. Sommer uses recursive Diophantine equations (Goodwin and Sin (1984)) to calculate the predictive forms of the Hammerstein and Wiener models. As is the case for linear predictive

models, the number of parameters in the predictor equation increases in proportion to the length of the prediction horizon.

The structure of a bilinear model is shown in Fig. 9.3.

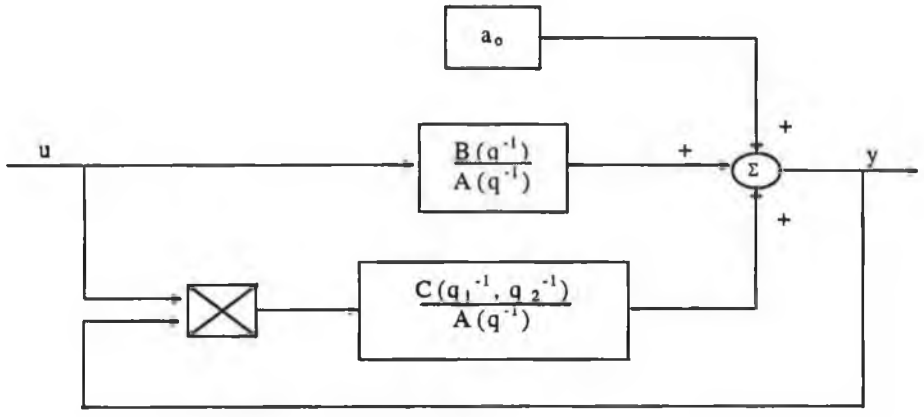


Fig. 9.3 Block Diagram of a Bilinear Model

It can be seen from Fig. 9.3 that internal feedback of the output system is incorporated into the system description. As a result of this internal feedback, the number of parameters in the prediction equation grows exponentially as the length of the prediction horizon increases. In addition, the prediction must be calculated using a recursive algorithm as no analytical expression is available to calculate the parameters of the prediction equation. Consequently, a predictive controller using a bilinear internal model is computationally costly particularly if the prediction horizon is long.

A predictive controller based upon a nonlinear model of a fossil fired power plant is developed by Gibbs and Weber (1992). The nonlinear model is a first-principles, nonlinear reduced order model. The authors cite improved performance over a wider operating range and additional diagnostic capabilities as the two primary advantages of this approach.

Neural network modelling techniques have been used to generate nonlinear, data-based boiler models (Brown *et al* (1994)), (Dai and Thompson (1994)). To date, no published results on their use in a predictive control strategy are available.

9.2.2 Cost Function

The cost function specifies the controller performance criteria by penalising undesirable system variables such as tracking error or controller action. It is completely independent of the controller implementation and as such no distinction can be made between a nonlinear controller cost

function and a linear controller cost function. Typical cost function statements and the significance of each cost function parameter to system response are described in Section 7.2.2.

9.2.3 Control Law

The control law determines the controller output, x, which minimises the cost function, $F(x)$. This is an optimisation problem which can be stated mathematically as:

$$\begin{aligned} &\textit{Minimise } F(x) \\ &\text{where} \\ &x^T = [x_1 x_2 \ldots\ldots x_n] \end{aligned} \tag{9.1}$$

The manipulated variables $x_1 \ldots x_n$, may be subject to constraints, such as simple bounds on one of more of the variables. In this case the optimisation problem is restated as:

$$\begin{aligned} &\textit{Minimise } F(x) \\ &\text{subject to} \\ &\quad l_i \le x_i \le u_i, \; i = 1,2,\ldots.n \\ &\text{where} \\ &\quad l_i = \text{lower bound on variable } x_i \\ &\quad u_i = \text{upper bound on variable } x_i \end{aligned} \tag{9.2}$$

Alternatively the constraints may be a linear or nonlinear function of one or more of the variables. The optimisation problem, is restated in a general way, to include such constraints:

$$\begin{aligned} &\textit{Minimise } F(x) \\ &\text{subject to} \\ &\quad c_1(x) \le b_1 \\ &\quad c_2(x) \le b_2 \\ &\quad \ldots\ldots\ldots\ldots \\ &\quad c_m(x) \le b_m \\ &\text{where} \\ &\quad m = \text{number of specified constraints} \end{aligned} \tag{9.3}$$

The solution to a nonlinear cost function, subject to nonlinear constraints, can be found using a numerical optimisation algorithm. Numerical optimisation has been used extensively in predictive control since the 1970's to find the solution to many types of constrained control problems (Richalet *et al* (1978), Cutler and Ramaker (1980), Moraria and Zafiriou (1989)). Traditionally, predictive control has been confined to linear internal controller models. More

recently, a nonlinear model has been employed as the internal controller model (Gibbs and Weber (1992)). One of the strengths of numerical control algorithms is versatility - numerical algorithms may be used to solve a large class of nonlinear control problems with or without constraints.

An array of numerical optimisation algorithms are available, many of which fall into one of two categories:

9.2.3.1 Gradient Search Methods

Gradient search methods attempt to improve the search efficiency by making use of function derivatives to choose a good search direction. Gradient methods are based on the Taylor expansion of the function, given here with terms involving third and higher order derivatives neglected,

$$F(x+\Delta x) \approx F(x) + g^T \Delta x + \frac{1}{2}\Delta x^T H \Delta x$$

where

$$\Delta x^T = [\Delta x_1 \Delta x_2 ..... \Delta x_n]$$

$$g^T = \left[\frac{\delta F}{\delta x_1} \frac{\delta F}{\delta x_2} .... \frac{\delta F}{\delta x_n}\right]$$

$$H = \begin{bmatrix} \frac{\delta^2 F}{\delta x_1^2} & \frac{\delta^2 F}{\delta x_1 \delta x_2} & ..... & \frac{\delta^2 F}{\delta x_1 \delta x_n} \\ \frac{\delta^2 F}{\delta x_2 \delta x_1} & \frac{\delta^2 F}{\delta x_2 \delta x_2} & ..... & \frac{\delta^2 F}{\delta x_2 \delta x_n} \\ ..... & ..... & ..... & .... \\ \frac{\delta^2 F}{\delta x_n \delta x_1} & \frac{\delta^2 F}{\delta x_n \delta x_2} & ..... & \frac{\delta^2 F}{\delta x_n^2} \end{bmatrix} \quad (9.4)$$

The expansion approximates the function value at $x+\Delta x$ by the function value at x plus a scalar approximation given by the last two terms on the right hand side. Optimisation methods which use the Jacobian gradient vector, g, only to calculate the scalar approximation are termed first order methods. Optimisation methods which also employ the Hessian matrix, H, of second order derivatives are termed second order methods.

The steepest descent method is the most basic first order method. The function value $F(x)$, is first calculated at an initial point, x. The Jacobian gradient at $F(x)$ is then used to determine the effect of Δx on $F(x+\Delta x)$. An examination of equation (9.4) shows that the maximum reduction in $F(x+\Delta x)$ occurs if Δx is in the direction of the negative gradient, g.

Second order gradient techniques utilise the Hessian matrix, H, of second order partial derivatives to increase the speed of convergence of the optimisation algorithm. Firstly, the Taylor series expansion is used to approximate the minimum value of the objective function, $F(x_{min})$, for some point, x, assumed to be near the minimum, x_{min}

$$F(x_{\min}) \approx F(x) + g^T \Delta x + \frac{1}{2} \Delta x^T H \Delta x \tag{9.5}$$

where $x_{min} = x+\Delta x$ and the Jacobian gradient vector, g, and the Hessian matrix, H are evaluated at x. Secondly, equation (9.5) is partially differentiated with respect to each element, Δx_j. The resulting equations are equated to zero, this being the first order condition for a minimum of a function.

$$\frac{\delta f(x_{\min})}{\delta x_j} = \frac{\delta f(x)}{\delta x_j} + \sum_{i=1}^{n} \Delta x_i \frac{\delta^2 f(x)}{\delta x_i \delta x_j} = 0 \quad j = 1,.....n \tag{9.6}$$

Equation (9.6), rewritten in matrix form is:

$$g = -H\Delta x \tag{9.7}$$

This equation can be solved to yield an approximation to the required movement, Δx, from the current point, x, to the minimum, x_{min}

$$\Delta x = -H^{-1} g \tag{9.8}$$

Equation (9.8) is utilised in all second order methods. If it is applied iteratively, to generate successive movements towards a minimum from an initial value, x_0, it is called Newton's method.

Gradient methods require that all nonlinear functions have continuous second order derivatives in the neighbourhood of a solution. If the Jacobian or Hessian matrix cannot be computed directly, finite difference approximations may be used.

9.2.3.2 Direct Search Methods

Direct search strategies (Gill and Murray (1974)) rely upon a direct comparison of the function values, $F(x)$, as calculated for different combinations of the optimisation variables. Individual direct search strategies differ in the method used to select the optimisation variables for the

comparison. For example, many simple direct search methods are based on a Monte Carlo approach in which trial points are generated randomly. The best of these points may be taken as the minimum or further attempts may be made to improve upon the result by defining a reduced search region in the vicinity of that point. Genetic algorithms (Kristinsson and Dumont (1992)), described in more detail in Section 9.3.4.2, are an example of a direct search technique.

The simplex method (Gill and Murray (1974)) is a popular unconstrained direct search procedure. A regular simplex is used to explore the parameter space - a regular simplex in n dimensions being $n+1$ mutually equidistant points. The method evaluates $F(x)$ at each vertex of the simplex to find the worst vertex (the one with the highest value of $F(x)$). This vertex is replaced by its reflection in the centroid of the other vertices. The simplex method may be modified in various ways to cope with constrained optimisation problems.

Gradient search methods are generally mathematically based and, as such, good convergence criteria can be achieved. This is not the case with direct search methods, which are heuristic in nature and aim to produce a good solution rather than an optimal one. One of the advantage of direct search methods is that they make few assumptions about the function, such as continuous derivatives, and thus can prove to be more reliable than gradient search methods in finding the global minimum.

9.3 Nonlinear Predictive Controller Development

9.3.1 General Control Strategy

The overall control strategy configuration adopted by the nonlinear predictive controller is identical to the linear predictive control strategy, described in Section 8.3.1. It is summarised in Table 9.1.

Controller	Manipulated Variables	Controlled Variables	Sampling Period (s)
Multivariable Main Steam Pressure and Drum Level Controller	Fuel Mass Flow Rate Feedwater Flow Rate	Main Steam Pressure Drum Level	1
Main Steam Temperature Controller	Attemperating Water Mass Flow Rate	Superheated Steam Temperature	0.1
Percentage O_2 in Stack Gases Controller	Air Mass Flow Rate	Percentage of O_2 in Stack Gases	0.1

Table 9.1 Summary of Control Strategy

9.3.2 Internal Controller Model

Three internal controller models are employed in total. The controller of the percentage of O_2 in the stack requires a model of the combustion-side of the process in order to predict the percentage of O_2 in the stack gases. The multivariable main steam pressure and drum level controller uses a model of the whole boiler in order to predict main steam pressure and drum level. The steam temperature controller requires a model of the superheater in order to predict steam temperature.

These three nonlinear models are implemented using three artificial neural networks. The neural network modelling process is described in Chapter 6.

9.3.3 Cost Function

The cost function definition is identical to the definition specified for the linear predictive controller, described in Chapter 8. The cost function is defined as the sum of the square of the error between the desired reference trajectory and the predicted model output at specified coincidence.

$$E = \sum_{i,j} (y_{m_i}(n+H_j) - R_i(n+H_j))^2$$

where

E = error at j^{th} coincidence point for i^{th} output
y_{m_i} = i^{th} model output
H_j = j^{th} coincidence point
R_i = reference trajectory for i^{th} output

(9.9)

In the case of the multivariable main steam pressure and drum level controller, the cost function includes two terms - the sum of the square of the errors for main steam pressure and the sum of the square of the errors for drum level. If desired, these terms can be weighted to prioritise the response of one controlled variable over the other. The cost function including weightings matrices is:

$$E = \sum_{i,j} \mu_i (y_{m_i}(n+H_j) - R_i(n+H_j))^2$$

where

μ_i = weighting assigned to error of i^{th} controlled variable

(9.10)

The nonlinear multivariable controller developed for this work aims to assign equal importance to tracking errors in main steam pressure and drum level. To achieve this, the variables must be scaled appropriately as main steam pressure tracking errors are typically a factor of 1e5 greater than drum level tracking errors. To scale the variables, the expected range of both variables is

determined and mapped linearly to the range [-1,1].

The desired reference trajectory is defined as an exponential curve between the current process output and the set-point. The system response can be tuned by varying the rate of change of this exponential curve or by choosing different coincidence points on this curve. This reference trajectory is described by:

$$R_i(n+H_j) = S_i - \lambda_i^{H_j}(S_i - y_{p_i})$$

where

R_i = Reference trajectory for process output i
S_i = Setpoint for process output i (9.11)
$y_{p_i} = i^{th}$ process output
$H_j = j^{th}$ coincidence point
$0 \le \lambda_i \le 1$

The physical relevance of λ_i can be seen by writing it in terms of the 63% rise-time of the desired exponential curve.

$$\lambda_i = \exp\left(\frac{-T_s}{\tau_i}\right)$$

where (9.12)

T_s = sampling period
τ_i = 66% rise - time

A fundamental advantage of using numerical control laws is the ability to handle constraints. Unconstrained linear control laws which do not account for actuator constraints assume that the control signals may vary instantaneously and infinitely in any direction. In fact, the maximum change in the applied control signal during each sampling period is limited by the maximum speed of the actuator and by actuator position limits.. The constraints imposed on the applied control signal, u, by actuator rate constraints, can be written in terms of the maximum speed of actuator.

$$\Delta u = T_s v$$

where

Δu_{max} = maximum change in applied control signal (9.13)
T_s = sampling period
v = maximum rate of change for actuator

The applied control signal is also subject to position constraints. If both position and rate constraints are taken into account the cost function must be optimised subject to both constraints.

For a SISO controller, the constrained cost function is

$$\underset{u}{Minimise}\ E(u)$$

subject to

$$\max(u-\Delta u, u_{\min}) \le u \le \min(u+\Delta u, u_{\max}) \tag{9.14}$$

where

$u_{\min}$ = low actuator position constraints

$u_{\max}$ = high actuator position constraints

Equation 9.14 shows that both actuator rate and position constraints can be implemented as simple bound constraints, which must be recalculated at each sampling interval using the current value of u.

The actuator constraints are presented in Table 9.2.

Manipulated Variable	Rate Constraints (kg/s^2)	Minimum Position Constraint (kg/s)	Maximum Position Constraint (kg/s)
Fuel Flow Rate	0.19	0.01	0.42
Feedwater Flow Rate	3.39	0.01	7.48
Attemperating Water Flow Rate	0.29	0.01	1.17
Air Flow Rate	1.92	0.01	7.67

Table 9.2 Actuator Constraints

9.3.4 Control Law

At each sampling interval, controller action is calculated using a numerical optimisation algorithm. The optimisation algorithm is initialised with a proposed control signal(s), u^* and uses an iterative procedure to find the optimal control signal(s), $u_{optimal}$. The sequence of events is shown in Fig. 9.4.

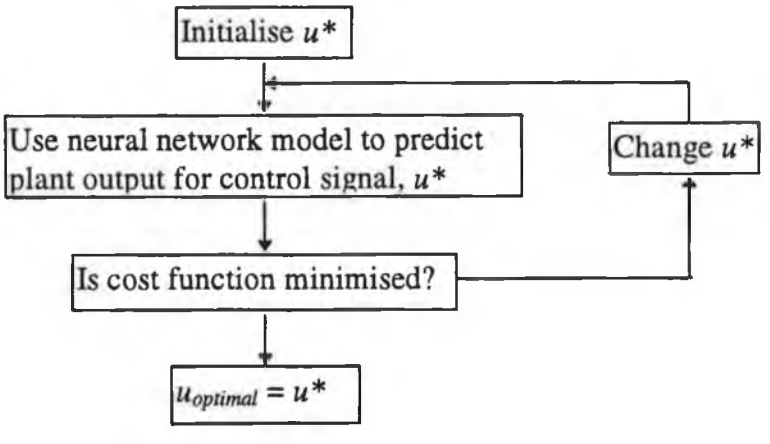


Fig. 9.4 Optimisation Procedure

In order to determine the control solution, the optimisation algorithm must be provided with the following:

1. Objective Function Value - this is equal to the value of the cost function for a given controller action.
2. Constraints Function or Bounds - this defines the constraints (if any) on the controlled variable or on the controller action. A constraint can be a simple bound or a function of the controller action.

A large number of numerical optimisation methods have been developed to solve particular types of optimisation problem as efficiently as possible. Optimisation problems are categorised by the properties of the objective function or the constraint function. Examples of different types of objective and constraint function properties are presented in Table 9.3a and Table 9.3b, respectively.

Examples of Objective Function Properties
1. Linear
2. Sum of squares of linear functions
3. Quadratic
4. Nonlinear
5. Sum of squares of nonlinear functions

Table 9.3a Examples of Objective Function Properties

Examples of Constraint Function Properties
1. None
2. Bounds
3. Linear
4. Sparse Linear
5. Nonlinear

Table 9.3b Examples of Constraint Function Properties

The optimisation problem for each of the nonlinear controllers used in this application has the following properties:

1. Objective Function is sum of squares of nonlinear functions
2. Constraints are simple bounds

The choice of optimisation function for this problem must also take into consideration that the first and second derivatives are to be calculated using finite differences. This can substantially reduce the efficiency of the algorithm.

9.3.4.1 Optimisation Using a Gradient Search Method

Based on the problem description, a gradient search algorithm is chosen which satisfies all of the problem criteria. The chosen optimisation algorithm is a quasi-Newton routine for finding the minimum of a nonlinear function $F(x)$ of n variables $x = (x_1, x_2 \ldots\ldots x_n)^T$. It requires function

values only and can handle simple bounds. Quasi-Newton methods approximate the Hessian matrix or its inverse at each step by accumulating information from the preceding steps. At each iteration, the Hessian matrix is modified to incorporate information about the curvature of the objective function along the last search direction. Quasi-Newton methods are more efficient than Newton-type methods because the Hessian matrix is not recomputed by finite differences at each iteration. (Newton-type methods calculate the Hessian matrix for the k^{th} step using derivative information from the neighbourhood of $x^{(k)}$ only)

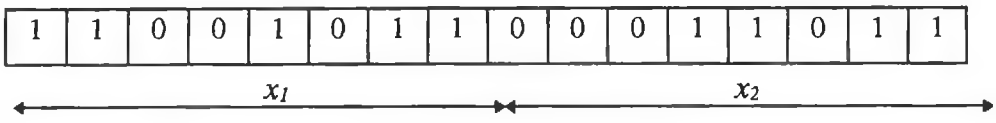
The algorithm is included in the *"NAG Foundation Toolbox"* (NAG Foundation Toolbox User's Guide (1995) a collection of numerical algorithms for use in the *MATLAB* development environment (MATLAB User's Guide (1993). It is denoted by "E04JAF" in the User's Guide, but will be referred to as the gradient search optimisation algorithm in this work.

The accuracy of the solution is improved if the objective function values and the function variable are scaled appropriately. Scaling is particularly important in the case of the multivariable drum level and main steam pressure controllers. Drum level and main steam pressure must be scaled to prevent either variable dominating the objective function value. Feedwater flow and fuel flow are also scaled to improve the numerical conditioning of the optimisation problem. The same scaling is used as for the neural network model inputs i.e. all the controlled and manipulated variables are scaled to lie within a range of [-1,1].

9.3.4.1 Optimisation Using a Direct Search Method

Gradient search methods rely on local gradient information to guide the search over the error surface. When the local gradient is zero in all directions, the search ceases. Consequently, it is possible for the search to converge to a local rather than global minimum. It is to be expected that controller performance could be improved by the use of an optimisation algorithm which has a greater likelihood of converging to the global minimum. Genetic algorithm optimisation is a direct search optimisation technique which has a greatly enhanced chance of finding the global minimum. Genetic algorithm optimisation is based upon the natural principles of evolution and in particular on the principle of survival of the fittest. The optimisation process commences with a set of potential solutions which is refined during successive generations to yield an improved set of solutions. The actual optimisation process can be tailored extensively to suit the given optimisation problem. Genetic algorithms have been applied successfully to many nonlinear modelling and control problems (Chipperfield *et al* (1994), Kristinsson and Dumont (1992)). The optimisation algorithm used for the three nonlinear controllers is described here. It uses standard genetic algorithm optimisation techniques.

1. A random set of initial solutions is generated. Each solution is referred to as an individual and the entire set of solutions is referred to as a population. In this work, a population of 10 individuals is used for the nonlinear controllers. This yields reasonable results and does not involve a large computational effort. The size of the computational effort increases as the population size is increased, largely because the cost function must be evaluated for every individual in the population. The individuals are encoded as strings known as a chromosome. In this instance the individuals are encoded in binary form but other representations may also be used. If a solution is composed of more than one variable, as for the multivariable main steam pressure and drum level controller, all of the variables are concatenated into a single chromosome as shown below:



2. The objective function value of each chromosome is calculated and the chromosomes are ranked according to their objective function values. Each individual is assigned a fitness value in accordance with their objective function value.

3. Individuals are selected from the population for reproduction in a probabilistic manner. This means that individuals with a high fitness value have a high probability of being selected for the next generation. For example, the "roulette wheel" selection mechanism involves mapping the population onto the circumference or area of a roulette wheel. Individuals with a high fitness value are allocated a larger portion of the roulette wheel In Fig 9.5 the circumference of the roulette wheel is equal to the sum of the fitness values of a population of 4 individuals. Individual 1 is the most fit and individual 4 is the least fit.

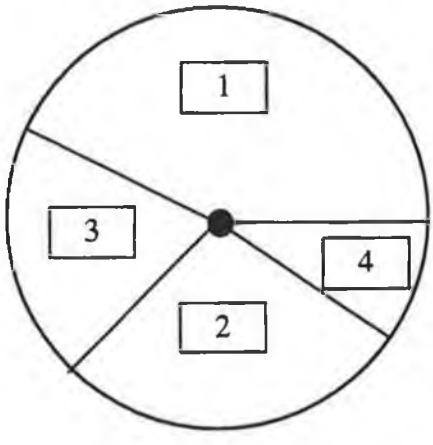
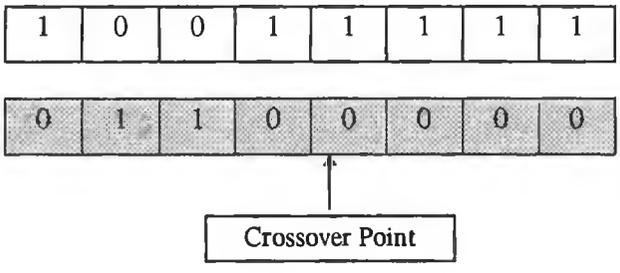


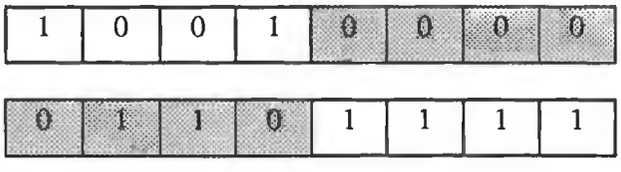
Fig 9.5 Roulette Wheel Selection

To select an individual the "roulette wheel" is "spun" by generating random number. The random number must lie between 0 and the sum of the fitness values of the population.

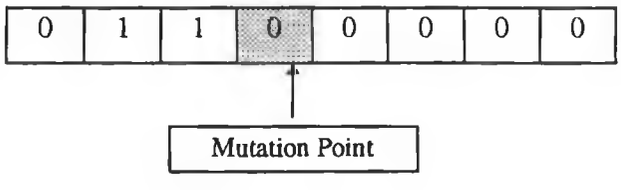
4. The information in the selected individuals can be mixed by recombining two individuals. Single point crossover, one of the simplest forms of recombination, involves two individuals which are referred to as parents. An integer position, *i*, known as the crossover point, is generated at random between *1* and the string length of the parents.



The two parents are crossed at position, *i*, to generate the two new offspring shown below.



5. Mutation can also be used to generate new individuals with new genetic information. Given the following binary-coded individual:



An integer position, *i*, is generated at random and the value of the bit at that position is inverted to generate the following mutated offspring:

6. The set of new individuals which have been generated by selection, recombination and mutation is called a new population. It is not necessary to keep all of this new population, nor is it desirable to discard all of the old population. The fittest members of the old population can be reinserted back into the new population. Reinsertion can be used to allow the fittest (or alternatively the oldest) members of the old population to continue into the next generation. This keeps the overall population size steady and ensures that the most fit members will

survive through successive generations. This procedure of preserving fit solutions from one generation to the next is known as "elitism". In this instance 10% of the old population are inserted into the new population at each generation.

7. The optimisation process is stopped after a specified number of generations or when a prespecified tolerance has been reached.

Genetic algorithms improves the likelihood of finding a global rather than a local minima in a number of ways:

1. The search commences at more than one initial point. These points are generated randomly and should be spread over the possible solution space. The disadvantage of this approach is that the computational overhead of optimisation increases as the size of the population increases. The objective function must be computed for each individual - this can be a significant computational effort, particularly if a long prediction horizon is employed.
2. The search does not rely on gradient information - searching continues even if a local minima has been reached
3. New solutions are generated in a probabilistic manner by recombination and mutation. This means that the probability of searching a particular string is never zero. It also promotes the exploration of the search space, rather than allowing the search to converge quickly to fit individuals.

Constraints may be taken into account in genetic algorithms by adding a penalty term to the cost function which tends to be large when the constraints are violated. This has not been implemented in this instance, as it increases the risk of "losing" a good solution which happens to be just outside the constraints. This is not a problem in gradient search algorithms, which can search along a constraint for a solution.

9.3.5 Integral Action

It has been shown in Section 8.3.3 that integral control action can be incorporated directly into the linear predictive control law by stating the cost function incrementally. However, it is not possible to incorporate integral action directly into the nonlinear predictive controller described in this chapter. This difference between the linear and nonlinear predictive controllers arises from a fundamental difference between the internal controller models of the two controllers. The linear controller uses a linearised state space model of the boiler. If the model is in steady state, the current model output must be equal to the predicted model output. (By stating the control law incrementally, the model cannot reach steady-state until the plant output equals the setpoint, thus

eliminating steady-state errors). In the case of the neural network model, the current model output may differ from the predicted model output even where the model is in "steady state" (meaning that both model output and predictor output are unchanging). The objective of the controller is to place the predicted model at the setpoint. Consequently, a steady state error may arise if the predicted model output is not equal to the actual plant output. If the setpoint signal is used as the reference trajectory the magnitude of the steady state error is equal to the difference between the plant output and the predicted output. In steady state, the error is then equal to:

$$\begin{aligned} ERR_i &= SP_i - y_{p_i} \\ &= y_{m_i}(n+H_j) - y_{p_i} \end{aligned} \tag{9.15}$$

where

ERR_i = Steady - state error for ith controlled variable

Integral action is incorporated into the nonlinear control law to remove any steady state errors, by adding the integrator output to the desired reference trajectory as defined by equation (9.11). The modified reference trajectory is:

$$R_i' = R_i + I_i$$

where

R_i' = Modified reference trajectory for i^{th} controlled variable (9.16)

R_i = Original reference trajectory

I = Integrator output for i^{th} controlled variable

The effect of the integrator action is minimised as much as possible in order to prevent unnecessary integrator action when an accurate internal controller model is available. If an accurate internal controller model is available, integral action can only reduce the performance of the controller. The integrator output is kept small by using the following strategies:

1. The integrator output is only updated if the plant output is within a certain region of the setpoint. For example, if main steam temperature is more than 0.3°C from the setpoint, the integrator output is not updated. Effectively, this assumes that the internal controller model can predict main steam temperature to within 0.3°C and thus bring main steam temperature to within 0.3 °C of the setpoint. The integral action is then used to reduce the setpoint error further. This region around the setpoint is referred to as the "region of integral action" from here on.
2. A small value of integrator gain is employed for each of the four controlled variables.

Within the region of integral action, the integrator output is updated as follows:

$$I_i = I_i + k_i(SP_i - y_{p_i})$$

where (9.17)

k_i = Integrator gain

The integrator gain, k_i, and the region of integral action are presented in Table 9.4 for each of the controlled variables:

Controlled Variable	Integrator Gain	Region of Integral Action
Main Steam Pressure	5000	2e-2 MPa
Main Steam Temperature	0.25	1e-3 °C
Drum Level	0.2	1e-2 mm
Percentage O_2 in Stack Gases	0.3	1e-3 %

Table 9.4 Specification of Integral Action

9.3.6 Application to Systems with Time Delay

The percentage of O_2 in the combustion gases cannot be measured in the combustion chamber because of the high temperatures present in the combustion chamber. The measurement takes place in the stack where the combustion gases have cooled to a lower temperature. Transportation of the combustion gases to the stack and a delay in the percentage O_2 sensor constitute a significant pure delay between controller action and the measured response of percentage O_2. The nonlinear predictive controller uses the same strategy to cope with time delay systems as the linear predictive controller. It effectively incorporates a Smith regulator (Smith (1958)) into the predictive controller which notionally places the pure time delay outside of the feedback loop. The process model is first separated into two separate entities - a model without time delay followed by a model of a pure time delay (see Fig 8.2). This model structure is expressed mathematically as:

$$y_m(n) = q^{-d} y_{m(adv)}(n)$$

$$y_{m(adv)}(n) = H_m(q)u(n)$$

where

$y_m(n)$ = output of model of entire process, including delay (9.18)

d = number of sample periods of delay

$H_m(q)$ = model of process, excluding time delay

$y_{m(adv)}(n)$ = output of $H_m(q)$

For control purposes it is assumed that the process variable, $y_{p(adv)}$, corresponding to $y_{m(adv)}$ is the controlled variable i.e. that the controlled system excludes the pure measurement delay. In practice, no measurable variable, $y_{p(adv)}$ exists corresponding to $y_{m(adv)}$. This difficulty is overcome by using the internal controller model to estimate the value of $y_{p(adv)}$, i.e. the internal controller model is used to predict the undelayed "measured" output.

$$y_{p(adv)}(n) = y_p(n) - y_m(n) + y_{m(adv)}(n) \tag{9. 19}$$

The reference trajectory must be initialised by this advanced process output instead of by the actual process output. The desired reference trajectory is now defined as follows:

$$\varepsilon(n+H_j) = \lambda_i^{H_j}\varepsilon_i(n)$$

where

$$\begin{aligned} &\varepsilon_i(n) = S_i - y_{p(adv)_i}(n) \\ &S_i \;=\; \text{Setpoint for i}^{\text{th}}\text{ process output} \\ &y_{p(adv)_i} \;=\; \text{i}^{\text{th}}\text{ advanced process output (controlled variable)} \\ &0 \le \lambda_i \le 1 \end{aligned} \tag{9.20}$$

9.4 Controller Tuning

9.4.1 Selection of Tuning Parameters at 90% Operating Point

The response of each of the controlled variables may be specified using the following three tuning parameters:

1. Time Constant of the Reference Trajectory
2. Position of the Coincidence Point
3. Integrator Gain

The effect of the third parameter - integrator gain - is not explicitly considered. A very small value of integrator gain is employed to prevent integral action having a significant effect on the transient response of the controlled variables. In addition the integrator output is updated only when the controlled variable is within a small region of the setpoint.

The strategy used to investigate the relationship between the first two controller tuning parameters and system response is the same as that used for linear predictive controller i.e. the

effect of the tuning parameters is investigated by carrying out step response tests for various combinations of these parameters. In addition, the controller tuning parameters and the controller performance indices used in this investigation are the same as those used for the linear predictive controller. The default set of tuning parameters, which provides a point of comparison for each set of tuning parameters investigated is presented in Table 9.5.

Controlled Variable	Time Constant of Reference Trajectory (s)	Position of Coincidence Point (s)
Main Steam Pressure	24	12
Main Steam Temperature	0.6	0.3
Drum Level	30	15
$\%O_2$ in Stack Gases	3.6	1.8

Table 9.5 Default Set of Tuning Parameters

A step response test is carried out on each of the controlled variables using the following combinations of parameters:

Test 1 Use default parameters

Test 2 Increase default time constant only by 33%

Test 3 Increase default coincidence point only by 33%

Test 4 Increase both default time constant and coincidence point by 33%

Test 5 Decrease both default time constant and coincidence point by 33%

The effect of the controller tuning parameters for one controlled variable is shown for that controlled variable only. Likewise only the response of the manipulated variable which has the greatest effect on the controlled variable is included in the set of results for that controlled variable.

The control signal of each controller is found using the gradient search optimisation method in preference to genetic algorithm optimisation as it has a smaller computation overhead.

Main Steam Pressure

A step change is applied to the main steam pressure setpoint. The response of main steam pressure and the corresponding controller action are shown in Fig. 9.6 and Fig. 9.7 respectively for all five combinations of the test parameters.

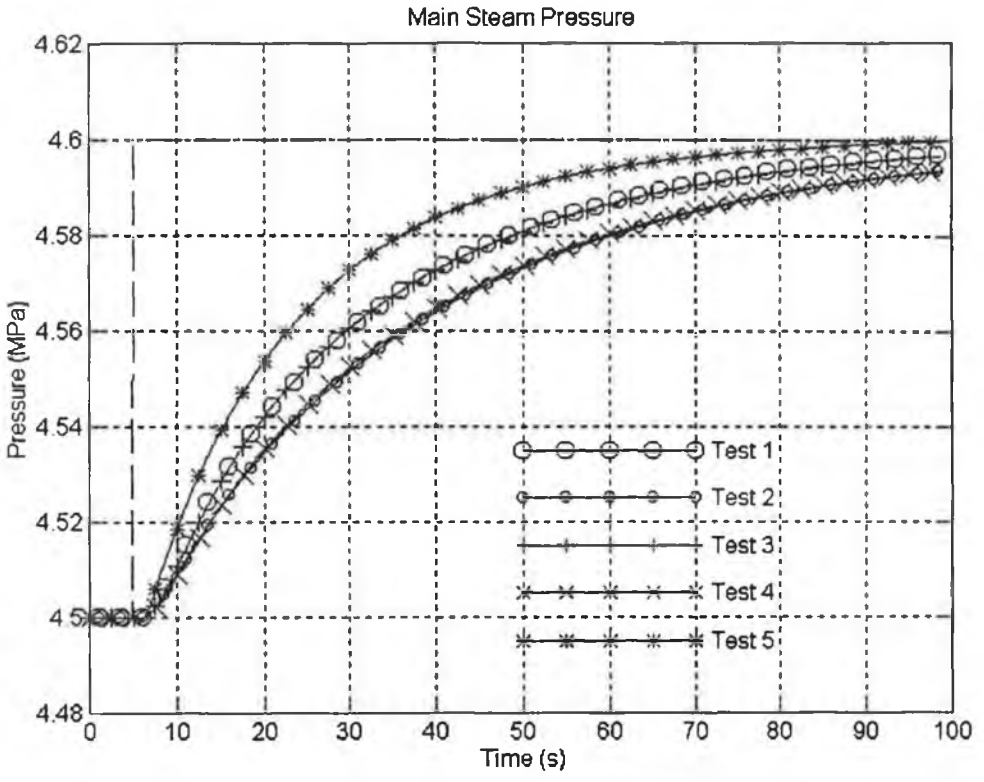


Fig. 9.6 Response of Main Steam Pressure to a Step Change in Setpoint

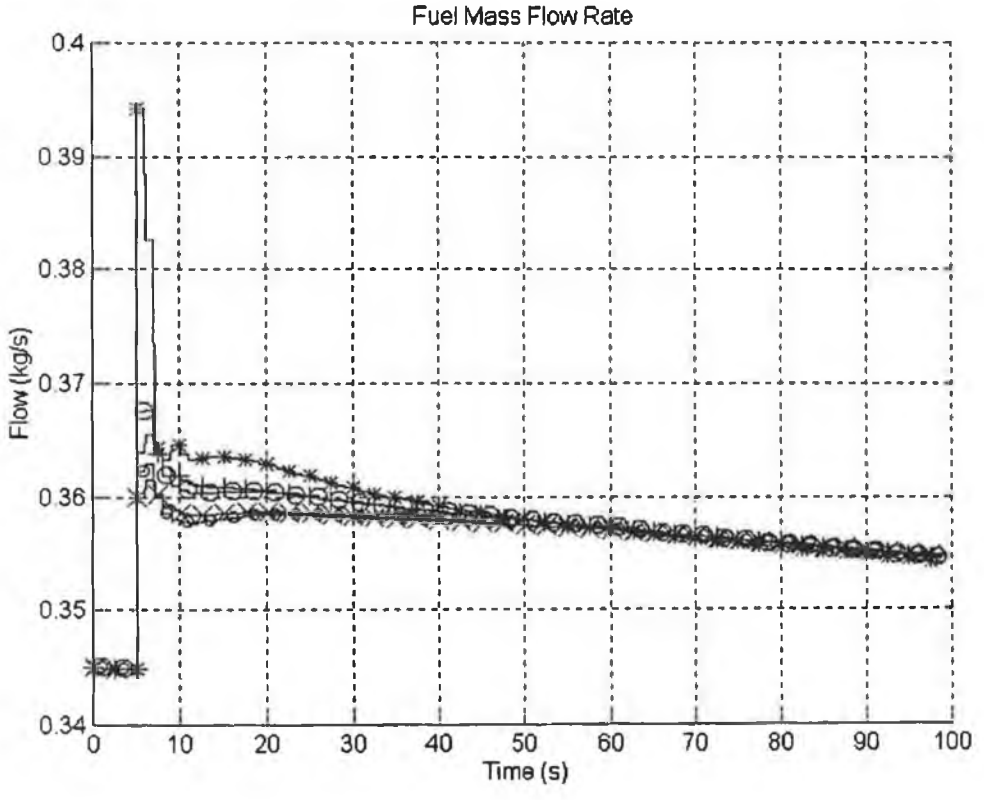


Fig. 9.7 Response of Fuel Flow to a Step Change in Setpoint

Fig. 9.6 shows that the system responses fall into three distinct groups, which could be loosely described as fast, medium and slow. A quantitative analysis of these results is carried out by defining two indices to quantify system response and controller activity. The 63% response time is defined as an index of system response. This index was chosen because it allows a direct comparison of the first tuning parameter (reference trajectory time constant or desired 63% response time) and the resulting system response (actual 63% response time). The first index for controller activity is equal to the maximum overshoot or undershoot of the manipulated variable. The second index for controller activity, *ISU*, is a measure of controller "energy". These indices have been defined in more detail in Section 8.5.1. The controller tuning parameters and performance indices for each of the five tests are presented in Table 9.6, with shading used to indicate a non-default tuning parameter.

Tuning for Main Steam Pressure Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Fuel Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	24	12	26.98	1.77e-2	2.54e-2
2	32	12	34.18	1.25e-2	1.52e-2
3	24	16	27.77	1.55e-2	2.45e-2
4	32	16	33.47	1.11e-2	1.53e-2
5	16	8	19.28	4.46e-2	5.61e-2

Table 9.6 Effect of Main Steam Pressure Controller Tuning Parameters

Test 1 The actual 63% response time (27s) is just three seconds longer than the desired 63% response time (24s). This indicates that it is possible to specify the actual plant response time to a reasonable degree of accuracy using the desired reference trajectory.

Test 2 The default reference trajectory time constant is increased to 32s, causing the plant response time to increase to 34s. This test confirms that the actual plant response time can be specified quite accurately *via* the desired reference trajectory. Fuel flow overshoot decreases by 29% and the *ISU* index by 40%.

Test 3 The coincidence point position is changed from 12s to 16s. Plant response time increases by less than 1s and fuel flow overshoot decreases by 12%. This suggests that the position of the coincidence point mainly effects high frequency controller activity and has

little effect on the overall plant response time. This is confirmed by comparing the *ISU* index with that of Test 1 - it can be seen that the overall controller action is unchanged.

Test 4 The reference trajectory time constant is increased to 32s and the coincidence point is moved to 16s. It is to be expected that the plant response time would be slower than that achieved in Test 2. However, plant response time is very slightly faster (0.71s) and more importantly fuel flow overshoot is 12% smaller than for Test 2. This test confirms that increasing the coincidence point can decrease high frequency controller activity without necessarily compromising plant response time.

Test 5 The reference trajectory time response is decreased to 16s and the coincidence point is placed at 8s. As a result an actual plant response time of 19.28s is obtained. The faster response time requires greater controller activity, in this case, the fuel flow overshoot is increased by 152% and the *ISU* index by 120%.

The results demonstrate that it is possible to specify the response time for main steam pressure with a reasonable degree of accuracy via the reference trajectory time constant. The results also demonstrate that the reference trajectory time constant has a much more dominant effect on system response than the position of the coincidence point. The position of the coincidence can however be used to increase or decrease high frequency controller activity.

The tuning parameters should be chosen so as to achieve a reasonable compromise between controller action and system response. If a fast response is required, the parameters used in Test 5 could be adopted. However, in the case of the 90% linear controller, the tuning parameters of Test 5 are rejected on the basis that increasing the response time of main steam pressure also increases the maximum deviation of drum level from the setpoint. For consistency, these tuning parameters are also rejected in the case of the nonlinear controller.

The parameters used in Test 3 are seen to offer a good compromise between controller action and system response. The controller tuning parameters of Test 2 and 4 are not as attractive because they increase system response without significantly decreasing controller activity.

Drum Level

The response of drum level to a step change in setpoint is presented in Fig. 9.8 The corresponding controller action - feedwater flow rate - is presented in Fig. 9.9.

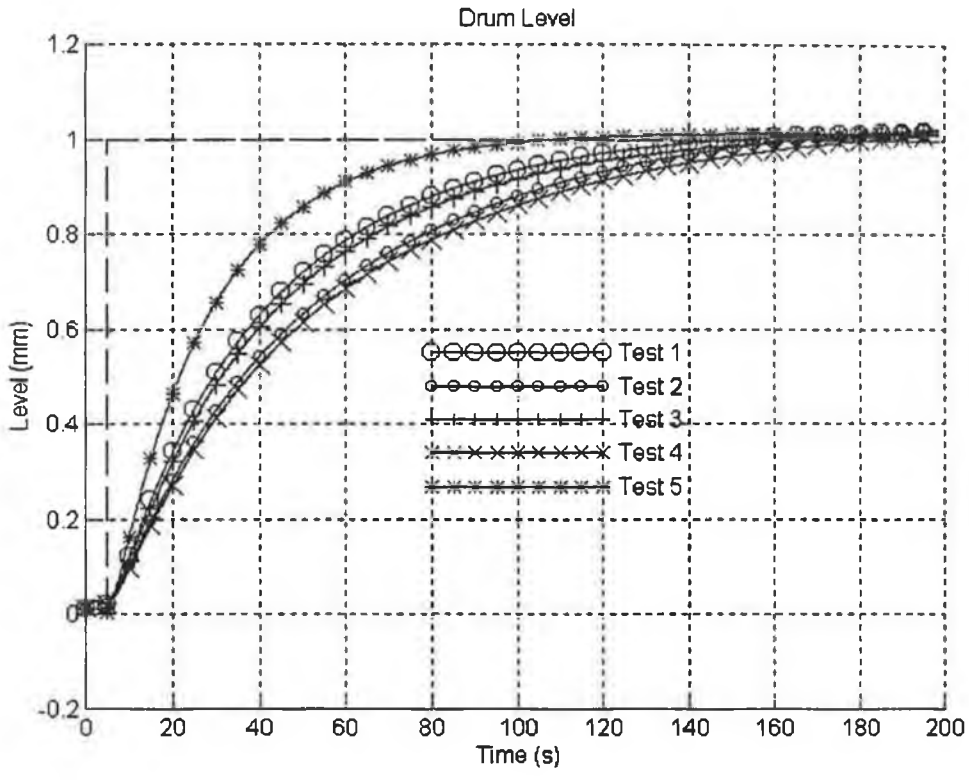


Fig. 9.8 Response of Drum Level to Step Change in Setpoint

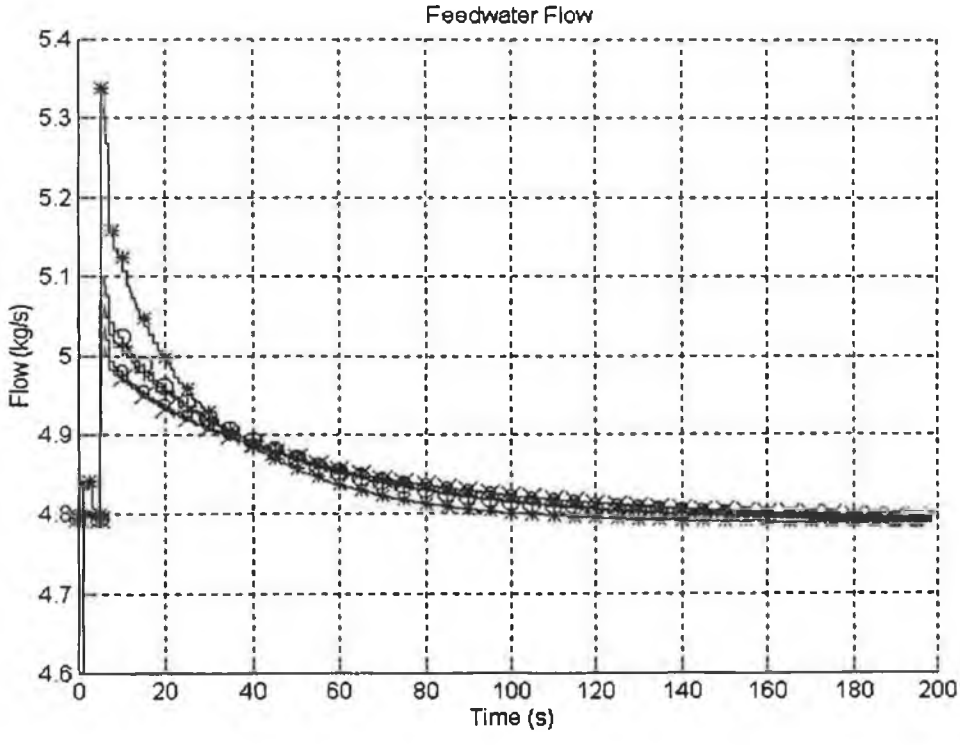


Fig. 9.9 Response of Feedwater Flow to Step Change in Setpoint

As for main steam pressure, the response obtained in the five tests may be classified as fast, medium or slow. The relationship between controller tuning parameters, drum level response and feedwater flow is summarised in Table. 9.7.

Tuning for Drum Level Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Feedwater Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	30	15	35.06	0.310	7.9
2	40	15	44.72	0.243	5.3
3	30	20	37.65	0.272	7.1
4	40	20	47.05	0.217	4.8
5	20	10	23.42	0.550	15.9

Table 9.7 Effect of Drum Level Controller Tuning Parameters

Test 1 A reasonable correlation exists between the reference trajectory time constant (30s) and the 63% response time of the controlled variable (35.0s).

Test 2 The reference trajectory time constant is increased to 40s, causing the plant response time to increase to 44.7s and the feedwater flow overshoot to decrease by 22%. As for Test 1, the actual response time is approximately 5s longer that the desired response time.

Test 3 The coincidence point position is changed from 15s to 20s. As a result, the feedwater flow overshoot decreases by 12% and the drum level response time increases to 37.6s.

Test 4 The reference trajectory time constant is increased to 40s and the coincidence point is moved to 20s. The plant response time (47.1s) decreases to 47.1s and the feedwater flow overshoot decreases by 30%. As for Test 3 the plant response time is approximately 7s longer than the desired response time.

Test 5 The reference trajectory time response is decreased to 20s and the coincidence point is placed at 10s. This causes the plant response time to decrease to 23.42s, the feedwater flow overshoot to increase by 77% and the *ISU* index to increase by 101%.

As for pressure, a close relationship exists between reference trajectory time constant and plant time response. The position of the coincidence point effects high frequency controller activity and has a lesser effect on plant response time. In the case of Test 3 and Test 4, a 5s increase in the position of the coincidence point increases the plant response time by approximately 2s.

The parameters used in Test 5 generate the fastest system response and result in the greatest controller activity. The parameter set used in Test 3 is more attractive - system response is slower but controller action has been reduced considerably.

Main Steam Temperature

The system response of main steam temperature and the corresponding attemperating flow rate are shown in Fig. 9.10 and Fig. 9.11 respectively, for each combination of tuning parameters.

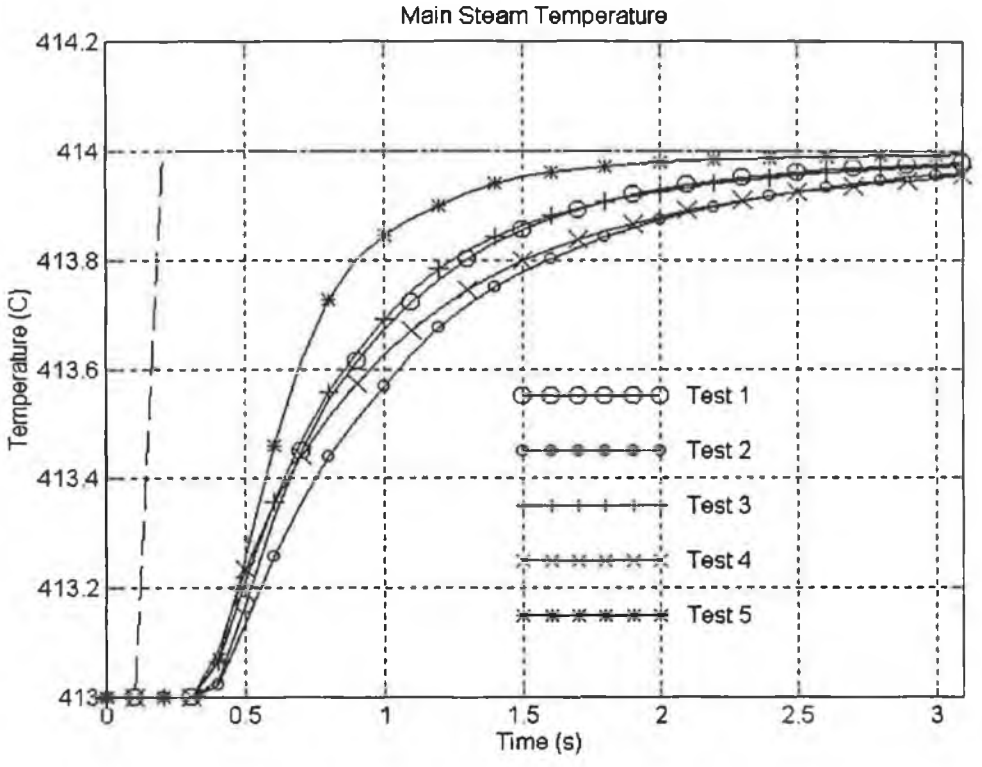


Fig. 9.10 Response of Main Steam Temperature to Step Change in Setpoint

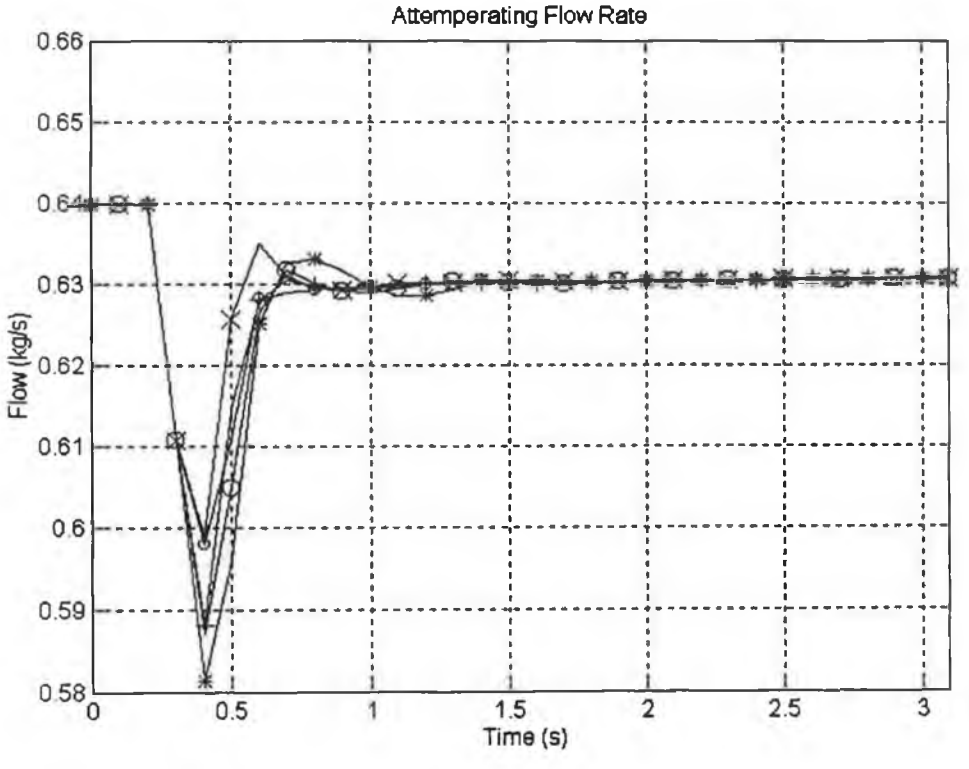


Fig. 9.11 Response of Attemperating Flow Rate to Step Change in Setpoint

Again the five system responses can be classified as slow, medium or fast, even though the separation between the three groups is less clear than in the case of main steam pressure or drum level.

Tuning for Main Steam Temperature Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Attemperating Flow Undershoot (kg/s)	ISU $(kg/s)^2$
1	0.6	0.3	0.83	4.68e-2	4.6e-3
2	0.8	0.3	1.01	3.60e-2	3.4e-3
3	0.6	0.4	0.80	4.59e-2	4.0e-3
4	0.8	0.4	0.91	3.50e-2	2.8e-3
5	0.4	0.2	0.61	5.25e-2	5.7e-3

Table 9.8 Effect of Main Steam Temperature Controller Tuning Parameters

Test 1 A response time of 0.83s is obtained given a desired response time of 0.6s.

Test 2 The desired response time is increased by 0.2s, causing the actual response time to increase by approximately 0.2s, attemperating flow undershoot to decrease by 23% and the *ISU* index to decrease by 26%.

Test 3 The position of the coincidence point is moved from 0.3s to 0.4s. This should result in less active control and a longer response time. In fact, the response time decreases by 0.03s even though the attemperating flow undershoot also decreases by 2%.

Test 4 Both the default desired response time and coincidence point position are increased by 33%. A comparison with Test 2 is interesting. The response time for Test 4 is 0.91s i.e. shorter than the response time of 1.01s achieved for Test 2. However, the controller action has also decreased marginally - undershoot has decreased by 3%. This illustrating the importance of correctly choosing the coincidence point position.

Test 5 Both the default desired response time and coincidence point position are decreased by 33%. As for Test 1, Test 2 and Test 3 the actual response time is approximately 0.2s greater than the desired response time. Attemperating flow undershoot increases by 12%.

In each case, the response time of main steam temperature has been specified by the reference trajectory time constant to within 0.25s. It is interesting to note that in two instances, increasing the position of the coincidence point can decrease both controller activity and plant time response.

The most attractive parameter set is that used in Test 5. It generates the fastest response time without requiring a large increase in controller activity.

It is interesting to compare the performance of the linear predictive controller and the nonlinear predictive controller for the controller tuning parameters of Test 5. In the case of the linear predictive controller, large oscillations of the control signal occur when this set of parameters is used. In the case of the nonlinear predictive controller, this set of controller tuning parameters do not cause large oscillations in the control signal. This can be explained by comparing the internal controller models of the linear and nonlinear predictive controllers.

- In the case of the linear predictive controller, the internal controller model is obtained by linearising and discretising a nonlinear first-principle plant model. It was shown in Section 8.5.1 that the frequency response of the discretised linearised model is inferior to that of the continuous linearised model at high frequencies. As the desired system response time decreases, the accuracy of the internal controller model also decreases. Ultimately, this leads to the oscillations which are evident in the control signal of the linear predictive controller for Test 5.

- In the case of the nonlinear predictive controller, the internal controller model is a neural network model, trained to predict the plant output at the next sample period i.e. the neural network model is a discrete model. The neural network model is capable of modelling the plant accurately for both low and high frequency input signals provided that the training data provides sufficient examples of both low and high frequency input data. It was shown in Section 6.4.2.2 that the training data was generated by combining low frequency variable amplitude, variable frequency square-type waves with high frequency noise. The performance of the nonlinear predictive controller for all the controller parameter tuning sets confirms that the neural network model is accurate over a wider frequency range than the discretised linearised model used by the linear predictive controller.

Percentage O_2 in Stack Gases

The step response of the measured percentage O_2 in the stack gases is shown in Fig. 9.12 for each combination of the controller tuning parameters. The corresponding air flow rate signal is shown in Fig. 9.13.

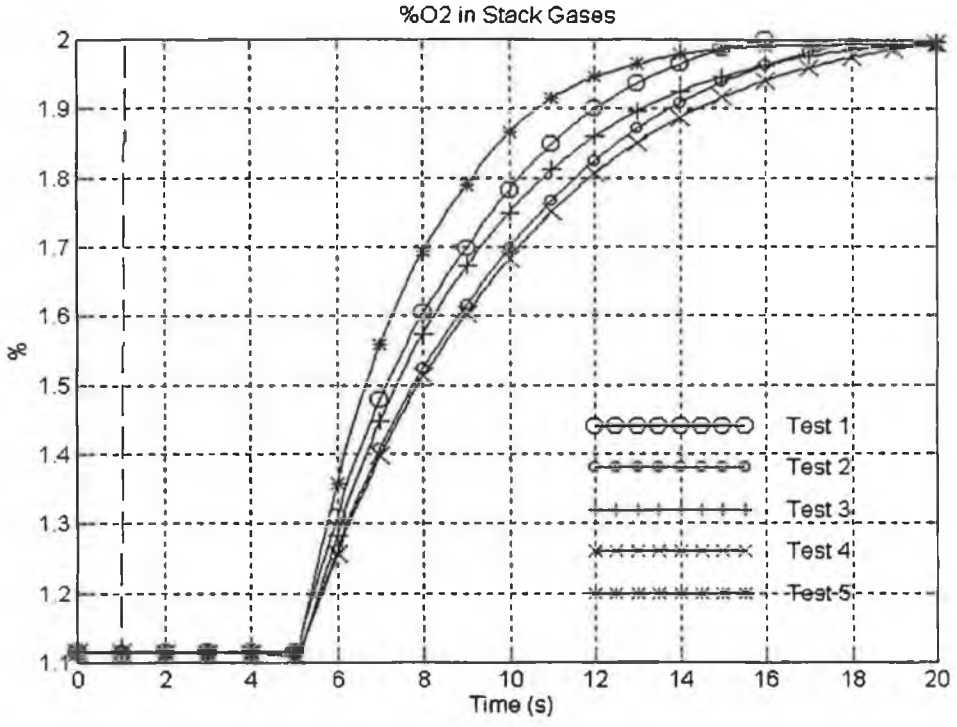


Fig. 9.12 Response of Percentage O_2 in Stack Gases to Step Change in Setpoint

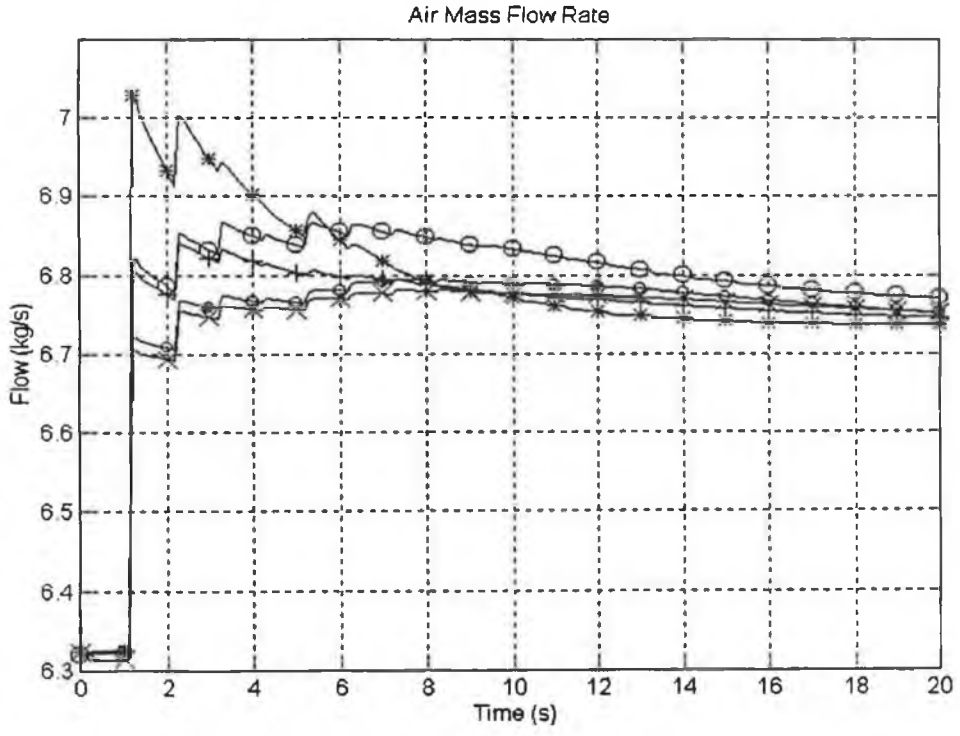


Fig. 9.13 Response of Air Flow Rate to Step Change in Setpoint

As for main steam pressure and main steam temperature, the five different system responses are separable into three distinct groups of fast, medium and slow responses.

Tuning for Percentage O_2 Control					
Controller Tuning Parameters			Performance Indices		
Test Nr.	Reference Trajectory Time Constant (s)	Position of Coincidence Point (s)	Actual 63% Response Time (s)	Feedwater Flow Overshoot (kg/s)	ISU $(kg/s)^2$
1	3.6	1.8	7.69	1.05e-1	10.0e-1
2	4.8	1.8	8.66	4.90e-2	5.6e-1
3	3.6	2.4	7.99	9.49e-2	6.3e-1
4	4.8	2.4	8.84	3.52e-2	4.9e-1
5	2.4	1.2	6.81	2.83e-1	20.4e-1

Table 9.9 Effect of Percentage O_2 Controller Tuning Parameters

Test 1 A response time of 7.69s is obtained using a desired response time of 3.6s. The large discrepancy between desired and actual response time is due to a delay of approximately 4s in percentage O_2 measurement. The response time excluding delay is approximately 3.69s.

Test 2 The desired response time is increased to 4.8s. The actual response time excluding delay is 4.66. Air flow overshoot decreases by 53% and the *ISU* index by 44%.

Test 3 The position of the coincidence point is changed to 2.4s.. This causes the response time to increase by 0.3s and the air flow overshoot to decrease by 10%. This is almost identical to the response of Test 1, indicating that the position of the coincidence point has a minor effect in this situation.

Test 4 The default desired time response is increased to 4.8s and the coincidence point position is changed to 2.4s. This causes the response time, excluding delay to change to 4.84s and air flow overshoot to decrease by 66%. This system response is very similar to that achieved in Test 2, again indicating that the position of the coincidence point mainly effects high frequency control action.

Test 5 The default desired time response is decreased to 2.4s and the coincidence point position is changed to 1.2s. This causes the response time, excluding delay to change to 4.84s. and air flow overshoot to increase by 169%.

In all cases there is excellent agreement between the desired plant response time and the actual plant response time. The minimum difference between desired and actual plant response time is 0.04s (Test 4) and the maximum is 0.33s (Test 3). The position of the coincidence point has a marginal effect on both plant response time and maximum air flow overshoot. This is most likely due to the fact the with the exception of Test 5, control action is sluggish. The position of the coincidence point would have a greater effect if the tuning parameters specified a fast response time. The controller action exhibits a spike-like effect every second. This is caused by variations in the fuel flow control signal. The fuel flow control signal is updated every second and acts as a disturbance input to the percentage O_2 controller.

The tuning parameters used in Test 5 offer a satisfactory compromise between controller action and system response. In all other cases, the controller action is very sluggish and approaching the open-loop response time in the case of Test 2 and Test 4.

The tests confirm that the controller tuning parameters are an effective and accurate means of specifying the response time of the controlled variables. In particular, the response time of the controlled variable is closely related to the response time of the reference trajectory. Specifying a slower reference trajectory response time reduces controller action and consequently increases the system response time.

An examination of the test results highlighted particular parameter settings which can achieve a good compromise between system response and controller action. This "improved" set of tuning parameters is listed in Table 9.10.

Controlled Variable	**Test Nr.**	**Time Constant of Reference Trajectory (s)**	**Coincidence Point (s)**
Main Steam Pressure (for slow response)	3	24	16
Main Steam Temperature	5	0.4	0.2
Drum Level (for slow response)	3	30	20
$\%O_2$ in Stack Gases	5	2.4	1.2

Table 9.10 Suggested Tuning Parameters

A comparison of the "suggested" parameter set with the tuning parameters for the 90% linear controller shows that the only difference between the two parameter sets is for the main steam temperature tuning parameters. In the case of the 90% linear controller, a slower response time is specified i.e. a desired reference trajectory of 0.6s and a coincidence point position of 0.4s is used. In order to facilitate comparison between the linear and nonlinear controllers, it is desirable to use the same main steam temperature tuning parameters for both the linear and nonlinear controller. In the case of the nonlinear controller, an examination of the response of main steam temperature to a set-point change (Fig. 9.10) indicates that good control performance can be attained using either the fast or slow tuning. In the case of the linear controller, an examination of the response of main steam temperature to a set-point change (Fig. 8.13) indicates that the control performance becomes very oscillatory when using the faster parameter set (i.e. reference trajectory equal to 0.4s and coincidence point position of 0.2s).

It follows that the slower parameter set could be adopted by both controllers, but not the faster parameter set. Thus implies that the performance of the nonlinear controller is to be compromised marginally if the same (slow) main steam temperature tuning parameters are to be adopted by both the 90% linear and nonlinear controllers. The extent to which main steam temperature response is compromised is investigated by simulating the response of main steam temperature using both the "improved" parameter set for the nonlinear controller and the tuning parameters used by the 90% linear controller.

The response of main steam temperature is simulated for a step change in load using both parameter sets and plotted in Fig. 9.14. using the following legend:

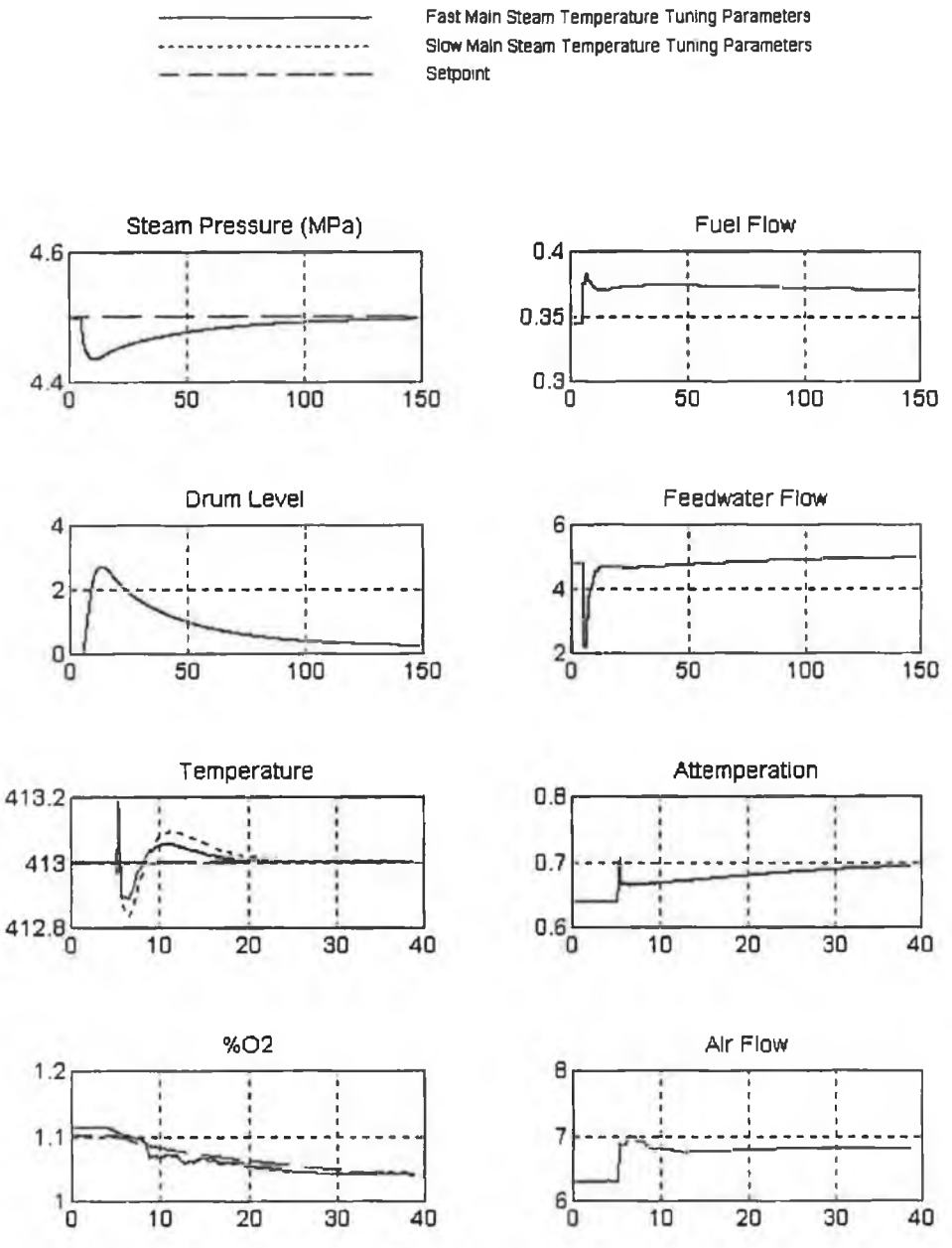


Fig. 9.14 Response to a Step Change in Load at 90% Using Two Sets of Controller Tuning Parameters

As expected, the response of main steam temperature is less tightly controlled using the "slow" parameter tuning. However, as the difference is very small (less than 0.05°C), the "slow" tuning parameters can also be adopted for the nonlinear controllers.

9.4.2 Selection of Tuning Parameters at 50% Operating Point

The set of controller tuning parameters adopted for the 90% linear controller are used as an initial choice of tuning parameters for the 50% linear controller. A good system response is obtained at 50% using this parameter set so this parameter set is also adopted for the 50% linear controller. The system response to a 5% step change in load at 50% using this set of parameters is shown in Fig. 9.15.

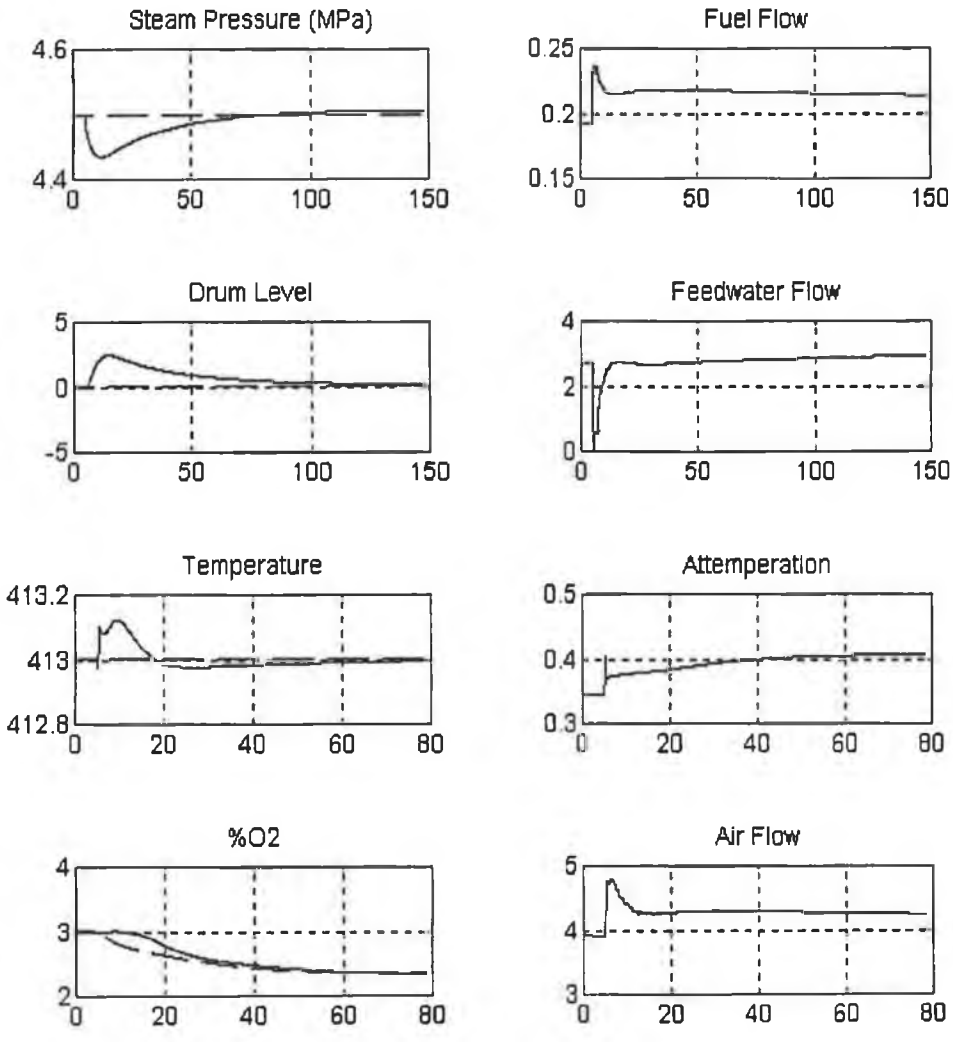


Fig. 9.15 System Response to a 5% Step Change in Load at 50% Load

9.4.3 Selection of Tuning Parameters at 10% Operating Point

The response to a 5% step change in load disturbance at 10% is shown in Fig. 9.16. The controller tuning parameters are the same as for the 50% and 90% operating point.

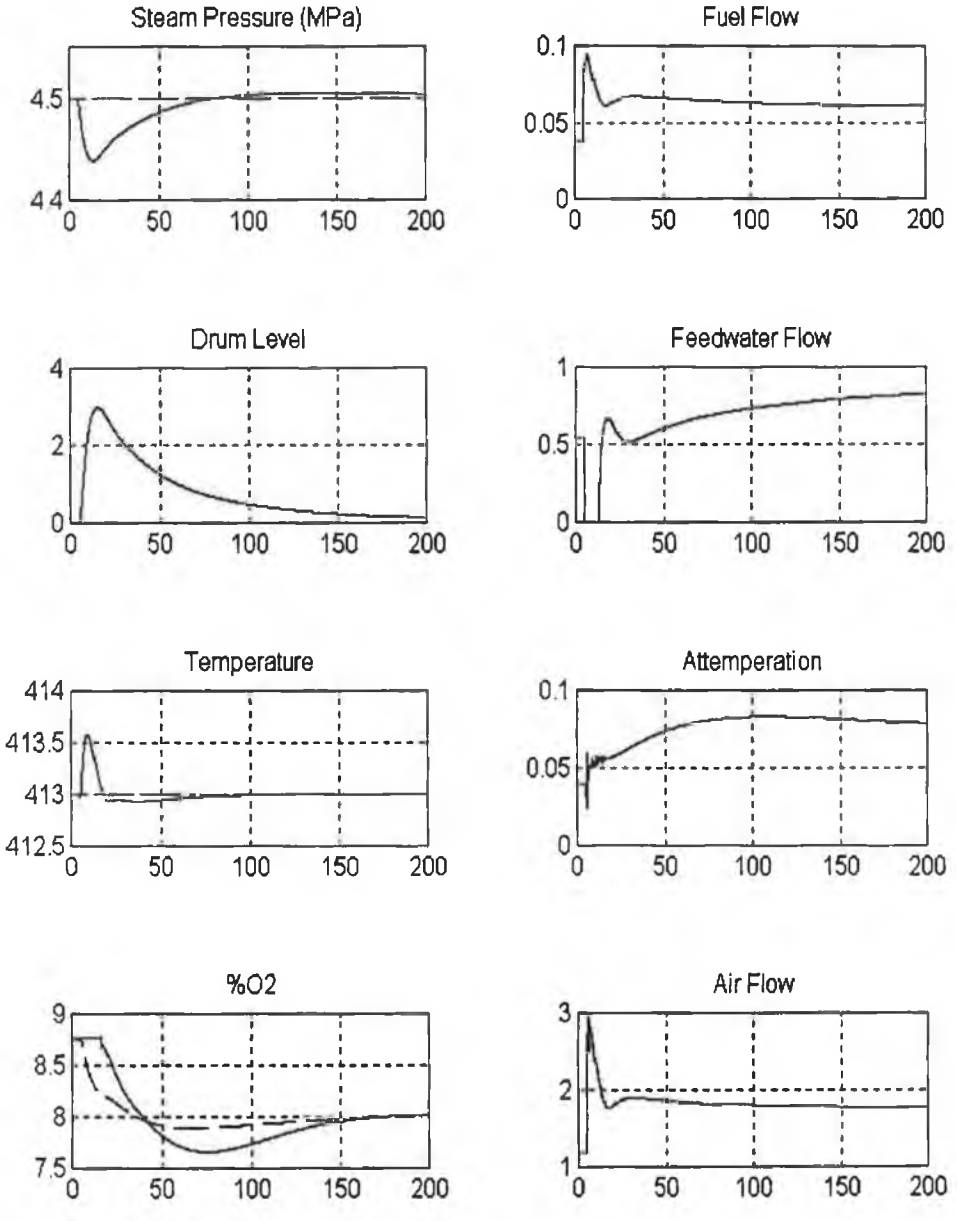
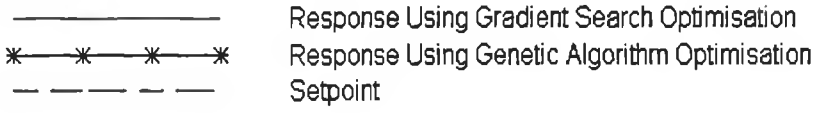


Fig. 9.16 System Response to a 5% Step Change in Load at 10% Load

9.5 Comparison of Gradient Search and Genetic Algorithm Optimisation Performance

The performance of each of the nonlinear controllers has been demonstrated for a 5% step change in load at the following loads - 10%, 50% and 90%. The control signal in all cases has been calculated using a conventional, gradient search optimisation method. The controller performance using gradient search optimisation method is compared to that using genetic algorithm optimisation for a 5% step load disturbance at a load of 70%. The system response is shown in Fig. 9.17 and uses the following legend.



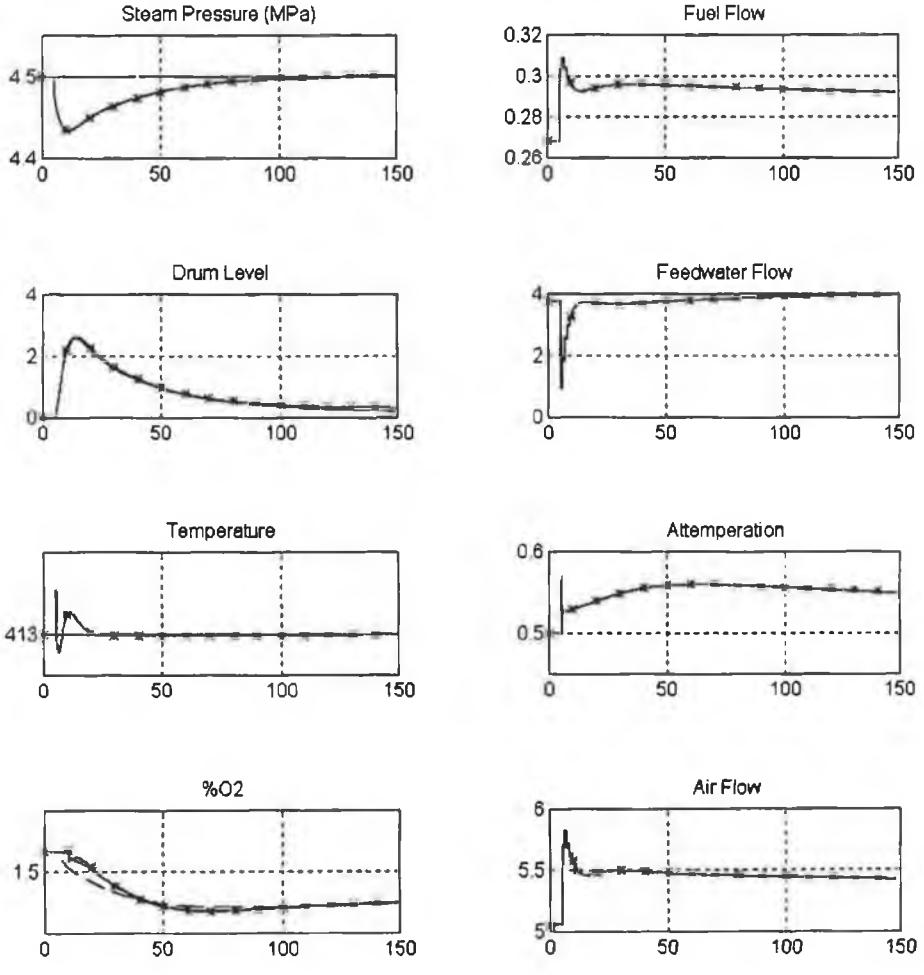


Fig. 9.17 System Response to a 5% Step Change in Load at 70% Load Using Gradient Search and Genetic Algorithm Optimisation

The system response in both cases demonstrates the same attractive features as that obtained using the fuzzified linear predictive controller. In particular, the predictive controller starts to increase feedwater flow before drum levels starts to decrease. This is necessary to prevent the drum level dropping below the setpoint in the after the transient *"Shrink and Swell"* effect has dissapeared. Such action is possible because the controller has explicit model-based knowledge about the expected behaviour of the planr.

There is little discernible difference between the system response obtained using a gradient search optimisation method and system response obtained using genetic algorithm optimisation. The reason for this can be understood by looking at a typical error surface (a plot of cost function *versus* proposed control action) for each of the three nonlinear controllers.

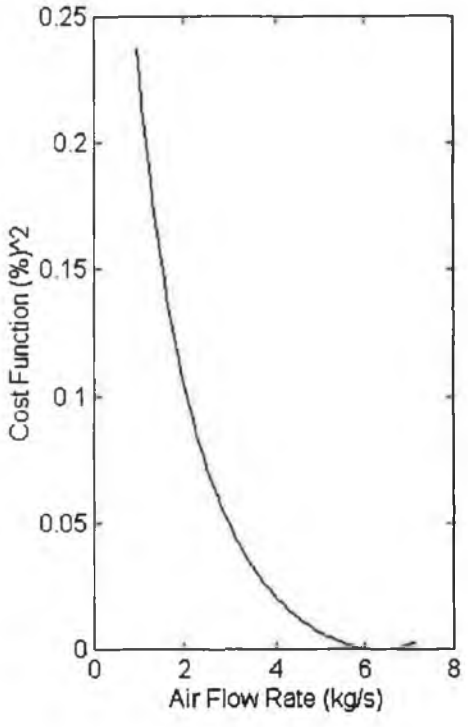


Fig. 9.18 Cost Function *versus* Air Flow for Percentage O_2 Controller

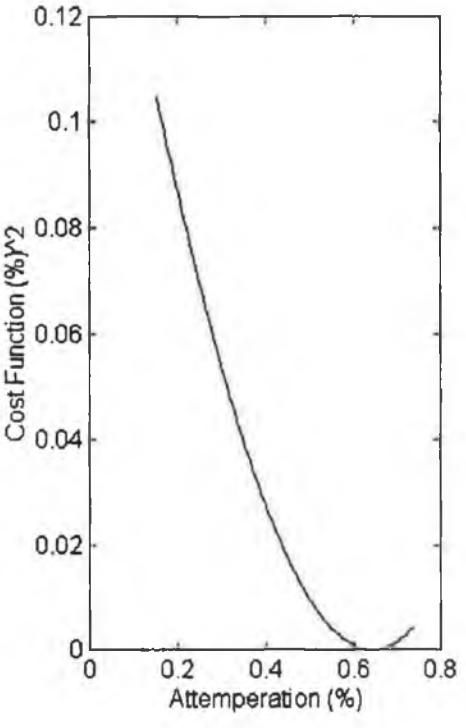


Fig. 9.19 Cost Function *versus* Attemperation for Main Steam Temperature Controller

In both cases, the error surface is very smooth and has no local minima. As a result the gradient search algorithm can be guaranteed to find the global minimum.

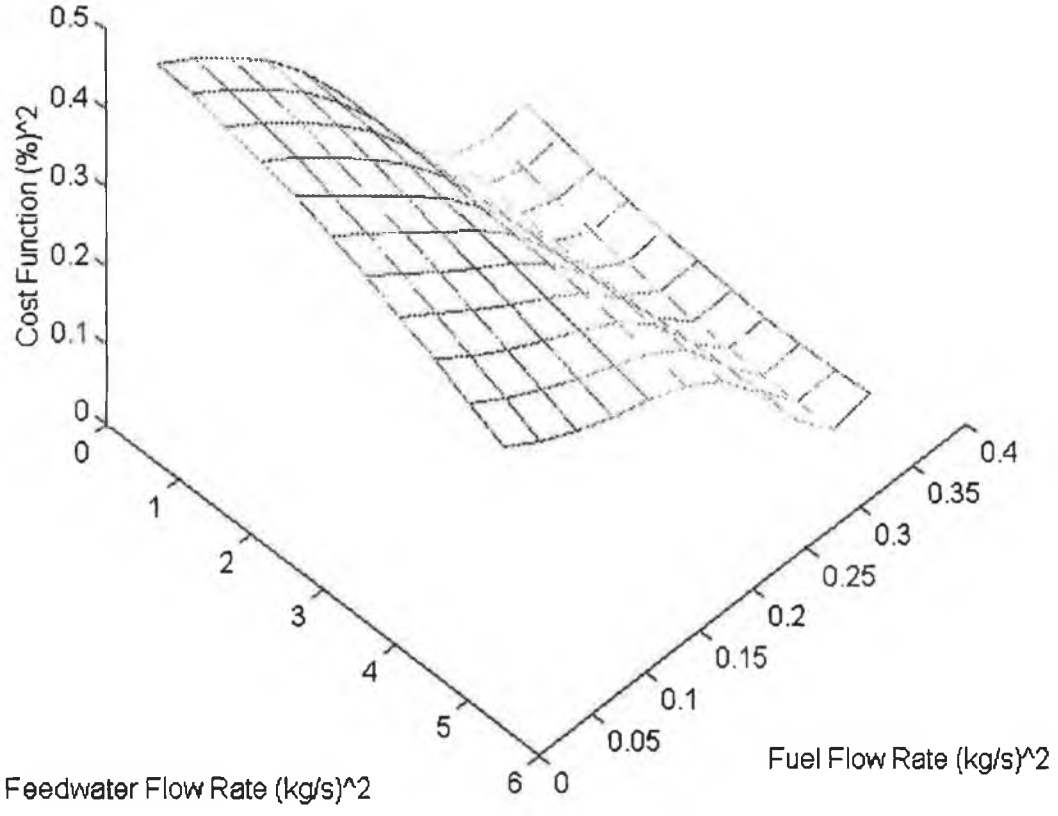


Fig. 9.20 Cost Function *versus*Feedwater Flow Rate, Fuel Flow Rate for Multivariable main Steam Pressure and Drum Level Controller

The situation is less straightforward for the main steam pressure and drum level controller. In this sample error surface, there is possibly one local minimum. Nevertheless, the error surface is very smooth and it is relatively unlikely that the gradient search algorithm could not find the global minimum.

The sample error surfaces suggest that it is not necessary to use genetic algorithms. However, these are very simple error surfaces - in the case of percentage O_2 and main steam temperature, the optimisation algorithms are required to minimise a function of one variable only. In the case of the multivariable main steam pressure and drum level controller, the optimisation algorithm is required to minimise a function of two variables only. If the control horizon is extended to more than one sampling period (i.e. the proposed control signal is not assumed to be a step function), it is to be expected that the contours of the error surface would become more complex and that a number of local minima could occur.

It is also of interest to compare the computational overhead of both optimisation methods. The number of million-floating point-operations (MFLOP) per second of simulation time is compared

for both optimisation methods and presented in Table 9.11. In the case of the genetic algorithms, it can be seen than the number of MFLOP counted is very dependent on the termination criteria for optimisation. The MFLOP count is compared for three different types of termination criteria.

Optimisation Method	Target	Multivariable Main Steam Pressure and Drum Level Controller (MFLOP)	Main Steam Temperature Controller (MFLOP)	Percentage O_2 Controller (MFLOP)
Gradient Search	Convergence at minimum	1.20	0.02	0.07
Genetic Algorithms	60 generations	16.37	0.59	2.98
Genetic Algorithms	Target Error	10.21	0.34	0.59
Genetic Algorithms	Target Error, given previous set of solutions	8.87	0.07	0.40

Table 9.11 Comparison of MFLOP for Genetic Algorithm and Gradient Search Optimisation

The first and very simple termination criteria is that optimisation is terminated after 60 generations. In this case it can be seen that the gradient algorithms execute between 390% and 3600% more MFLOP than the gradient search algorithms. This termination criteria does not take into account that the genetic algorithms may have determined a good solution after just 1 or 2 generations, however. This can be taken into account by specifying that optimisation is discontinued when the genetic algorithms have achieved a cost function which is below a specified target. In the case of percentage O_2 controller, this reduces the number of MFLOP by 480%. The effect of changing the termination criteria to a target cost function is dictated to a large extent by the size of the specified target cost function and by the contours of the error surface. The target cost function is based upon the maximum allowable error between the desired prediction horizon and the predicted plant output at the coincidence point. The maximum allowable error specified for each of the controller variables is presented in Table 9.12.

Controlled Variable	Target Error
Main Steam Pressure	1 e-4 MPa
Drum Level	1 e-3 mm
Main Steam Temperature	1 e-3 °C
Percentage O_2 Controller	1e -3 %

Table 9.12 Target Error for Genetic Algorithm Optimisation

In the case of the multivariable main steam pressure and drum level controller, it cannot be

gauranteed that the target error is achieved for both variables as the cost function for a multivariable controller is a function of the combined target errors. For example, the predicted main steam pressure error may slightly exceed its target error if the predicted drum level error compensates for this by achieving a smaller error than the target error.

One more factor should be taken into consideration. For gradient search optimisation, the optimisation algorithm is provided with a good initial solution at each sampling period - the value of the current controller action. However, the genetic algorithm has been randomly initialised and may be at a computational disadvantage as a result. This can be investigated by including the current control signal into the initial set of solutions for the genetic algorithm. The last row of Table 9.11 show the effect of this on computational effort. For all three controllers, the computational effort has been significantly reduced - (by 45% for the multivariable main steam pressure and drum level controller, 79% for the main steam temperature controller and by 32% for the percentage O_2 controller. This is the termination criteria used to generate the results in Fig. 9.17.

Even in the "best case" scenario, however, the computation overhead of the genetic algorithms is substantially greater than that of the gradient search algorithm - 633% greater for main steam pressure and drum level controller, 337% greater for the main steam temperature controller and 497% greater for the percentage O_2 controller. This is not surprising as genetic algorithm optimisation requires that the cost function is evaluated at each sampling period for a set of 10 possible solutions whereas the gradient search optimisation algorithm must evaluate it for one solution only. The computational complexity of the genetic algorithms is only an issue if it can be shown that it is not to execute this algorithm within the sampling period. The total number of MFLOP required to run all three controllers for one second at their respective sampling rates is presented in Table 9.13. The "best-case" values are used for the genetic algorithm optimisation method.

Optimisation Algorithm	MFLOP/S
Gradient Search	2.1
Genetic Algorithms)	13.6

Table 9.13 MFLOPS/s for Gradient Search Optimisation and Genetic Algorithm Optimisation

Both of these controllers can be implemented on a Digital Signal Processing (DSP) board. The TMS320C30, for example, can execute 33.3MFLOP per second.

9.6 Conclusions

The results demonstrate the feasibility of controlling a boiler process using a nonlinear predictive control methodology. In particular it demonstrates the feasibility of using a single neural network model to act as the internal controller model over the entire operating range of the boiler.

Controller performance has been compared using a conventional gradient search optimisation method and a global optimisation technique i.e. genetic algorithm optimisation. Controller performance is found to be be identical for both optimisation algorithms. This implies that the error surface is relatively smooth and that the conventional gradient search optimisation technique is capable of finding the global minimum under the tested conditions. Results also indicate however that the genetic algorithms have a much greater computational overhead than the gradient search algorithm. This is not a significant advantage, however, since the results shown that the control strategy can be executed on a standard PC using either optimisation method within the sampling time constraints. Given that gradient search optimisation and genetic algorithm optimisation can achieve equal controller performance, the choice of optimisation algorithm is essentially dicatated by computational constraints and by the importance of achieving a global solution. If speed of execution is important, the gradient search algorithm is the obvious choice. However, if execution speed is not important, genetic algorithms may prove to be a "safer" solution.

10 Comparison of PI, Fuzzified Linear and Nonlinear Predictive Control

10.1 Introduction

The boiler process is complex multivariable system with both static and dynamic nonlinearities. Given that conventional boiler control strategies rely upon linear control techniques, it is feasible that improved control could be achieved through the application of a nonlinear predictive control strategy.

A nonlinear controller can be developed by extending or modifying an essentially linear control strategy. For example, a nonlinear controller may be based upon a single linear model of the plant if the parameters of the model are updated with changes in the plant operating point. The fuzzified linear predictive controller, described in Chapter 8, is an example of a nonlinear controller which is based upon a linear controller strategy. This controller is based upon three linear controllers, designed to operate at the 10%, 50% and 90% operating points. Fuzzy logic interpolation is used to combine the output of these three linear controllers so as to provide equal control performance over the entire operating range of the plant. Alternatively, a nonlinear controller may be based upon a dedicated nonlinear model of the plant. The nonlinear predictive controller, described in Chapter 9, is of this type. It is possible that a dedicated nonlinear model can model the plant more accurately than a piece-wise linear model and thus yield more accurate controller performance.

To date, PI control is widely used for boiler control and is certainly the obvious benchmark strategy for any assessment of controller performance. The implementation and tuning of the PI control strategy used in this comparison are described fully in this chapter.

The best choice of control strategy for any system is likely to depend on a wide range of factors. The most obvious issue is control performance - this must be assessed according to a number of criteria and under a variety of different conditions. A second important issue to be considered for an advanced control strategy is that of feasibility , whether the controller can be executed within the specified sampling period. Both of these issues are addressed within this chapter.

10.2 PI Control of a Drum-Type Boiler

10.2.1 Development of PI Control Strategy

Conventional boiler control employs SISO PI controllers, which use a single manipulated variable to regulate each of the controlled variables. In this instance there are four variables to be controlled: main steam pressure, drum water level, main steam temperature and percentage of O_2 in the stack gases. Likewise, there are four variables which may be manipulated: fuel flow rate, feedwater flow rate, attemperating water flow rate and air flow rate. A description of standard, industrial boiler PI control is given in Chapter 3.

Implementation of a PI control strategy first requires that a suitable manipulated variable must be selected to regulate each of the controlled variables. This requires knowledge about the relationship between the manipulated variables and controlled variables. For example, feedwater flow rate has a very strong effect on drum level, and is a suitable choice as manipulated variable for the drum level controller. It is not difficult to choose the best manipulated variable to regulate each of the four controlled variables, as there exists four noticeably strong relationships between the controlled and manipulated variables - pressure is strongly affected by fuel flow rate, drum level by feedwater flow rate, steam temperature by attemperating water flow rate and the percentage of O_2 in stack gases by air flow rate. This combination of controlled and manipulated variables is very common among boiler PI control strategies.

Some boiler PI strategies may include additional controlled and manipulated variables, such as reheater temperature and burner tilt respectively. In addition, some boilers may use an alternative strategy to control steam temperature, such as allowing some of the steam to bypass the superheater (in place of attemperating water). The particulars of each strategy is dependent on the construction of the boiler. However, the four controlled variables described here are fundamental to boiler control and are included in all boiler control strategies.

Table 10.1. lists the four controlled variables and their corresponding manipulated variables employed in the PI control strategy.

Controller	Controlled Variable	Manipulated Variable
1	Steam pressure	Fuel flow
2	Drum level	Feedwater flow
3	Steam temperature	Attemperating flow
4	Percentage of O_2 in stack gases	Air flow

Table 10.1 Controller and Manipulated Variables for each PI Controller

The overall control strategy is represented schematically in Fig. 10.1

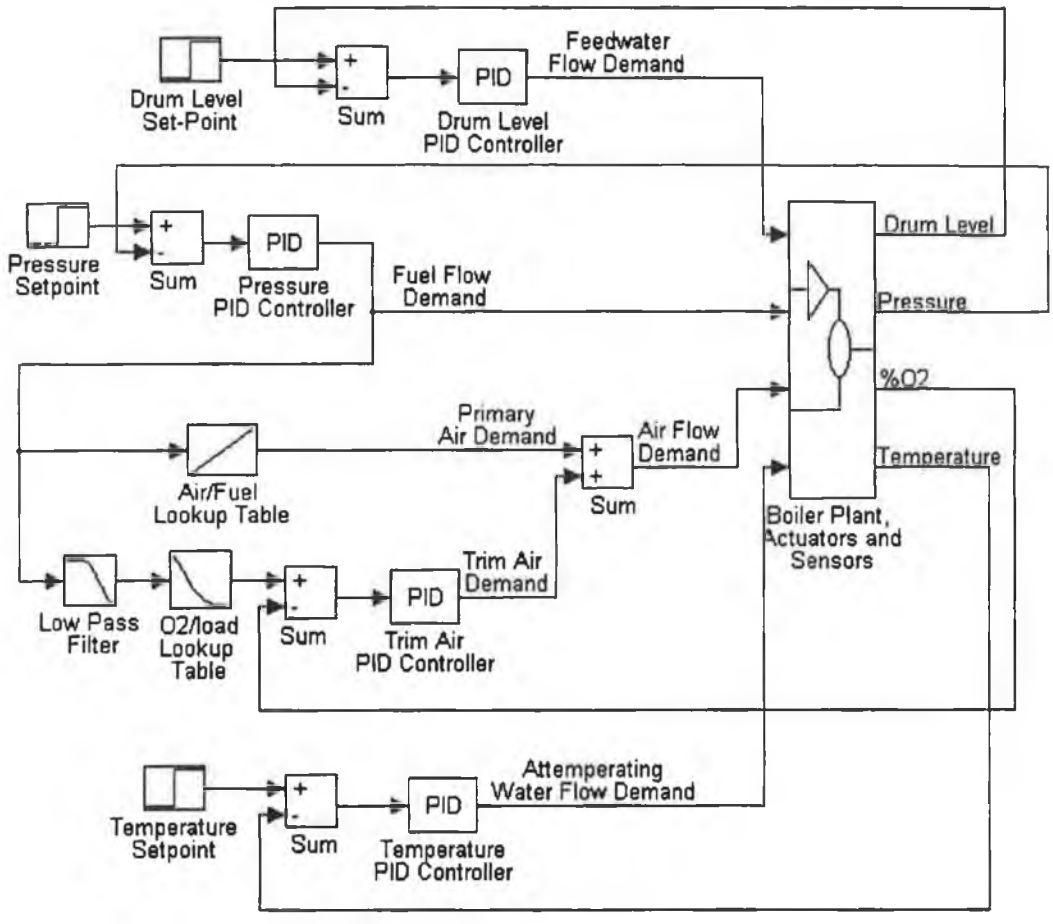


Fig. 10.1 Boiler PI Control Strategy

It can be seen from Fig. 10.1 that air flow demand signal is not wholly dictated by the trim air controller. The output of the trim air controller is summed with a feedforward signal, termed the primary air flow demand signal which is directly related to the fuel flow rate. The primary air demand signal is necessary for safety reasons in a boiler PI control strategy. If fuel flow increases, the air flow must immediately be increased. However, the trim air controller does not react until measured level of percentage O_2 in the stack gases drops. This is dangerous as there is a significant delay between the decrease of percentage O_2 in the combustion gases and the decrease of measured percentage O_2 in the stack gases. Consequently, the percentage of O_2 in the combustion gases may remain very low or even drop below zero for some time, resulting in unburned fuel and the production of poisonous carbon monoxide gases. To prevent this scenario, the percentage O_2 controller output signal is summed with primary air demand signal. The primary air demand signal is a function of fuel flow and usually constitutes 90% of the total air demand signal. The trim air controller is responsible for fine tuning the air flow demand signal to keep measured percentage O_2 at its setpoint.

A primary air demand signal is not necessary for the fuzzified linear or nonlinear controller of percentage O_2 as both of these controller include a fuel flow disturbance model. The disturbance model allows the controller to react immediately and appropriately to variations in fuel flow.

The PI controllers used to control the boiler process are analog. This is analogous to the widely adopted practice of implementing digital PI controllers with a very fast sampling rate in order to attain practically the same performance as an analog PI controller.

10.2.2 Tuning of PI Controllers

Each of the PI controllers is tuned on a trial and error basis. The effect of tuning parameter variations is evaluated around the operating point for a 10% load. Plant dynamics are slower at low loads, so the resulting PI parameters should result in stable control at all loads. The final choice of controller tuning parameters is presented in Table 10.2.

Controller	Proportional Gain	Integrator Gain
Drum Level Controller	1.3	3e-2
Main Steam Pressure Controller	4e-7	2e-8
Main Steam Temperature Controller	-3e-2	-3e-3
Trim Air Demand Controller	1e-1	8e-3

Table 10.2 Controller Tuning Parameters for Boiler PI Control Strategy

It should be noted that the same PI controller parameters are used at all operating points. It is possible that controller performance could be improved by making the tuning parameters a function of plant operating point. This is not implemented here as it is not standard practice for industrial PI control strategies however, which is the benchmark for the development of this PI control strategy.

The same controller tuning parameters are also used at all operating points for the fuzzified linear and nonlinear predictive controllers, but the inclusion of an internal controller models in these controllers does ensure that the controller behaviour is automatically adapted to the plant operating point.

It can also be noted that the controller parameter values for the main steam temperature controller are negative. This is explained by the negative gain relationship between attemperation and main steam temperature i.e. an *increase* in attemperation produces a *decrease* in main steam temperature.

10.3 Comparison of Controller Performance

10.3.1 Description of Comparison Criteria

Controller performance is assessed by subjecting each controller to tests which are designed to demonstrate the ability of the controller under different conditions. Two types of tests are used:

- Step Load Disturbance Tests
- Step Setpoint Change Tests

In order to test the performance of each of the controllers over the full operating range of the plant, the tests are carried out at each of the following operating points - 10%, 30%, 50%, 70% and 90%. The performance of each of the controllers under these tests may be objectively compared using suitable performance indices. Six performance indices have been selected which provide a good measure of controller performance under a variety of conditions. The first four performance indices provide a measure of system response. The last two performance indices provide a measure of controller action.

1. Integral Squared Error - *ISE*

   This is defined as the sum of the squares of the error between the controlled variable and the setpoint over a defined time-span.

$$ISE = \sum_{k=k_{min}}^{k=k_{max}} \left(SP(k) - y(k) \right)^2 \qquad (10.1)$$

   where

   SP = Setpoint

   y = Controlled variable

2. Maximum Deviation from the Setpoint - DEV_{max}

   This is equal to the maximum absolute deviation of the controlled variable from the setpoint, following a disturbance.

$$DEV_{max} = \max(|SP - y|) \qquad (10.2)$$

   This index is not applicable for setpoint tests.

3. Response Time - T_{res}

   The response time is defined as the time required for the controlled variable to reach a specified region around the setpoint - y_{res}. In the case of a setpoint change this value is taken to be 37% of the difference between the new and the old setpoint. The response time is

measured from the time at which the setpoint change occurs.

$$y_{res} = SP_{new} - 0.37(SP_{new} - SP_{old}) \tag{10.3}$$

In the case of a load disturbance, y_{res} is specified in terms of the maximum deviation from the setpoint. The response time is measured from the time at which the disturbance occurs.

$$y_{resp} \approx SP \pm 0.37(DEV_{max}) \tag{10.4}$$

The maximum deviation from the setpoint following a load disturbance is dependent on the magnitude of the disturbance, the current operating point and to a lesser extent on the controller response. In order to perform a fair comparison between each of the three controllers, it is necessary to specify the same value of y_{res} for each of the three controllers. At each operating point, the maximum deviation from the setpoint is calculated for all three controllers for a 5% load disturbance. The mean maximum deviation of the fuzzified linear and the nonlinear predictive controller is then used to calculate a common value of y_{res}.

$$y_{res} = (y_{res_{lin}} + y_{res_{nl}}) / 2 \tag{10.5}$$

The maximum deviation of the PI controller is not included in the calculation of the mean maximum deviation because it greatly exceeds that of either of the predictive controllers. For example, in the case of main steam temperature, the maximum deviation for the predictive controllers will never exceed y_{res} if the maximum deviation for the PI controller is included in the calculation of y_{res}.

The value of y_{res} used for each of the controlled variables at each operating point for a 5% change in load is presented in Table .10.3

Controlled Variables	y_{res}				
	10%	**30%**	**50%**	**70%**	**90%**
Main Steam Pressure	23e-3 MPa	23e-3 MPa	23e-3 MPa	23e-3 MPa	23e-3 MPa
Main Steam Temperature	0.13 °C	0.12 °C	0.04 °C	0.03 °C	0.05 °C
Drum Level	1.1 mm	1.0 mm	1.0 mm	1.0 mm	1.0 mm
Percentage O_2 in the stack gases	0.26%	0.28%	0.10%	0.07%	0.01%

Table 10.3 Response Time Value for Controlled Variables

It can be seen from Table 10.3 that practically the same value of y_{res} can be used by main steam pressure and drum level at all 5 operating points. This not the case for the percentage

of O_2 in the stack gases or for main steam temperature. At lower operating points, the maximum deviation from the setpoint is greater and as a result a larger value of y_{res} is employed.

It can be seen that the response time is intended to correspond in an approximate manner to the 63% response time of the controlled variable.

4. Settling Time - T_{settle}

The settling time is defined as the time required by the controller variable to reach and stay with a specified tolerance band around the setpoint ($SP \pm y_{settle}$). In the case of a setpoint change, the tolerance band is ±5% of the difference between the old setpoint and the new:

$$y_{settle} = 0.05\left|SP_{new} - SP_{old}\right| \tag{10.6}$$

In the case of a disturbance the tolerance band is approximately ±5% of the maximum deviation from the setpoint:

$$y_{settle} \approx \pm 0.05(DEV_{max}) \tag{10.7}$$

In order to compare the three controllers it is necessary to use the same value of y_{settle}. for all three controllers in the case of a load disturbance. The value for y_{settle} is calculated from the corresponding value for y_{resp}.

$$y_{settle} = \frac{5}{37}(y_{res}) \tag{10.8}$$

The value of y_{settle} used for each of the controlled variables at each operating point is presented in Table 10.4.

Controlled Variables	y_{settle}				
	10%	**30%**	**50%**	**70%**	**90%**
Main Steam Pressure	3e-3 MPa	3e-3 MPa	3e-3 MPa	3e-3 MPa	3e-3 MPa
Main Steam Temperature	0.01 °C	0.01 °C	0.01 °C	0.01 °C	0.01 °C
Drum Level	0.1 mm	0.1 mm	0.1 mm	0.1 mm	0.1 mm
Percentage O_2 in the stack gases	0.05%	0.035%	0.01%	0.01%	0.01%

Table 10.4 Settling Time Value for Controlled Variables

5. Maximum Overshoot/Undershoot of Manipulated Variable - OS_{max}

This is the maximum overshoot or undershoot of the manipulated variable measured in relation to its final steady state value. It provides a measure of initial controller activity in response to a setpoint change or load disturbance.

$$OS_{\max} = \max(|u - u_{ss}|) \tag{10.9}$$

6. Integral Squared Control Action- ISU

This is equal to the sum of the squares of the deviation of the control signal from its mean value over a defined time span. It is a measure of total "energy" of the control signal over that time span.

$$ISU = \sum_{k=k_{\min}}^{k=k_{\max}} u(k) - \bar{u})^2$$

where (10.10)

$$\bar{u} = \frac{\sum_{k=k_{\min}}^{k=k_{\max}} u(k)}{k_{\max} - k_{\min} + 1}$$

10.3.2 Step Load Disturbances

Step disturbances in load are triggered by several common operations in the downstream plant - opening or closing a valve, switching on or off a steam consuming process. Step load disturbances cause a sudden increase (or decrease) in the plant energy reserves - the control system must react swiftly to match the new steam consumption requirement and maintain process variables such as drum level, steam pressure and temperature at their setpoints.

A 5% step load disturbance is applied to the plant at five different operating points - 10%, 30%, 50%, 70% and 90%. The response of each of the controlled variables is shown together with that of its associated manipulated variable for the test at the 70% operating point only. The response at the other operating points is shown by plotting the performance indices obtained at the 10%, 30%, 50%, 70% and 90% operating points.

Main Steam Pressure

Fig. 10.2a and 10.2b shows the response of main steam pressure and fuel flow rate to a 5% step change in load when the plant is at the 70% operating point. The following legend is used to distinguish the response of the three controllers:

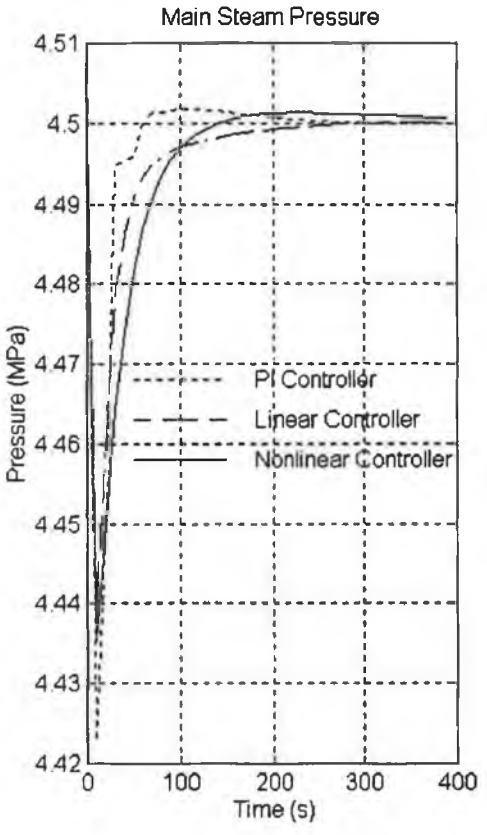


Fig 10.2a Response of Main Steam Pressure to a Step Change in Load

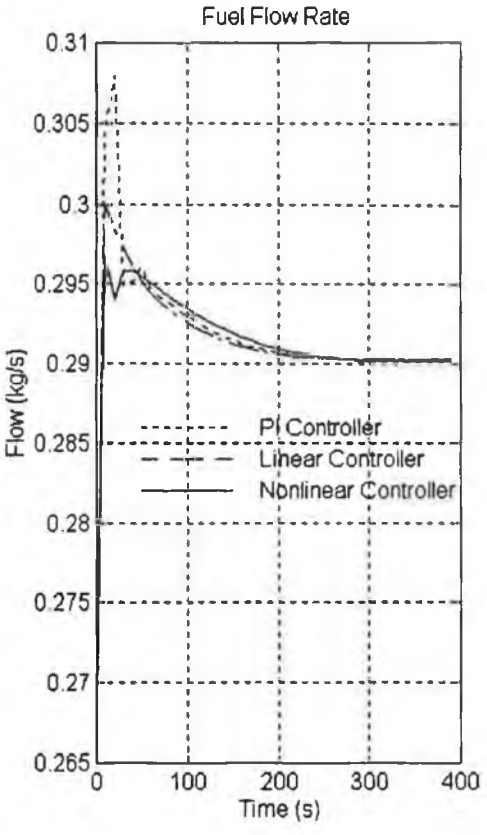


Fig 10.2b Fuel Flow Rate after a Step Change in Load

Good pressure control has been achieved by both the fuzzified linear and nonlinear predictive controllers. The predictive controllers increase fuel rapidly after the load disturbance to minimise the deviation of main steam pressure from the setpoint and then apply just sufficient fuel to move main steam pressure to its setpoint on a smooth exponential trajectory. This is a very economical response in terms of fuel costs. The PI controller injects a very large amount of fuel and consequently produces the fastest response. Despite this large injection of fuel however, the PI controlled system shows the greatest deviation of pressure from the set-point. This deviation could be reduced by increasing the proportional gain of the PI controller but this is likely to result in main steam pressure overshooting the setpoint. This is to be avoided if possible - it yields no benefits in terms of plant performance and requires fuel to be expended unnecessarily. The predictive control strategies reduce the maximum deviation by reacting immediately to the load disturbance. This immediate reaction is in fact a feedforward control action, which is driven by the effect of a load disturbance on the internal controller model.

This test highlights an important advantage of predictive control. It is designed to accurately

track a specified reference trajectory. This offers the possibility of specifying optimal reference trajectory i.e. a trajectory which optimises boiler efficiency or minimises stresses on the metal tubes in the boiler (Dieck-Assad *et al* (1987)).

Fig 10.3 presents the performance indices for main steam pressure for each controller for a 5% step load disturbance at 10%, 50% and 90%. The four plots on the top two rows are performance indices of system response. The two plots on the third row are indices of controller action. The following legend is used:

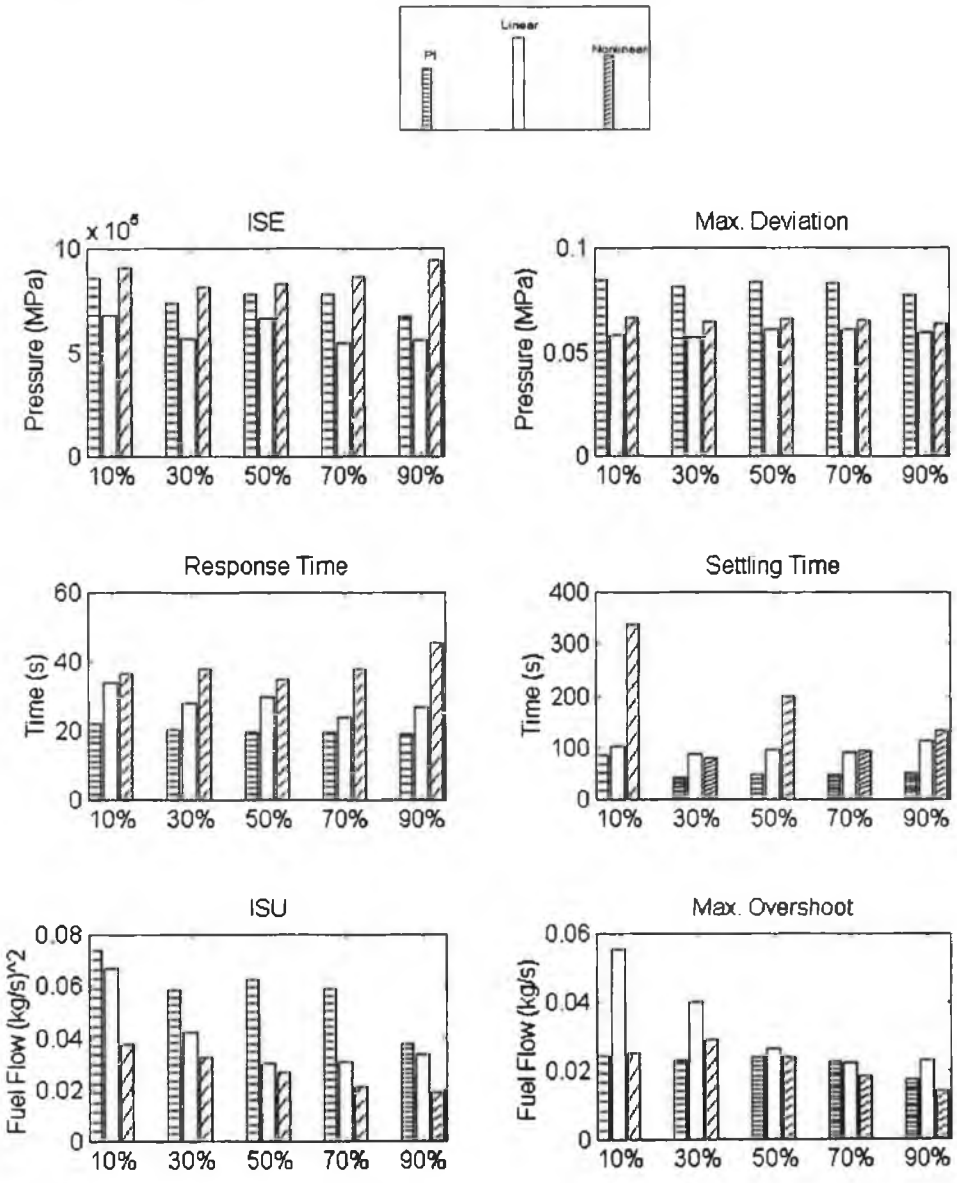


Fig 10.3. Performance Indices of Main Steam Pressure following a 5% Step Change in Load at 10%, 30%, 50%, 70% and 90% Operating Point

The average *ISU* index of the PI control strategy is approximately 400% greater than the average *ISU* index of the nonlinear predictive controller at all operating points. Despite this, the average *ISE* index of the PI controller is just 12% smaller than the average *ISE* index of the nonlinear predictive control strategy.

1. The fuzzified linear controller also applies a less energetic control signal than the PI control strategy, in terms of its *ISU* index but achieves a smaller *ISE* index at all operating points. This can be understood by comparing the maximum deviation of main steam pressure from the setpoint for both controllers. At all operating points, the PI control strategy records a considerably larger maximum deviation from the setpoint(approximately 30%) than either the fuzzified linear or nonlinear predictive controller. These large deviations from the setpoint are penalised quite heavily by the *ISE* index.
2. It can be seen that the settling time of the nonlinear predictive controller is noticeably slow at the 10% operating point. This is be explained by the fact that the neural network model of the nonlinear controller is considerably less accurate at the 10% operating point than at the 90% operating point. Integral action is used to move the plant output exactly to the setpoint, but this a lot less effective than predictive control action. To illustrate this, the settling time and the final steady-state value of the integrator output at the 10% operating point are compared to the same values at the 50% and 90% operating points. The final steady-state value of the integrator output is comparable to the prediction error of the nonlinear model.

Index	**Operating Points**		
	10%	**50%**	**90%**
Settling Time (s)	338	199	133
Steady-State Prediction Error (MPa)	4.5e-3	2.5e-3	7.2e-4

Table 10.5 Settling Time and Integrator Output of Nonlinear Controller at 10%, 50% and 90% Operating Points

Table 10.5 confirms that the settling time of the controller and its final steady-state value of integrator output (prediction error) are strongly related. The integrator output is greatest at the 10% operating and smallest at the 90% operating point. Likewise, the settling time is longest at the 10% operating point and shortest at the 90% operating point.

In order to provide an overall measure of controller performance, the mean *ISE* and *ISU* indices is first calculated for each of the three controllers. For example the mean *ISE* (*MISE*) index for the PI control strategy is calculated as follows:

$$MISE_{PI} = mean(ISE_{10} + ISE_{30} + ISE_{50} + ISE_{70} + ISE_{90}) \tag{10.11}$$

The *MISE* and mean *ISU (MISU)* indices of all three controllers are then calculated as a percentage of the largest *MISE* and *MISU* index of the three controllers. For example the percentage *MISE (PMISE)* index for the PI controller strategy is calculated as follows:

$$PMISE_{PI} = \frac{MISE_{PI}}{\max(MISE_{PI}, MISE_{LINEAR}, MISE_{NONLINEAR})} \times 100 \qquad (10.12)$$

Ideally, both the *PMISE* index and the *PMISU* index should be as low as possible. A good control strategy should apply controller action as effectively as possible in order to minimise tracking error. To generate one single measure of controller performance the mean of the *PMISE* and *PMISU* index of each controllers is calculated and normalised with respect to the largest calculated value. The resulting value, termed the "*Pressure Control Index*" is a measure of the overall performance of that controller with respect to the control of main steam pressure.

The *PMISE*, percentage *MISU* (*PMISU*) and Pressure Control Index are presented in Table 10.6.

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
PMISE (%)	87	69	100
PMISU (%)	100	70	47
Pressure Control Index (%)	100	74	77

Table 10.6 Indices of Controller Performance for Main Steam Pressure (in Response to a 5% Step Change in Load)

The indices confirm that the PI control strategy did not apply controller action to best possible effect. For example, the *PMISU* index of the PI controller is 53% greater than of the nonlinear controller, even though the *PMISE* index of the nonlinear controller is just 13% greater than that of the nonlinear controller. The fuzzified linear controller achieves the minimum *Pressure Control Index*, and the nonlinear controller also achieves a similar rating.

In summary, by this measure, fuzzified linear predictive control can improve main steam pressure control by 26% in comparison to conventional PI control. Nonlinear predictive control can improve main stream pressure control by 23%.

Main Steam Temperature

Fig. 10.4a and Fig. 10.4b shows the response of main steam temperature and attemperating flow rate respectively to a 5% step load disturbance at the 70% operating point

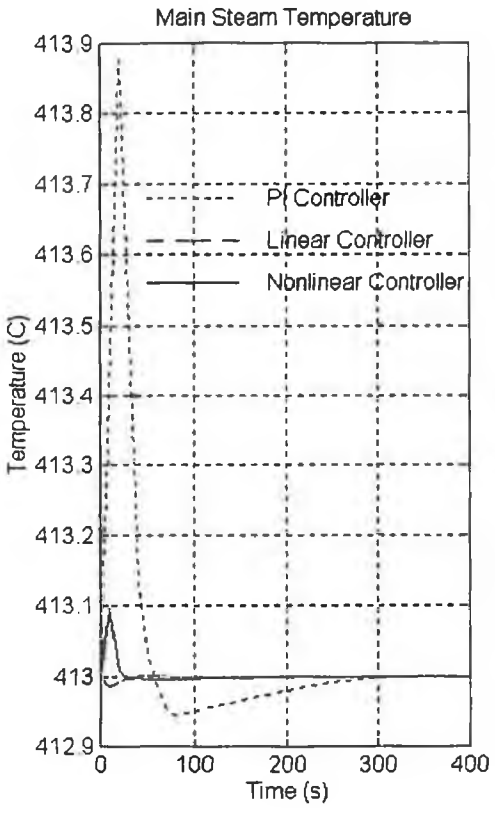


Fig 10.4a Response of Main Steam Temperature to a Step Change in Load

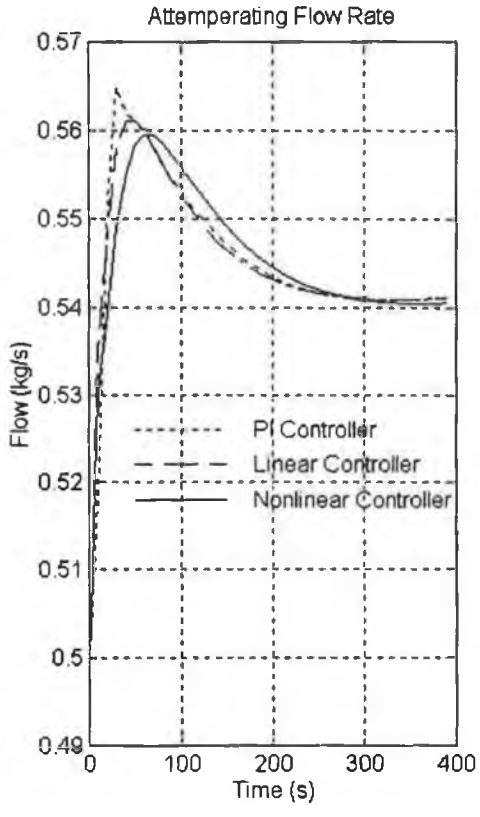


Fig 10.4b Attemperating Flow Rate after a Step Change in Load

A step change in load triggers two distinct effects on main steam temperature. The first effect is a non-minimum phase response in main steam temperature. A step increase in load causes a sharp increase in temperature which is followed by a decrease in temperature. This sudden increase in temperature is explained by comparing the thermodynamic properties of two volumes of steam which are at the same temperature but at different pressures. The volume of steam at the higher pressure has a higher enthalpy than the volume of steam at the lower pressure. If the pressure of the high pressure volume should suddenly drop, the steam temperature will increase until the "excess" enthalpy has been lost. A step increase in load causes a sudden decrease in steam pressure and as a result, a short increase in steam temperature.

Following this short, sharp increase in temperature, the steam temperature starts to decrease. This decrease is caused by the increased load which increases the mass flow rate of steam through the superheater. Less heat is now available to the steam flowing through the superheater per mass of steam., thus reducing the temperature of the superheated steam leaving the superheater. If no corrective action were taken, the reservoir of steam in the drum would be gradually depleted and steam pressure and temperature would decrease indefinitely. Under controlled conditions, the decrease is halted by increasing the fuel flow rate and thus increasing the steam generation rate and the heat available to the superheated steam.

This sequence of events can be seen clearly in the plot of PI controlled temperature. The first and narrow spike corresponds to the nonminimum phase effect. The second and wider spike corresponds to the affect of increased fuel flow. Essentially, the first spike is generated by fast processes occurring with the superheater whereas the second spike is cause by the effect of slower drum processes. The two spikes cannot be clearly seen in the plot of temperature for either the nonlinear or fuzzified linear predictive controllers. In both cases, the effects of a change in load have been pre-empted to a great extent by controller action. In both cases, a sudden increase in attemperation may be seen which limits the non-minimum phase effect of a step load increase. This is followed by a slower change in attemperation which limits the effect of an increased fuel flow on temperature.

The performance indices of the PI, fuzzified linear and nonlinear main steam temperature controllers at the 10%, 30%, 50%, 70% and 90% operating point are shown in Fig. 10.5

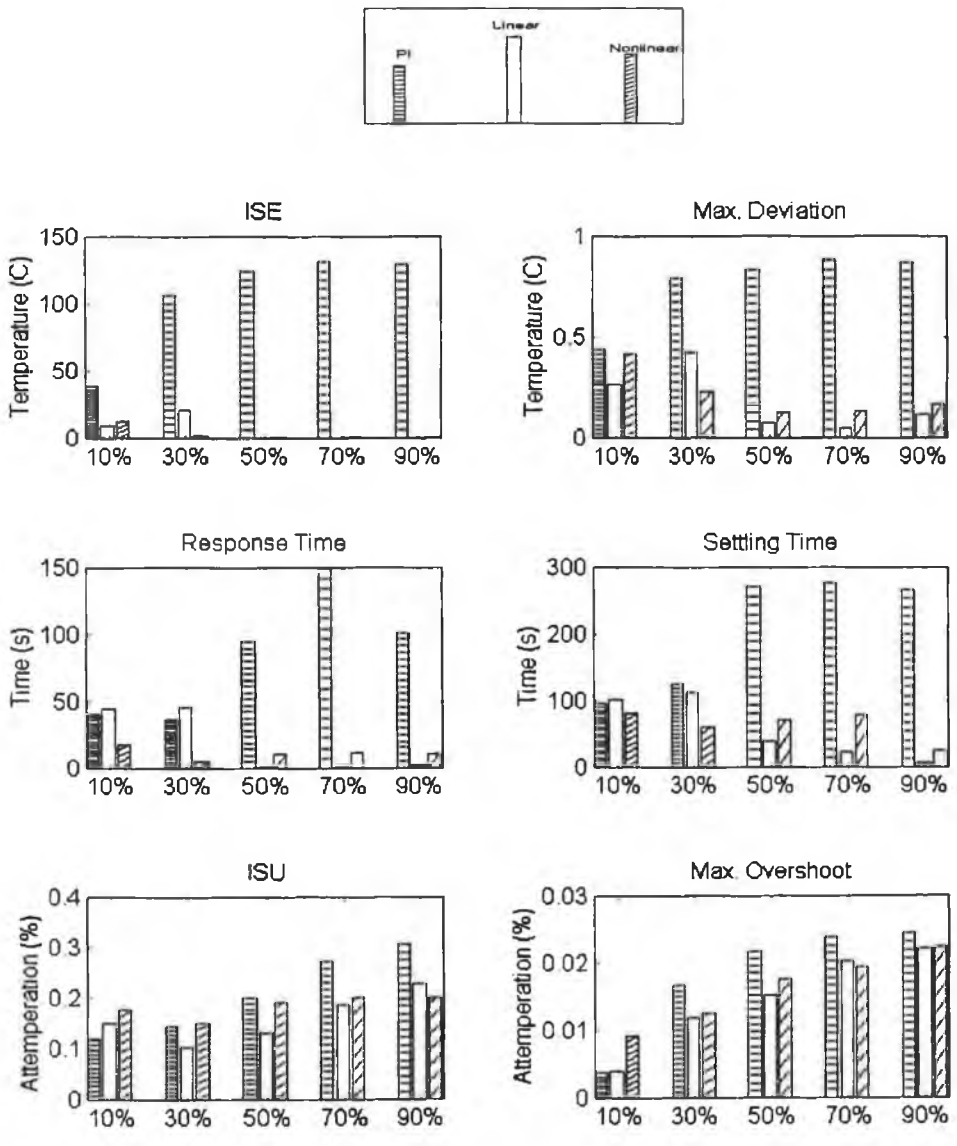


Fig 10.5 Performance Indices of Main Steam Temperature at 10%, 30%, 50%, 70% and 90% Operating Points

1. The most striking feature of Fig. 10.5 is the very superior performance of both the fuzzified linear and nonlinear predictive controllers at all operating points However, this improved controller performance is not accompanied by a large increase in the *ISU* or maximum overshoot index. In effect, the timing of the applied control action is far more significant in this instance than the magnitude of the control action. This significant improvement in performance can be attributed to the use of an internal controller model. The exact role of the internal controller model and how it achieves this improvement is described in full in the preceding paragraphs.

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
PMISE (%)	100	6	3
PMISU (%)	100	77	89
Temperature Control Index (%)	100	41	46

Table 10.7 Indices of Controller Performance for Main Steam Temperature (in Response to a 5% Step Change in Load)

The fuzzified linear and nonlinear predictive control shown an *MISE* index of just 6% and 3% respectively of the *MISE* error for the PI control strategy. As stated earlier, this reduction in tracking error does not require extra control action. The *MISU* index of the fuzzified linear controller and nonlinear controller is in fact smaller than the *MISU* index of the PI control strategy.

According to this measure of controller performance the fuzzified linear controller can improve conventional PI control of main steam temperature control by 59% and nonlinear predictive control can improve it by 54%.

Drum Water Level

The response of drum water level and feedwater flow rate to a 5% step load disturbance is shown in Fig. 10.6a and Fig. 10.6b respectively.

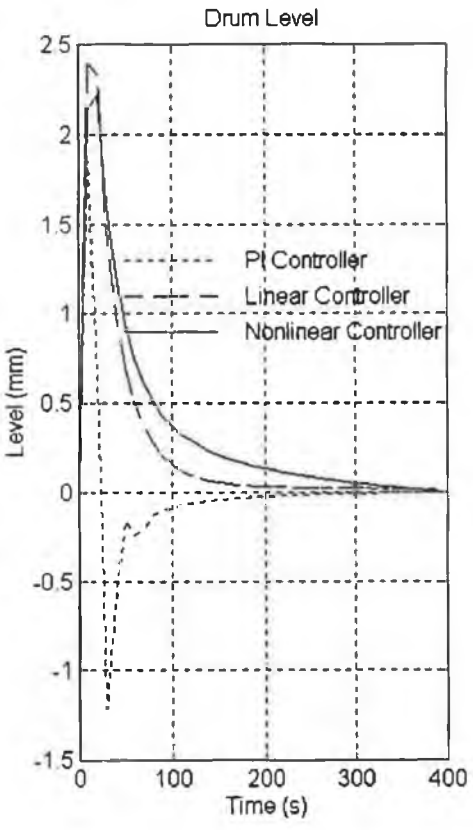


Fig 10.6a Response of Drum Level to a Step Change in Load

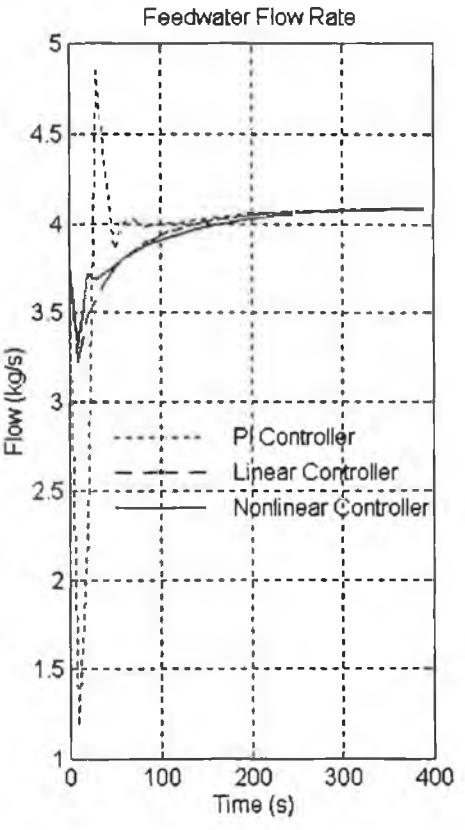


Fig 10.6b Feedwater Flow Rate after a Step Change in Load

The function of an internal controller model is again very evident in the control of drum water level. A step increase in load causes a nonminimum phase effect in drum water level i.e. a short-term increase in drum water level due to transient decrease in fluid density, followed by a long-term decrease in drum water level due to the increased steaming rate. The predictive controllers first predict the transient increase in drum water level and compensate by quickly decreasing feedwater flow. This prevents the drum level from rising as high as for the PI controlled system. Likewise, both of the predictive controllers then predict that drum level will start to fall and proceed to increase feedwater flow before the drum water level reaches the setpoint. This is very important as it ensure sufficient feedwater is available to match the increased steaming rate.

The PI controller continues to decrease feedwater flow until the drum water level reaches the setpoint with the consequence that a large undershoot of drum water level occurs. Feedwater flow must again be increased to bring drum level to its setpoint and to match the increased rate of evaporation from the drum. If the feedwater pump cannot increase the feedwater flow rate

sufficiently quickly, there exists a danger that the drum will "dry-out". To prevent the possibility of dry-out, steam flow can be added to the PI controller output as a feedforward signal. This ensures that the steam flow leaving the drum is matched by the feedwater flow entering the drum. The effect of using steam flow as a feedwater flow signal is illustrated in Fig. 10.7. The PI drum water level controller is de-tuned to ensure that it does not cancel the effect of the feedforward flow signal. The proportional gain, P, is set to 1e-1 and the integrator gain is set to 0.

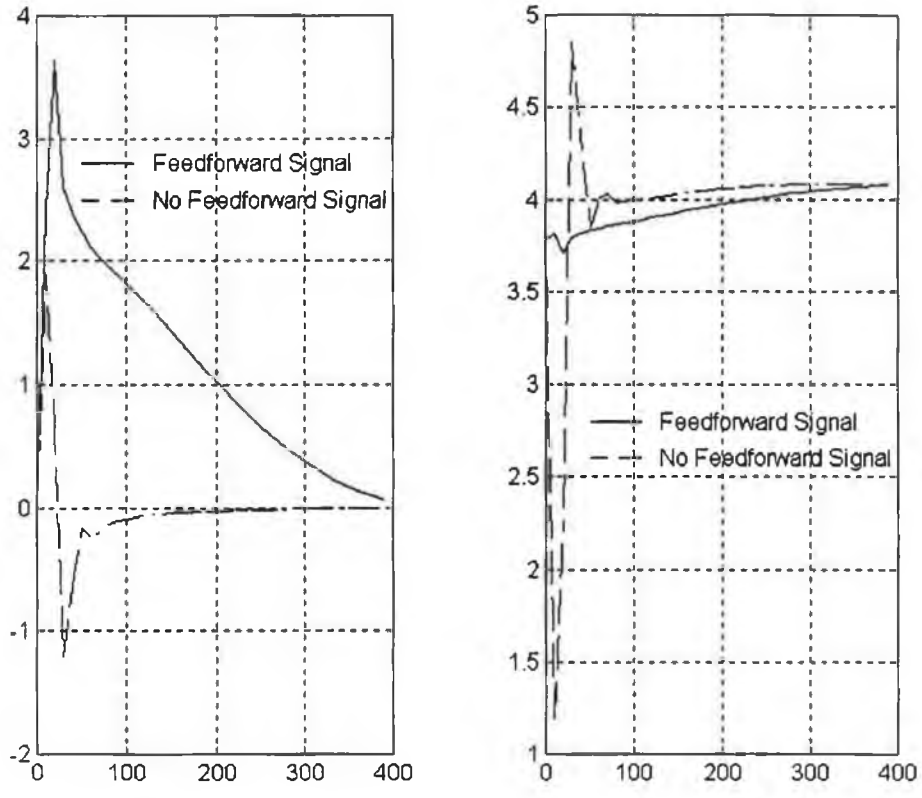


Fig 10.7 Response of Drum Level to a 5% Step Change in Load Using Steam Flow as a Feedforward Signal

In this case feedforward signal decreases controller action, but causes the transient increase in drum water level to increase. The results presented below are based on the original drum level controller (i.e. no feedforward flow signal) on the basis that the controller action does not cause feedwater pump limitations to be exceeded and yields better drum level control.

The performance indices of the three controller at the 10%, 30%, 50%, 70% and 90% operating

points are shown in Fig. 10.8, using the following legend:

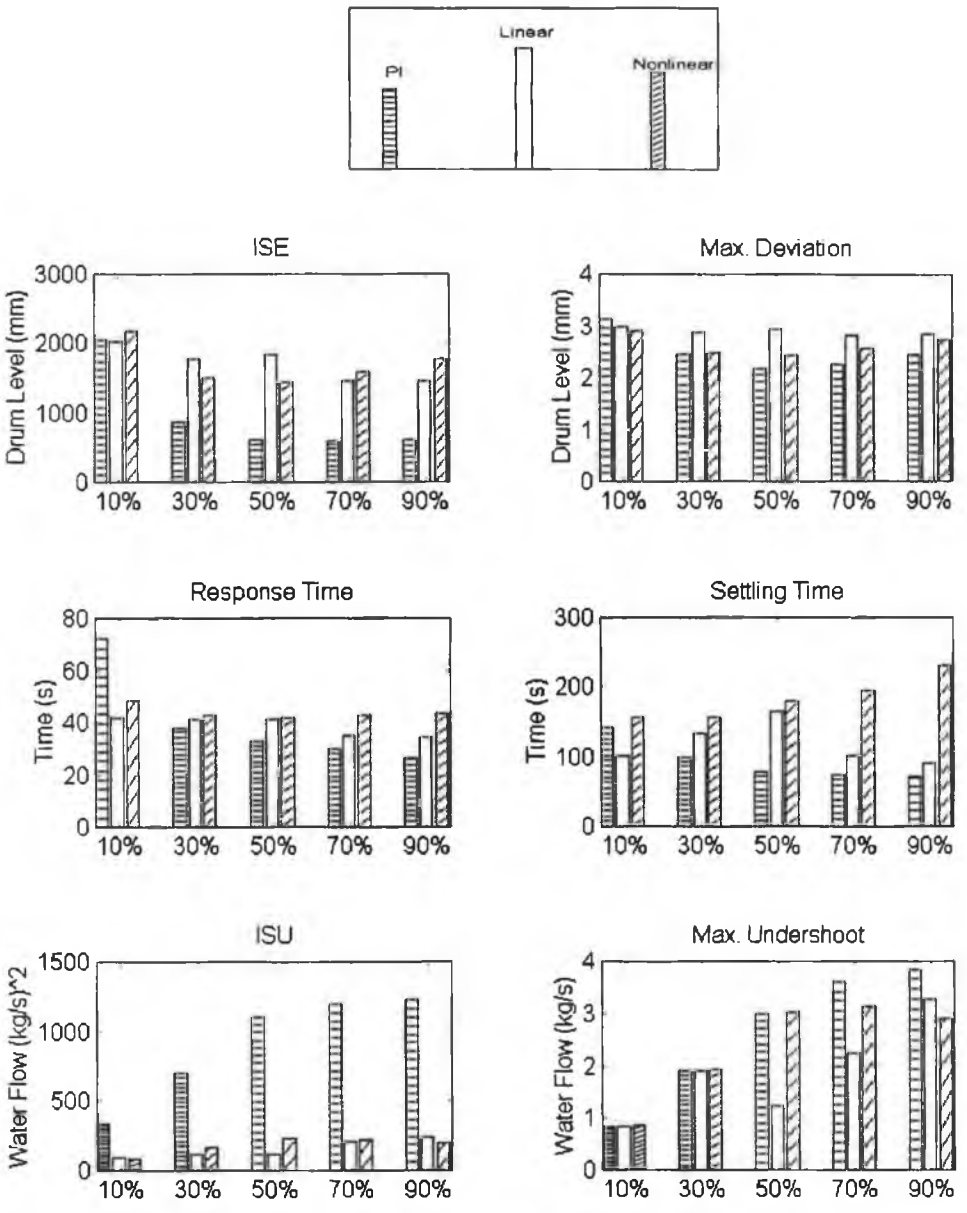


Fig 10.8 Response of Drum Level to a 5% Step Change in Load at 10%, 30%, 50%, 70% and 90% Operating Points

1. The PI controller applies a considerably more energetic control signal than either of the predictive controllers (approximately 500% greater, as measured by the *ISU* index) at all the operating points and in consequence achieves a smaller *ISU* signal at the 30%, 50%, 70% and 90% operating points.
2. At the 10% operating point, increased control activity has not paid off in terms of system performance for the PI controller. This can be understood by noting that the maximum undershoot of feedwater flow is identical for all three controller. This is the maximum

possible undershoot without reducing feedwater flow below zero. As a result all three controllers were limited in terms of controller activity.

The overall performance of each of the three controllers in terms of the *PMISE, PMISE,* and the mean of these two indices (*Drum Level Control Index*) is presented in Table 10.8.

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
PMISE (%)	55	100	99
PMISU (%)	100	17	20
Drum Level Control Index	91	100	85

Table 10.8 Indices of Controller Performance for Drum Level (in Response to a 5% Step Change in Load)

According to this measure, the performance of the PI, the fuzzified linear and the nonlinear predictive controllers are broadly comparable. The additional controller activity of the PI controller has been balanced by the resulting reduction in tracking error.

Percentage O_2 in the Stack Gases

Fig. 10.9 and Fig. 10.9b shows the response of percentage O_2 and air flow rate respectively to a 5% step load disturbance at the 70% operating point.

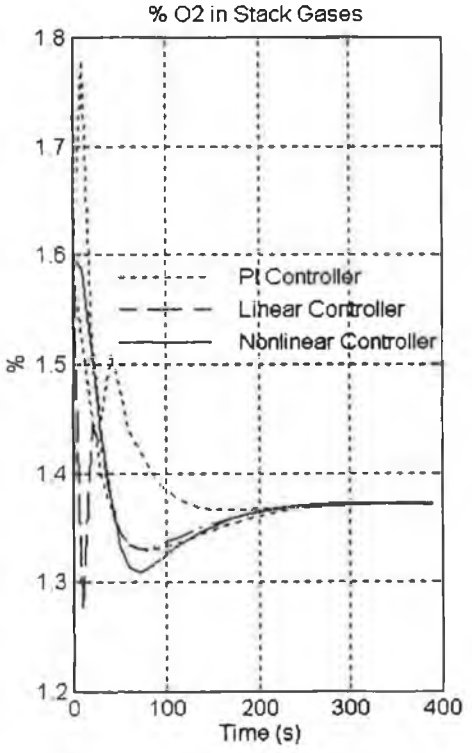


Fig 10.9a Response of $\%O_2$ to a Step Change in Load

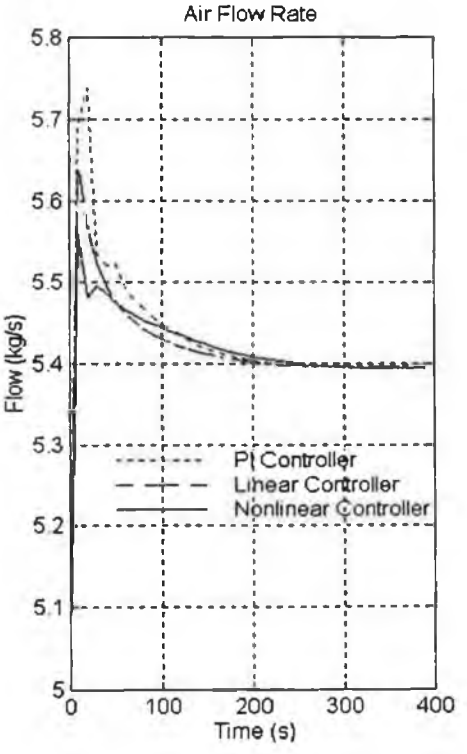


Fig 10.9b Air Flow Rate after a Step Change in Load

The percentage O_2 controllers are required to respond to both a change in load and a change of set-point. Again a smaller maximum deviation from the setpoint occurs using the predictive controllers than using the PI controller.

The performance indices of the percentage O_2 controllers at the 10%, 30%, 50%, 70% and 90% operating points are shown in Fig. 10.10, using the following legend:

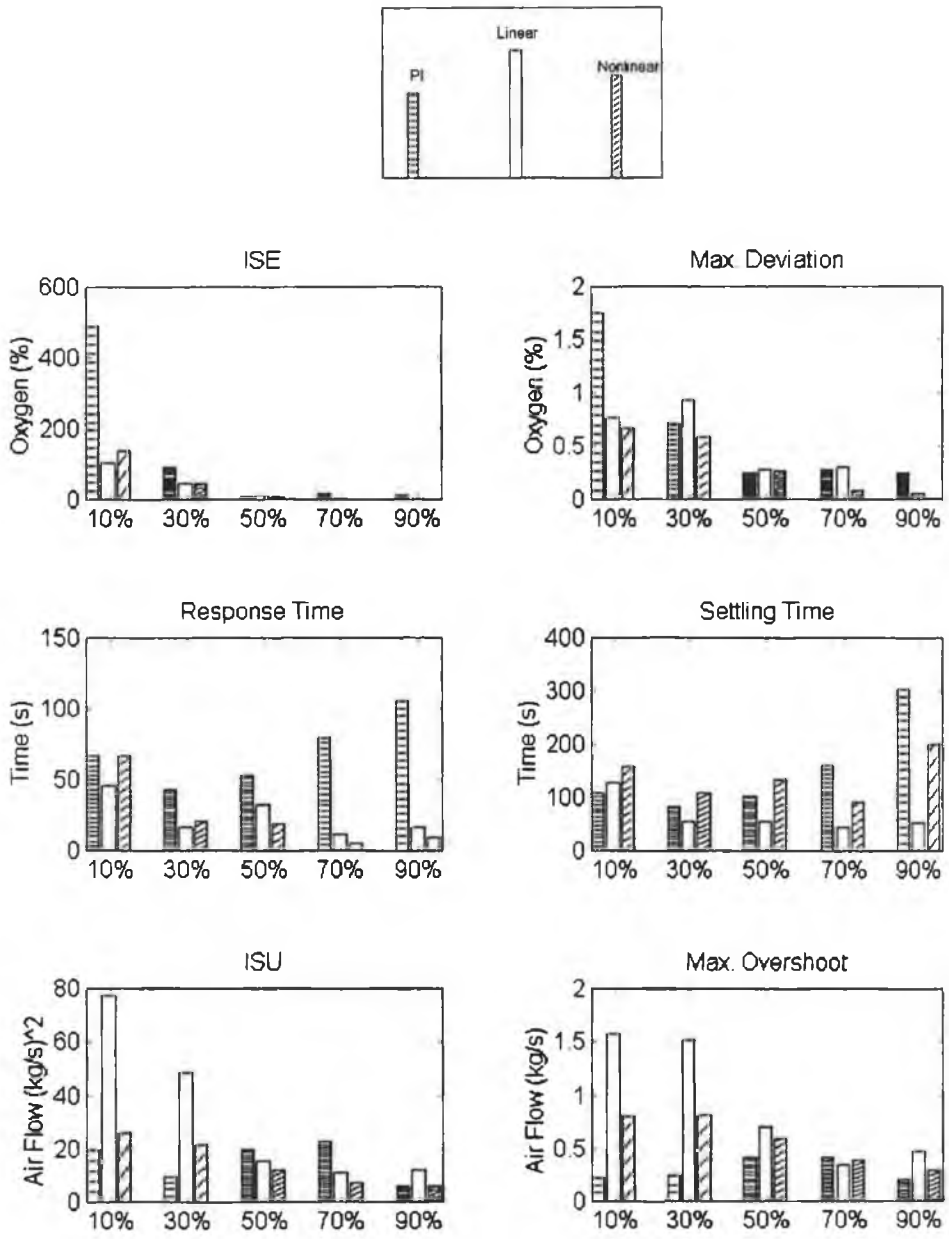


Fig 10.10 Performance Indices of Percentage O_2 following a 5% Step Change in Load at 10%, 30%, 50%, 70% and 90% Operating Points

1. The performance of the PI controlled system is extremely poor at the 10% operating point. At the 10% operating point the maximum deviation from the setpoint using the PI controller is approximately twice as big as the maximum deviation for of either the fuzzified linear or

nonlinear predictive controller. This is not surprising as the time delay of percentage O_2 is longest at the 10% operating point and the PI control strategy has no specific mechanism for controlling time delay systems. A reduction in the maximum deviation of the percentage O_2 from its setpoint can have important economic benefits. This issue is described in full in Section 8.2

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
PMISE (%)	100	27	31
PMISU (%)	47	100	44
Percentage O_2 Control (%) Index	100	86	52

Table 10.9 Indices of Controller Performance for Percentage O_2 (in Response to a 5% Step Change in Load)

The indices confirm that a predictive control strategy can reduce tracking error considerably. The *PMISE* indices of the fuzzified linear controller and nonlinear predictive controller are about 70% smaller than the *PMISE* index of the PI controller. The *PMISE* index of the fuzzified linear controller is 4% smaller than that of the nonlinear predictive controller, but it is interesting to that the *PMISU* index of the fuzzified linear controller is 66% greater than the *PMISU* index of the PI control strategy. It can be seen from Fig. 10.10 that the extra controller action is applied by the fuzzified linear controller at the 10% and 30% operating points. The reason for this extra controller effort is explained in Section 8.5.3.

10.3.2 Set-Point Variations

It is described in Section 10.2.1 how boiler efficiency can be improved by relating the setpoint of the percentage of O_2 in the stack gases to the boiler operating point. This feature has been implemented for each of the three controllers and verified by applying a step load disturbance. A change in load results in a change in fuel flow which in turn results in a change in the setpoint of the percentage of O_2. in the stack gases. The percentage O_2 controller must simultaneously react to the disturbance and to a change in setpoint. The performance of the three percentage O_2 controllers under these conditions has been presented in Section 10.3.2.

Boiler efficiency may also be improved by relating the main steam pressure setpoint to load. When the load demand is low, the steam from the boiler is reduced by partially closing a single throttling valve or by fully closing some of the valves in a bank of valves for example. This has the undesirable affect of increasing the pressure loss associated with these fittings. To counteract this the pressure must be held at a higher pressure than necessary. This in turn requires that the

boiler feed pumps consume more power than necessary, thus reducing the overall boiler efficiency. If the main steam pressure setpoint is reduced at low loads, the cross sectional area of the throttling valve can be increased and the pressure loss across the throttling valve is reduced. This in turn reduces the power which must be consumed by the feed pumps merely to overcome the pressure head across the throttling valve. This type of operation is referred to as "sliding pressure" mode.

A step change in the setpoint of main steam pressure is applied to the plant at five different operating points - 10%, 30%, 50%, 70% and 90%. The response of main steam pressure is shown together with that of fuel flow (as defined for the PI control strategy). for the test at the 70% operating point only. The response at the other operating points is shown by plotting the performance indices obtained at the 10%, 30%, 50%, 70% and 90% operating points.

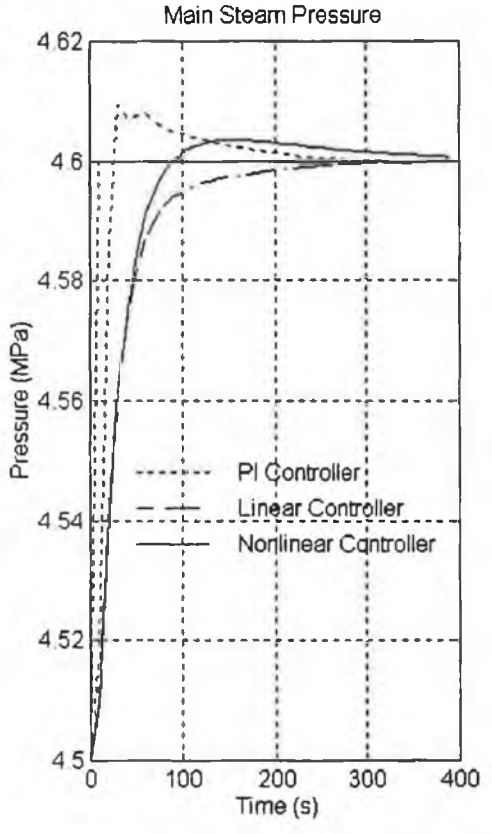


Fig 10.11a Response of Main Steam Pressure to a Step Change in Setpoint

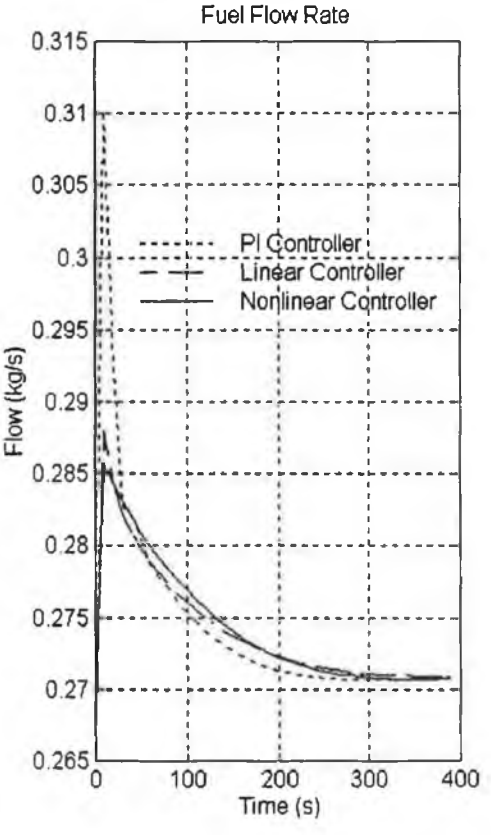


Fig 10.11b Fuel Flow Rate after a Step Change in Setpoint

The step change in main steam pressure setpoint is successfully tracked by each of the three controllers. The PI controller produces the fastest response, but at a significant cost in terms of controller action. The controller action of the fuzzified linear and nonlinear predictive controller is much more economical and in addition is The nonlinear controller also causes main steam

pressure to overshoot the setpoint but to a lesser extent. This overshoot is caused by inaccuracy of the internal controller model and is gradually reduced by integral action. There is no overshoot in the main steam pressure signal for the fuzzified linear controller. This is the best response in terms of conservation of energy.

Fig 10.3 presents the performance indices measured for each of the three controllers for the setpoint change: The four plots on the top two rows are performance indices of system response. The two plots on the third row are indices of controller action. The following legend is used:

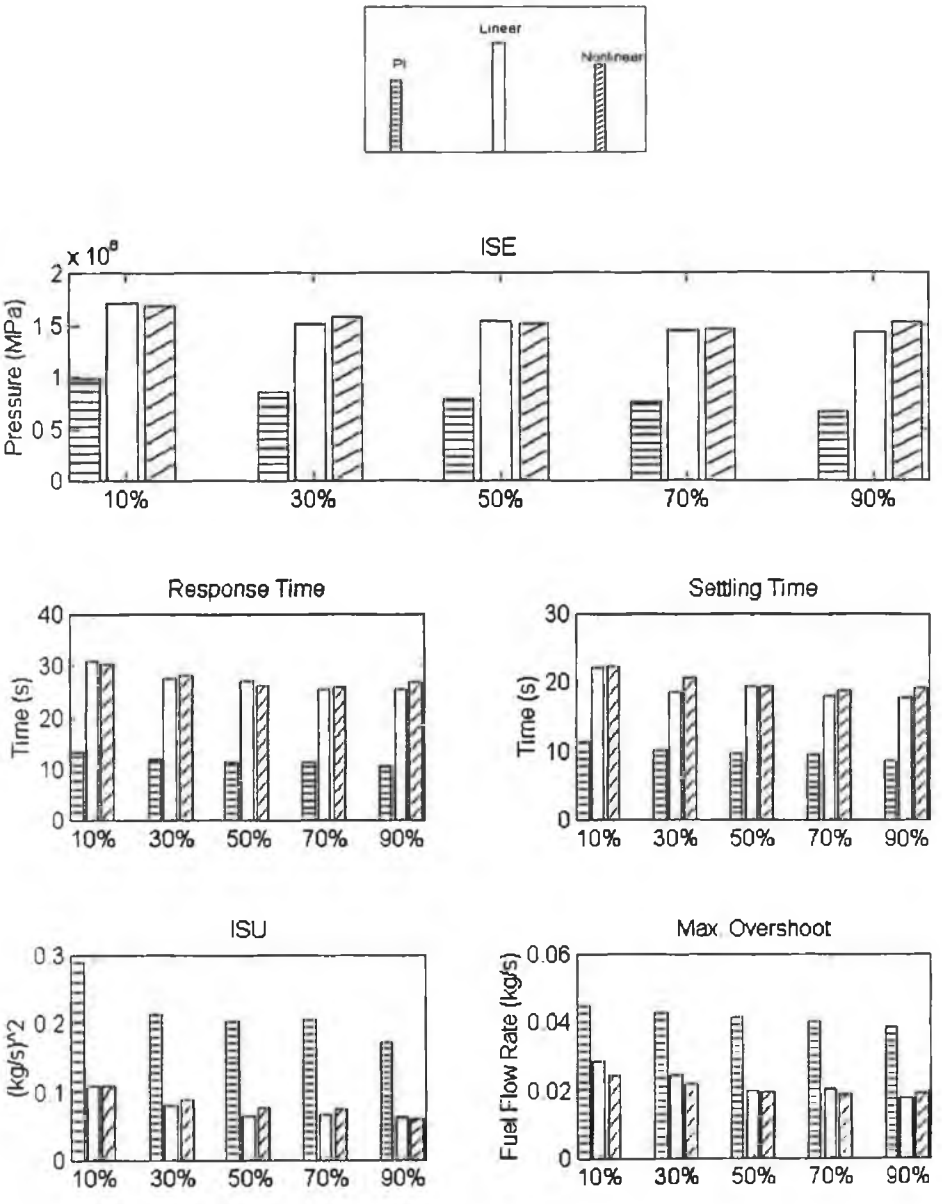


Fig 10.12. Performance Indices of Main Steam Pressure following a Setpoint Change at 10%, 30%, 50%. 70% and 90% Operating Points

1. The performance indices follow a similar trend at all five operating points. The PI control

strategy achieves the smallest *ISE*, response time and settling time at all five operating points. This is not surprising given that the PI controller provides a more active control signal, as measured by the *ISU* and maximum overshoot index.

2. The response of the fuzzified linear and nonlinear controllers are very similar at all operating points. These two controllers employ similar control signals and achieve a comparable performance in terms of *ISE*, response time and settling time.

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
PMISE (%)	52	98	100
PMISU (%)	100	35	37
Pressure Control Index (for Setpoint Variation) (%)	100	88	91

Table 10.10 Indices of Controller Performance for Main Steam Pressure (in Response to a 0.1 MPa Step Change in Main Steam Pressure Setpoint)

By this overall measure of controller performance, the fuzzified linear and nonlinear predictive control can improve upon PI control of main steam pressure following a setpoint change by 12% and 9% respectively. In the case of a step change in load, however, the fuzzified linear controller and nonlinear predictive controller can improve upon PI control of main steam pressure by 26%. The apparent discrepancy between PI control performance for a setpoint test and a load test can be attributed to the relatively simplicity of the setpoint test - load is constant so the benefits of a model-based control methodology are not as significant. The primary advantage of predictive controller for a setpoint test is that it forces main steam pressure to follow an approximately exponential reference trajectory This reduces overshoot and conserves fuel. Overshoot can be completely eliminated if the internal controller model represents the plant exactly.

10.3.2 Summary of Controller Performance

The overall performance indices calculated for the load response and set-point response tests are presented again in Table 10.11. A global performance index is calculated for each controller by calculating the mean of these indices and then normalising with respect to the maximum of the calculated mean indices.

Index	PI Controller	Fuzzified Linear Controller	Nonlinear Controller
Pressure Control Index(%)	100	74	77
Temperature Control Index(%)	100	41	46
Drum Level Control Index(%)	91	100	85
Percentage O_2 Control Index(%)	100	85	52
Pressure Control Index (for Setpoint Variation) (%)	100	88	91
Normalised Overall Index(%)	**100**	**76**	**70**

Table 10.11 Indices of Controller Performance for Main Steam Pressure (in Response to a 0.1 MPa Step Change in Main Steam Pressure Setpoint)

According to this overall index, the nonlinear predictive controller can improve boiler control performance by 30% in comparison to conventional PI control and by 6% in comparison to the fuzzified linear controller. This index gives a measure of performance of each controller in terms of controller action and tracking error, relative to the performance of the other controllers.

10.4 Effect of Controller Models on Controller Performance

It can be seen from Table 10.11 that the fuzzified linear and nonlinear predictive controller achieve comparable performance in terms of *ISU* and *ISE*. It can also be seen from Fig 10.2a to 10.12 that performance of both controllers is also comparable with respect to maximum deviation from the setpoint and plant response time. Despite this the performance of the nonlinear controller is generally inferior to that of the fuzzified linear predictive controller with respect to plant settling time. The mean settling time for both controllers (measured for setpoint and load step change tests over five different operating points) are presented in Table 10.12.

	Fuzzified Linear Controller	Nonlinear Predictive Controller
Mean Settling Time	85	138

Table 10.12 Mean Settling Time of Fuzzified Linear and Nonlinear Predictive Controller

It can be seen that the mean settling time of the nonlinear predictive controller is 62% larger than that of the nonlinear predictive controller.

One possible reason for the difference in settling time index is that the internal controller model of the fuzzified linear controller is more accurate. It is not possible to prove or disprove this for all cases, but simulations results presented in Chapter 5 and Chapter 6 demonstrated that both the linearised boiler models and the neural network models can represent the first-principles model very accurately. The accuracy of the neural network model is demonstrated by applying a 1% step change in fuel flow to the first principles models at 90%, the internal controller model of the fuzzified linear controller and the internal controller model of the nonlinear controller. A 16-step ahead prediction of main steam pressure is calculated by both controllers and compared to the first-principles model output - which has been advanced by 16 steps. The predicted and advanced process outputs are plotted in Fig. 10.13.

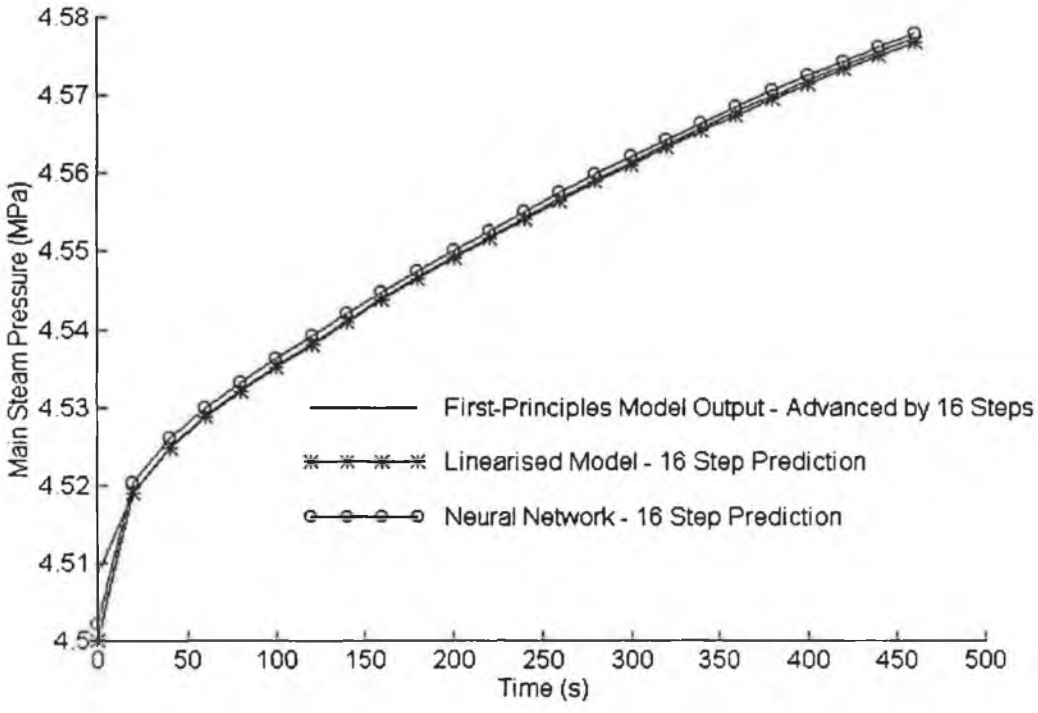


Fig. 10.13 Comparison of Accuracy of Internal Controller Model of Fuzzified Linear and Nonlinear Predictive Controller

The Integral Absolute Error (*IAE*) and Integral Squared Error (*ISE*) of the neural network model and the fuzzified model output are presented in Table 10.13.

Controller	*IAE*	*ISE*
Linearised Model	1.2e7	4.7e10
Neural Network Model	3.7e6	5.7e9

Table 10.13 Comparison of *IAE* and *ISE* of Linearised and Neural Network Models

It can be seen from Table 10.13 that the accuracy of the neural network model is in fact considerably better than that of the linearised model.

The second possible reason for the difference between the performance of the two controllers is a difference in the cost function definition of the controllers. The cost function of each linear controller is defined incrementally whereas the cost function of the nonlinear controller is defined in absolute terms. In the case of a linear controller the controller is required to minimise the error between the predicted *change* in model output and the desired *change* in plant output. In the case of the nonlinear controller, the controller is required to minimise the error between the predicted plant output and the desired plant output. The cost function is stated incrementally for the linear controller because it is shown in Section 8.3.3 that this yields an incremental or integral control law which eliminates steady state errors. This is a very attractive feature but it is only feasible provided that the model can achieve steady-state conditions when the plant is in steady-state. For this reason, it cannot be applied to the neural network controller, as the neural network model, is not guaranteed to attain a steady-state condition when the plant is in steady-state. This problem is illustrated in Fig 10.14

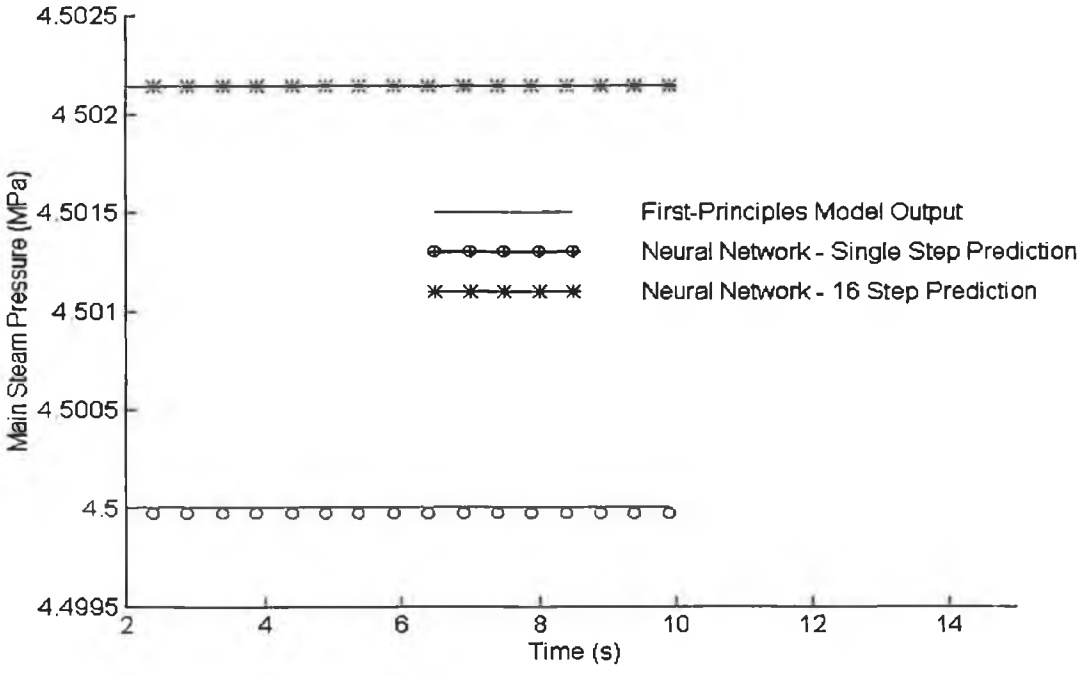


Fig. 10.14 Neural Network Single-Step and Multi-Step Prediction of First-Principles Model

It can be seen from Fig 10.14 that the single step ahead and multi-step ahead neural network predictions are constant. This is due to the fact that the inputs to the neural network model are the inputs, states and outputs of the first-principles model. If the first-principles model is in steady-state, it follows that the neural network inputs will be constant and likewise the neural network model output. It is also clear from Fig. 10.14 that the single- and multi-step ahead neural network prediction are unequal. This can be understood if it is noted that the single step-ahead prediction of the first-principles model output is not exactly equal to the first-principles

model output i.e. the neural network did not learn the conditions for steady-state exactly and is predicting that the first-principles model output will change by a small amount. The one-step ahead neural network output is then used to be update the neural network inputs in order to calculate the two-step ahead prediction. At each prediction step, the error between the prediction and actual model output increases slightly.

Weak integral action is added to the nonlinear control law to eliminate steady-state errors but this is not as effective as the inherent integral action of the fuzzified linear model. This can be seen by comparing the response time and settling time of main steam pressure following a 5% setpoint disturbance.

Criteria	Fuzzified Controller	Nonlinear Controller
Response Time	33.9	36.6
Settling Time	103	338
Prediction Error	2.19e5	4.5e3

Table 10.14 Comparison of Settling Time, Response Time and Prediction Error of Fuzzified Linear and Nonlinear Controller

It is interesting to note that the response time of the fuzzified linear controller and nonlinear controller are practically identical. The settling time of the nonlinear controller is over three times longer than that of the fuzzified linear controller however. To demonstrate that this is not attributable to greater model inaccuracy on the part of the nonlinear controller, the prediction errors of both controllers are also included. The prediction for the fuzzified controller is nearly 4.8 times greater than the prediction error for the nonlinear controller.

Clearly, the benefits of inherent integral action are most noticeable when the plant is close to the setpoint, (relatively to the accuracy of the neural network model). In practical terms, the neural network model is sufficiently to accurate to bring the plant output to the setpoint . Given that the neural network model is very accurate, the nonlinear controller is not at a serious disadvantage in terms of controller performance. This is confirmed by the fact that the global performance index of the nonlinear controller is smaller than that of the fuzzified linear controller.

10.5 Comparison of Controller Implementation Requirements

Implementation requirements address the specific hardware requirements of each controller. These include:

1. Hardware Requirements to Implement Control Law
2. Sensors and Actuators Required

The computational overhead of the PI controller cannot be evaluated on the basis that it is an analog controller. This is not really significant, however, as it can be assumed that it almost negligible. The computational overhead of calculating the solution of the nonlinear controller using both gradient search and genetic algorithm optimisation is discussed in Section 9.5. The number of MFLOP per second required to implement the nonlinear controller is presented in that section and is shown here again, for comparison with the fuzzified linear controller.

Controller	MFLOP/S
Fuzzified Linear Controller	0.017
Nonlinear Controller (Gradient Search Optimisation)	2.1
Nonlinear Controller (Genetic Algorithm Optimisation)	13.6

Table 10.15 MFLOP/s for Gradient Search Optimisation and Genetic Algorithm Optimisation

It can be seen that the computational overhead of the nonlinear controller using gradient search optimisation is over 100 times smaller than that of the fuzzified linear controller. This does not present any implementation problems however, as a standard Digital Signal Processing board (TMS320C30) can execute 33.3 MFLOP per second.

The extra computational overhead of the nonlinear controller can be attributed mainly to differences in the computational overhead of the optimisation algorithm. The nonlinear controller uses a numerical optimisation algorithm (gradient search or genetic algorithms). A numerical optimisation algorithm finds the optimal solution by iteratively calculating the cost function for varying values for controller action. The fuzzified linear controller uses least squares optimisation, an analytical optimisation method which calculates the cost function only once during each sampling period. It is also important to compare the computional effort required calculate the cost function - this is proportional to the computational effort of predicting the plant output. The number of MFLOP used to predict the plant output using the internal controller model of the fuzzified linear controller and the nonlinear controller is presented in Table 10.16. The internal controller model of the fuzzified linear controller incorporates three linearised models and is referred to here as a fuzzified linear model.

Model	Main Steam Pressure and Drum Level (MFLOP)	Main Steam Temperature (MFLOP)	Percentage O_2 in Stack Gases (MFLOP)
Fuzzified Linear Model	3.98 e-3	5.6 e-4	3.0 e-5
Neural Network	1.28 e-3	1.8 e-4	2.8 e-4

Table 10.16 MFLOP/s for One-Step Prediction Using 3 Linearised Boiler Models and Neural Network

It can be seen that the neural network model of main steam pressure and drum level uses approximately one third as many MFLOP as the corresponding fuzzified linear model, whereas the neural network model of percentage O_2 in stack gases uses approximately 10 times more MFLOP than the corresponding fuzzified linear controller. In any case, it can be seen the computation overhead of the neural network models is broadly comparable to that of the fuzzified linear model.

2. Actuator and Sensor Requirements

Each of the four SISO PI controllers is supplied with the output of a single sensor and activates a single actuator. A further device is required to provide the primary air demand signal;. In total, the full PI controlled system requires 4 sensors and 4 actuators plus a .device to generate the primary air demand signal.

The fuzzified linear predictive controller requires a sensor for each of the four controlled variables and an actuator for each of the four manipulated variables. A further three sensors are required to measure three disturbance inputs - feedwater flow temperature, number of open valves at boiler output and main steam flow.

The nonlinear predictive controller uses the same sensors and actuators as the fuzzified linear predictive controller. However, it also needs an additional set of sensors to measure the value of boiler state variables. In all 11 additional sensors are required. These additional measurements are required to initialise the neural network model at the start of each sampling period.

The hardware requirements for each of the three controllers are summarised in Table 10.17.

Hardware Requirements	**PI Control Strategy**	**Fuzzified Linear Predictive Control Strategy**	**Nonlinear Predictive Control Strategy**
Computational	Analog or PLC Circuitry	DSP Board	DSP Board
Actuators	4 (plus device to generate primary air demand signal)	4	4
Sensors	4	7	18

Table 10.17 Comparison of Hardware Requirements

10.6 Conclusions

The three boiler control strategies have been compared on the basis of controller performance and implementation requirements.

The comparison demonstrates that nonlinear predictive control can achieve significantly improved boiler control performance than conventional PI boiler control. Two types of nonlinear predictive control strategy are assessed. The first strategy is based upon a set of linear controller which are interpolated using fuzzy logic. According to a defined index of controller performance, which considers controller action and tracking error, this fuzzified linear controller can improve boiler control by 24% in comparison to a PI control strategy. The second nonlinear predictive control strategy is based upon dedicated nonlinear plant models. According to the defined index of controller performance, this strategy can improve boiler control by 31% in comparison to a PI control strategy.

In particular, the comparison highlights some attractive features of the nonlinear predictive and fuzzified linear controllers:

- The maximum deviation of the controlled variable from the setpoint is reduced by prompt control action following a load disturbance. This is illustrated very clearly by the main steam temperature response. The control signal of the PI controller is not more active than that of the fuzzified linear or nonlinear predictive controller. The mean ISU of the PI controller (measured over 5 operating points, following a 5% increase in load) is in fact twice than of the nonlinear predictive controller. However, the predictive controllers include internal disturbance models and can react more quickly to a disturbance than the PI controller. As a result, the mean maximum deviation of steam temperature from the setpoint is reduced from a value of 0.8 °C for the PI control strategy to approximately 0.2 °C for both of the predictive controllers.
- The important of an internal controller model is also demonstrated in the case of drum water level control following a load disturbance. The fuzzified linear and nonlinear controllers can predict that drum water level is subject to a nonminimum phase effect and react accordingly. The PI controller can not predict that drum water level is subject to a nonminimum phase effect and as a result allows drum level to fall below its setpoint. This could result in drum-*'dry-out'*. The risk of *'dry-out'* can be decreased by using steam flow as a feedforward control signal for drum level control, but this increases the drum level tracking error significantly.

- The importance of taking plant time delay into account in the controller design has been confirmed. The percentage O_2 measurement delay increases as the operating point of the plant decreases. At the 10% operating point the measurement delay is 12s. At this operating point the nonlinear and fuzzified linear predictive controller can reduce the maximum deviation of percentage O_2 from the setpoint from a value of 1.7% to approximately 0.7%.

There is little difference between the performance of the fuzzified linear predictive controller and the nonlinear predictive controller. The two controllers achieved comparable performance indices in all situations with the exception of the performance index for settling time. The mean settling time of the nonlinear controller is 62% longer than the mean settling time of the fuzzified linear controller. The fuzzified linear controller can eliminate steady-state errors more effectively than the nonlinear controller because the control law has inherent integral action. In practice, this is not a significant problem in terms of controller performance as it only indicates small deviations from the setpoint.

11. Conclusions

The capacity of nonlinear predictive control to deal systematically with all of the boiler controller issues *and* to yield superior control to industry-standard PI control has been demonstrated in this work. Using a general measure of performance, which considers both tracking error and controller action for four controlled variables, a nonlinear predictive control strategy based upon a nonlinear plant model improves boiler control performance by 30% in comparison to a boiler PI control strategy. A nonlinear controller strategy which uses fuzzy logic interpolation to combine the outputs of three linear controllers improves boiler control performance by 24%. The performance index is measured for four controlled variables, subject to both load disturbances and setpoint variations at five different operating points.

The significance of this improvement in terms of overall plant efficiency is illustrated by considering some of the important areas where improvements are achieved:

- Both the nonlinear and fuzzified linear predictive controllers apply just sufficient control action to move the controlled variables to the setpoint along an approximately exponential trajectory. This practically eliminates setpoint overshoot and consequently excess controller action. This has obvious benefits for the control of a variable such as main steam pressure - setpoint overshoot increases fuel costs and yields no benefits. Setpoint overshoot can also be eliminated in a PI control strategy by decreasing the P and I gains, but only at the cost of a greatly increased maximum deviation from the setpoint following a load disturbance.
- Average maximum deviation of main steam temperature from setpoint after a 5% step change in load is reduced from 0.8 °C for the PI control strategy to approximately 0.2 °C for both predictive control strategies. Reducing steam temperature variations increases plant life-time by reducing thermal stresses on the superheated metal tubes.
- Average maximum deviation of percentage O_2 in the stack gases from the setpoint after a 5% step change in load is reduced from 1.7% for PI strategy to 0.7% for fuzzified linear and nonlinear predictive strategy. This implies that the percentage O_2 setpoint can be reduced without increasing the risk of sub-stoichiometric combustion. A reduction in the percentage O_2 setpoint causes the excess air in the combustion gases to be reduced with a direct impact on combustion efficiency.

The nonlinear predictive controller, based upon a dedicated nonlinear model, achieves a marginally better performance than the fuzzified linear controller according to the defined performance index. However, this does not necessarily imply that the nonlinear controller is a "better" controller. Controller performance must be weighed against other design issues, such as

development and implementation effort. Each aspect of the fuzzified linear and nonlinear predictive controller design is considered here - the motivation for choosing a particular design approach, advantages and disadvantages of that approach, and with the benefit of hindsight, whether it was the best choice for the problem.

Controller Methodology

A predictive control methodology was chosen for the following reasons:

- Predictive control strategy is a model-based strategy. All knowledge about the plant is incorporated directly into the controller design This is a far more systematic approach than conventional PI control. For example, in conventional PI controller, fuel flow must explicitly be provided to the percentage O_2 controller as a feedforward flow signal. In effect, controller performance is dependent on human *a-priori* knowledge that fuel flow is a disturbance input which effects percentage O_2 by a given amount. In a model-based controller, the effect of each plant input on the controlled variables is provided to the controller *via* the model so that appropriate feedforward control action can be taken automatically.
- Controller objectives are defined *via* a cost function definition. Effectively, a cost function translates controller objectives and modelled information about plant behaviour into a mathematical optimisation method.
- Plant constraints can be explicitly specified by the plant engineer in the controller cost function. If this is not possible, the controller must be tuned in a way which ensures that the plant constraints are not violated. This is not a rigorous approach, its success depending largely on the skill of the plant engineers.
- Predictive controller can be based upon any type of model, including nonlinear and multivariable models.
- It can be adapted without difficulty to control time-delay systems.

Predictive control proved to an excellent choice of control methodology for the boiler process. A predictive controller can be developed quickly - discounting model development which accounted for 90% of the design effort. The developed controllers are also easy to tune since the response time of the controlled variables could be specified to a high degree of accuracy. For example, specifying a desired 63% response time of 24.68s for main steam pressure yielded an actual response time of 25.28s in the case of the fuzzified linear controller and 26.98s in the case of the nonlinear controller.

A nonlinear approach is used because of the nonlinearity of the boiler dynamics. Initially, the nonlinear approach focused on the approach which yielded maximum flexibility - i.e. a nonlinear

controller, based upon a neural network model of the plant and optimised using a numerical optimisation algorithm, either gradient search or the global, genetic algorithm optimisation technique. The fuzzified linear controller was developed for comparative purposes - to determine whether comparable controller performance could be achieved by a simpler and less computationally intensive, but still nonlinear approach.

Control Configuration

It was intended to use a full multivariable approach. but this was not practicable because of the stiffness of the boiler plant. A sampling period of 1s is suitable for the control of both main steam pressure and drum water level and of 0.1s for the control of main steam temperature and percentage O_2 in the stack gases. To overcome this problem, a hybrid single-loop, multi-loop controller structure is developed:

- Main steam pressure and drum level which show a strong level of interaction are controlled by a multi-loop controller using feedwater flow rate and fuel flow rate as the manipulated variables. All other inputs to the boiler, including attemperation and air flow, are provided to this controller as disturbance inputs.
- Percentage O_2 in the stack gases is affected by the air flow rate and fuel flow rate only. It is controlled by a single-loop controller using air flow as the manipulated input. Fuel flow rate is provided as a disturbance input.
- Main steam temperature is controlled in the superheater by a single-loop controller using attemperation as the manipulated variable. The controller is based upon a model of the superheater only. It is provided with all the inputs the first-principles superheater model, including drum outputs.

This approach addresses the problem of boiler stiffness but still retain most of the benefits of a full multivariable design by providing each controller with all the relevant disturbance inputs.

Internal Controller Model

It was decided to base the nonlinear predictive controller upon neural network models for a number of reasons. Neural network models have the potential to model any nonlinear, multivariable plant. They are generated from data so development effort should minimal and they are computationally efficient.

Simulation results confirm that the neural network modelling can yield accurate nonlinear models of a boiler plant. For example, over a 4 step prediction horizon, a neural network can typically predict main steam temperature to within 0.001 °C. It has also been confirmed that the computational overhead of simulating a neural networks is very similar to that for a linearised

boiler model. The neural network model of main steam pressure and drum water level executes 1.28 MFLOP for each one-step-ahead prediction. A linearised model executes 1.33 MFLOP for a one-step-ahead prediction.

Development of a "stand-alone" superheater model for the main steam temperature controller presented serious difficulties. Many of the disturbance inputs of the superheater model are in fact drum outputs. These variables are highly correlated both with each other and with other disturbance inputs such as boiler load and fuel flow rate. The neural network model was unable to learn the true relationship between the model input and output variables because of the degree of correlation between these variables. A short-term solution was to effectively "detach" the first-principles model of the superheater from the first-principles model of the drum and generate data by applying square wave type input signals for each of the superheater input variables. This is a feasible solution if a good first-principles model of the boiler is available but, obviously, cannot be used to generate data from a real boiler plant. Alternatively, it may be necessary to model the full boiler and superheater process in order to predict main steam temperature. This is more logical approach given that the source of the problem here is trying to generate a "stand-alone" superheater model using data which is not "stand-alone".

A fundamental problem of using neural network models for control is that the neural network models cannot be guaranteed to reach a steady-state condition, even if the model is provided with steady-state inputs. Neural networks can learn the conditions for steady state plant output but only to a certain degree of accuracy. In other words, even if the plant is in steady state, the neural network model is likely to predict that it is changing slowly. This has implications for the control law. It is not possible to build implicit integral action into the control law by stating the cost function incrementally. Weak integral action has been added to the output of the nonlinear predictive controller to eliminate steady-state errors due to prediction error but this is not as effective as the inherent integral action of the fuzzified linear controller. This problems does not arise for linear or first-principles nonlinear model as in both cases, there is guaranteed to exist combinations of inputs and states which yield a steady-state model output.

The fuzzified linear predictive control is based upon linearised models of the first-principles model. Consequently, this particular controller requires the availability of a first-principles boiler model, however, the method could also be implemented using a data-based linear model. Linearised models are limited in that each linearised model can represent the plant at one operating point only. However, fuzzy logic interpolation is used effectively to develop a nonlinear controller based upon several linear controllers. The success of this approach is confirmed by comparing the performance of the fuzzified linear control strategy with the

performance of the nonlinear predictive control strategy. Using an index of controller performance which measures the performance of each controller over the entire operating range of the plant, the fuzzified linear controller is rated within 6% of the nonlinear predictive controller.

A first-principles model was developed to simulate the plant behaviour for the purposes of assessing the proposed control strategies. The fundamental advantage of the first-principles model is that it preserves the physical laws of conservation of mass and energy. The output of a the first-principles model is not highly accurate but is guaranteed to be make realistic predictions of plant behaviour under all conditions. It can also be used to provide an understanding of the boiler processes and can be customised to model various drum-type boilers. The disadvantages of this method are that it requires a very large development effort and is not suitable for use as an internal controller model because of its computational overhead. As the first-principles model is a continuous model it is not possible to make a direct comparison of the computational overhead of first-principles model with the computational overhead of the neural network and fuzzified linear models. However, 2.45e-3 MFLOP are used to execute a single integration step for the first-principles model. This is equivalent to the number of MFLOP required to simulate the neural network model main steam pressure and drum level for one second. To simulate the first-principles model for one second it may be necessary to run over a hundred integration steps.

Choice of Cost Function

A very simple cost function is found to be sufficient to ensure good controller performance for both the fuzzified linear and nonlinear controller.

The cost function has two parameters - time constant of the desired reference trajectory and the position of a single coincidence point on the reference trajectory. In addition, the cost function assumes that controller action is a step function. Despite the simplicity of the cost function, it is found that plant response can be specified very accurately using the time constant of the desired reference trajectory.

Controller Optimisation Algorithm

The nonlinear predictive controller is optimised using either gradient search optimisation of genetic algorithm optimisation. Results confirm that genetic algorithms can determine the optimal solution with the same degree of accuracy as the conventional gradient search optimisation. Genetic algorithms increase the computational effort of control law calculation by

approximately a factor of 6, but a count of the number of MFLOP required to execute the nonlinear predictive controller using genetic algorithms shows that the controller can be implemented on a DSP board within the sampling time constraints. The nonlinear controller using gradient algorithm optimisation runs 13.5 MFLOP per second and a DSP board can execute 33.3 MFLOP per second. If execution time is not crucial, genetic algorithms should be used in preference to gradient search optimisation because this method is more likely to find the global minimum in a nonlinear error surface than a gradient search algorithm.

Least squares optimisation is used for the fuzzified linear controller. The primary advantage of this method is that it can be calculated very quickly. The number of MFLOP required to execute the fuzzified linear controller 123 times less than that for the nonlinear predictive controller using gradient search optimisation. The disadvantage of this method is that it is unable to handle constraints. This strongly outweighs the "advantage" of a fast control solution for several reasons. The boiler process is relatively slow and does not require a fast control algorithm, computational power is cheap and the ability to operate the boiler on the constraints is important in order to achieve maximum efficiency and plant response times.

Implementation Issues

Both the fuzzified linear and nonlinear predictive controllers are feasible control solutions in that it is possible to execute both controllers within the sampling period constraints on a DSP board.

The fuzzified linear control strategy requires 7 sensors and the nonlinear control strategy requires 18. There is no absolute restrictions on the number of sensors which may be used, but in practical terms, increasing the number of sensors, increases the risk of using a badly calibrated sensor. A badly calibrated sensor has consequences in terms of model accuracy. For that reason, it would be preferable to reduce the number of sensors required by the neural network model.

The fundamental concepts of predictive control can be readily understood and conceptualised. Controller objectives and controller performance can be presented visually by displaying a receding horizon plot of the desired plant output, the measured plant output, and optionally the projected plant output. An reasonably easily understood methodology is very important in terms of implementation of the controller in a boiler plant. A decision to implement an advanced control methodology is generally based both on human factors, such as acceptance by plant engineers and operators, as well as on controller performance.

Controller tuning is based upon a physically meaningful tuning parameter, the time constant of the desired reference trajectory rather than on cost function weighting matrices.

Summary

To summarise, the many advantages of neural networks models, such as their ability to model any nonlinear function and their accuracy are to some extent outweighed by the fact that a predictive controller based upon a neural network model cannot incorporate inherent integral action. Given, that the fuzzified linear controller can achieve almost comparable performance to the nonlinear controller and includes in-built integral action, it is felt to be a more effective control strategy. The power of the fuzzified linear controller can be attributed to a fuzzy logic interpolation technique, which has been demonstrated to achieve good performance at all operating points. The performance of the fuzzified linear controller could be further improved by replacing the linear least squares optimisation method with genetic algorithm optimisation.

Future Work

Two areas are identified for further investigation;

The potential of predictive control could be examined further. For example, the ability of the controller to force plant output to follow a specified trajectory could be exploited to optimise the shape and speed of the reference trajectory profile with respect to a plant performance criteria, such as efficiency or thermal stress reduction. In addition, it should be investigated whether a more sophisticated cost function definition can improve controller performance further. For example, the control horizon could be set to a value greater than one, i.e. the projected plant output is not assumed to remain constant after the first sampling period.

The performance of the nonlinear controller could be improved if inherent integral action could be included in the control law. Mechanisms for ensuring that neural networks can reach steady state conditions should be investigated.

References

1. Alatiq. I.M. and Meziou, A.M., "Simulation and Parameter Scheduling Operation of Waste Heat Steam-Boilers", *Computers Chemical Engineering*, Vol. 16, No. 1, pp. 51-59, 1982.

2. Åkesson, I, "Boiler Steam Temperature Control Using a True Adaptive Regulator", *IEE 4th Workshop on Selftuning and Adaptive Control*, 1987.

3. Anakwa, W.K.N. and Swamy, M.N.S., "Design of a Minimum Order Dynamic Pole Placement Controller for a Boiler", *19th Asilomar Conf., Circuits and Computers*, Pacific Grove, CA, USA, 1985.

4. Anderson, J.H., "Dynamic Control of a Power Boiler", *Proc. of Inst. Elec. Eng.*, Vol. 116, pp. 1257-1268, 1969.

5. Åström K.J. and Wittenmark, B. "On Self-Tuning Regulators ", *Automatica*, Vol. 9, pp. 185-199, 1973.

6. Åström K.J. and Bell, R.D., "A Nonlinear Model for Steam Generation Processes", *Proc. IFAC*, pp. 395-398, Sydney, Australia, 1993.

7. Åström K.J. and Bell, R.D., "A Fourth Order Nonlinear Model for Drum-Boiler Dynamics", *Proc. IFAC*, pp. 31-36, San Francisco, USA, 1996.

8. Bhat, N.V., Minderman, Jr., P. A., McAvoy, T. and Wang, M.S., "Modelling Chemical Process Systems via Neural Computation", *IEEE Control Systems Magazine*, April, 1990.

9. Broderick, J.. Pellegrinetti, G., Bentsman, J. and Blauwkamp. R., "Application of Robust Adaptive Control to a Real-Time Power Plant Model", *Proc. of the 4th IEEE Conf. on Control Applications*, pp. 403-409, Albany, USA.

10. Brown, M.D., Irwin, G.W., Hogg, B.W. and Swidenbank, E., "Modelling and Control of Generating Units using Neural Network Techniques", *Proc. of the 3rd IEEE Conf. on Control Applications*, Vol. 1, pp. 735-740. Glasgow, U.K., 1994.

11. Campo, P.J. and Morari, M., "∞-Norm Formulation of Model Predictive Control Problems, *Proc. American Control Conf.*, pp. 339-343, Seattle, Washington, USA, 1986.

12. Chien, K., Ergin, E.I., Ling, C. and Lee, A. "Dynamic Analysis of a Boiler", *4th Instruments and Regulators Conf. of the ASME*, Delaware, USA, pp. 1809-1819, April, 1958.

13. Chipperfield, A.J., Fleming, P.J. and Fonseca, C.M., "Genetic Algorithm Tools for Control Systems Engineering", *Adaptive Computing in Engineering Design and Control*, Plymouth, UK, September, 1994.

14. Chisci, L. and Mosca, E., "Stabilising Predictive Control: the Singular Transition Matrix Case", *Advances in Model-Based Predictive Control*, Oxford University Press, pp. 122-130, 1994.

15. Chow, C.M. and Clarke, D.W.m "Actuator Nonlinearities in Predictive Control", *Advances in Model-Based Predictive Control*, Oxford University Press, pp. 245-259, 1994

16. Clarke, D.W. and Gawthrop, P.J., "A Self-Tuning Controller", *IEE Proc. D*, Vol. 123, pp. 633-640, 1979.

17. Clarke, D.W., Mohtadi, C. and Tuffs, P.S., "Generalised Predictive Control - Part I. The Basic Algorithm", *Automatica*, Vol. 3, No. 2, pp. 137-148, 1987a.

18. Clarke, D.W., Mohtadi, C. and Tuffs, P.S., "Generalised Predictive Control - Part II. Extensions and Interpretations ", *Automatica*, Vol. 3, No. 2, pp. 149-160, 1987b.

19. Clarke, D.W., "Application of Generalised Predictive Control to Industrial Processes", *IEEE Control Systems Magazine*, Vol. 8, No. 2, pp. 49-55, 1988.

20. Clarke, D.W. and Scattolini, R., "Constrained Receding-Horizon Predictive Control", *IEE Proc. D,* Vol. 138, No. 4, pp. 347-354, July, 1991.

21. Cori, R. and Maffezzoni, C., "Practical-Optimal Control of a Drum Boiler Power Plant", *Automatica*, Vol. 20, No. 2, pp. 163-173, 1984.

22. Cuttler, C.R. and Ramaker, B.L., "Dynamic Matrix Control - A Computer Control Algorithm", *Proc. JACC*, Vol. WP5-B, San Francisco, CA, USA, 1980

23. Dai. H. and Thompson, J.W. "Use of Neural Networks for Modelling the Steam Generator of a Nuclear Power Plant", *1994 Canadian Conf. on Electrical and Computer Eng.*, Halifax, Canada, pp. 734-738, September, 1994.

24. Demuth, H. and Beale, M., *Neural Network Toolbox User's Guide,* The MATHWORKS Inc., January 1994.

25. de Mello, F.P. "Boiler Models for System Dynamic Performance Studies", *IEEE Transactions on Power Systems*, Vol. 6, No. 1, pp. 66-74, February 1991.

26. Dieck-Assad, G., Masada, G.Y. and Flake, R.H., "Optimal Set-Point Scheduling in a Boiler Turbine System", *IEEE Trans. on Energy Conversion,* Vol. EC-2, No. 3, September (1987).

27. Ding. Z. and Hogg, B.W., "Multivariable Self-tuning Control of a Boiler-turbine System", *IEE Int. Conf. on Advances in Power System Control, Operation and Management,* Hong Kong, November, 1991.

28. Dukelow, S.G. *The Control of.Boilers*, Instrument Society of America, 1986.

29. Elshafei, A.L., Abdel-Magied, M., Rashad, H. and Bahgat. A. "Design and Implementation of a Laguerre Self-Tuner for Boiler Drum-Level Control", *Proc. of the American Control Conf.*, pp. 2024-2028, Seattle, USA, June 1995.

30. El-Rabaie, N.M. and Hogg, B.W., "Self-Tuning Control of Steam Pressure in a Drum Boiler", *Proc. of the 24th University Power Engineers Conf.*, Belfast, U.K., September, 1989a.

31. El-Rabaie, N.M. and Hogg, B.W., "Application of Multivariable Generalised Predictive Control to a Boiler System", *IEE Int. Conf. on Control*, Vol. 1, pp. 461-465, Edinburgh, UK, March, 1991

32. El-Sayed, M.A.H., Eteiba, M.B. and Sheirah, M.A., "Self-tuning Control of Thermal Power Plants", *Int. Journal of Electrical Power and Energy Systems*, Vol. 11, No. 1, January, 1989.

33. Fernandes-del-Busto, R., Sánches, E., Urbieta, P., Kaufman, H., "Application of Model Reference Adaptive Agorithm to Combustion Control of a Power Unit Boiler", *American Control Conf.*, pp. 618-619, Boston, MA, USA, 1985.

34. Fessl, J., "An Application of Multivariable Self-Tuning Regulators to Drum Boiler Control", *Automatica*, Vol. 22, No. 5, pp. 581-585, 1986.

35. Fujiwara, T. and Miyakawa, H., "Application of Predictive Adaptive Control System for Steam Temperature Control in a Boiler Plant", *Proc. of the 29th Conf. on Descision and Control*, pp. 2181-2182, Honolulu, Hawaii, December, 1990.

36. García, C.E., Prett, D.M. and Morari, M., "Model Predictive Control: Theory and Practice - a Survey", *Automatica*, Vol. 25, No. 3, pp. 335-349, 1989

37. Gibbs, B.P. and Weber, D.S., "Nonlinear Model Predictive Control for Fossil Power Plants, *Proc. of the American Control Conf.*, Vol. 4, pp 3091-3097, June, 1992.

38. Goldberg, D.E., *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison-Wesley, 1989.

39. Goodwin, G. and Sin, K., *Adaptive Filtering, Prediction and Control*, Prentice-Hall, INC Englewood Cliffs, New Jersey, 1984.

40. Hanby, V.I. "Combustion and Pollution Control in Heating Systems", Springer-Verlag London Limited, 1994.

41. Hogg, B.W. and El-Rabaie, N.M., "Generalised Predictive Control of Steam Pressure in a Drum Boiler", *IEEE Trans. on Energy Conversion*, Vol. 5, No. 3, pp. 485-492, 1990.

42. Hogg, B.W. and El-Rabaie, N.M., "Multivariable Generalised Predictive Control of a Boiler System", *IEEE Trans. on Energy Conversion*, Vol. 6, No. 2, pp. 282-288, 1991.

43. Horowitz, I.M., *Synthesis of Feedback Systems*, Academic Press, Lonfon, 1963.

44. Hunt, K.J., Sbarbaro, D., Zbikowski, R. and Gawthrop, P.J., "Neural Networks for Control Systems - A Survey", *Automatica*, Vol 28, No. 6. pp. 1083-112, 1992.

45. Irwin, G., Brown, M., Hogg, B. and Swidenbank, E., "Neural Network Modelling of a 200 MW Boiler System", *IEE Proc. Control Theory Applications*, Vol. 142, No. 8, pp. 529-535, November, 1995.

46. Kitami, T., Mizutani, H., Fujimoto, J. Yamada, T., Uchida, M.m Nakamura, H., "Inverse Response Control Method for Steam Temperature of Thermal Power Unit", *7th World IFAC Congress*, pp. 81-88, Helsinki, 1978.

47. Kristinsson, K. and Dumony, G.A., "System Identification and Control Using Genetic Algorithms", *IEEE Trans. on Systems, Man. and Cybernetics*, Vol. 22, No. 5, 1992.

48. Kwan, H.W. and Anderson, J.M. "A Mathematical Model of a 200 MW Boiler", *International Journal of Control*, Vol. 12, No. 6, pp. 977-998, 1970.

49. Kwon, W.H. and Pearson, A.E., "On the Stabilization of a Discrete Constant Linear System", *IEEE Trans. Automatic Control*, Vol. 20, No. 6, pp. 800-801, 1975.

50. Lindsley, D., *Boiler Control Systems*, McGraw-Hill Book Company, 1991.

51. Linkens D.A. and Mahfouf, M. "Generalised Predictive Control (GPC) in Clinical Anaesthesia", *Advances in Model-Based Predictive Control*, Oxford University Press, pp. 429-445, 1994.

52. Manayathara, T., Bentsman, J., Pellegrinetti, G. and Blauwkamp, R., "Application of Stabilising Predictive Control to Boilers", *Proc. of th 3rd IEEE Conf. on Control Applications*, pp. 1637-1642, Glasgow, U.K., August, 1994.

53. Man Gyun Na and Hee Cheon No. "Design of Model Reference Adaptive Control System for Steam Generators", *Nuclear Engineering and Design*, Vol. 122, pp. 301-311, 1990.

54. *MATLAB User's Guide,* The MATHWORKS Inc., 1993

55. Matsumura, S., Ogata, K., Fujii, S., Shioya, H. and Nakamura, H., "Adaptive Control for Steam Temperature of Thermal Power Plant", *Proc. 12th IFAC*, pp. 697-703, Sydney, Australia, 1993.

56. Matsumura, S., Ogata, K., Fujii, S., Shioya, H. and Nakamura, H., "Adaptive Control for the Steam Temperature of Thermal Power Plant", *Control Eng. Practice*, Vol. 2, No. 4, pp. 567-575, 1994.

57. McDonald, J.P., Kwatny, H.G. and Spare, J. H. "A Nonlinear Model for Reheat Boiler Turbine-Generator Systems Parts I and II - Development", *Proceedings of the 12th JACC*, St. Louis, pp. 227-236,1971.

58. McDonald, J. and Kwatny, H., "Design and Analysis of Boiler-Turbine-Generator Controls Using Optimal Linear Regulator Theory", *IEEE Trans. on Automatic Control*, Vol. AC-18, No.3, pp. 202209, 1973.

59. Morari, M. and Zafiriou, E., *Robust Process Control*, Prentice-Hall, 1989.

60. Nakamura, H. and Akaike, H., "Statistical Identification for Optimal Control of Supercritical Thermal Power Plants", *Automatica*, Vol. 17, No. 1, pp. 143-155, 1981.

61. Nakamura H. and Uchida, M., "Optimal Regulation for Thermal Power Plants", *IEEE Control Systems Magazine*, Vol. 9, No. 1, pp. 33-38, 1989.

62. Narenda, K.S. and Parthasarathy, K., "Identification and Control of Dynamical Systems using Neural Networks", *IEEE Trans. on Neural Networks*, Vol. 1, No. 1 pp. 4-27, March 1990.

63. Nicholson, H. "Dynamic optimisation of a boiler", *Proceedings of the IEE*, Vol. 111, No. 8, pp. 1479-1499, August 1964.

64. Nomura, M. and Sato, Y., "Adaptive Optimal Control of Steam Temperatures for Thermal Power Plants", *IEEE Trans. on Energy Conversion*, Vol. 4, No. 1, pp. 25-33, March, 1989.

65. Pellegrinetti, G., Bentsman, J. and Kameshwar, P., "Control of Nonlinear Steam Generation Process Using H_∞ Design", *Proc. American Control Conf.* , pp. 1292-1297, Boston, USA, June, 1991.

66. Pellegrinetti, G. and Bentsman, J., "Control of Nonlinear Steam Generation Process with Time Lags Using a H_∞ Design", *Proc. American Control Conf.* , pp. 1839-1843, Chicago, USA, 1992.

67. Pellegrinetti, G. and Bentsman, J., "Nonlinear Control Oriented Boiler Modeling - A Benchmark Problem for Controller Design", *IEEE Trans. on Control Systems Technology*, Vol. 4, No. 1, pp. 57-64, January, 1996.

68. Ping Yang and Hogg, B.W., "Continuous-Time Generalised Predictive Control of a Boiler Model", *IFAC Control of Power Plants and Power Systems*, pp. 93-96, Munich, Germany, 1992.

69. Rawlings, J.B. and Muske, K.B., "The Stability of Constrained Receding Horizon Control", *IEEE Trans. Automatic Control*, Vol. AC-38, No. 10, pp. 1512-1516, 1993.

70. Reinschmidt, K. and Ling, B., "Neural Networks for Plant Simulation and Control", *Proc. of the American Power Conf.*, pp. 796-801, Chicago, USA, 1994.

71. Richalet, J., Rault, A., Testud, J.L. and Papon, J., "Model Predictive Heuristic Control: Applications to Industrial Processes", *Automatica*, Vol. 14, pp. 413-428, 1978.

72. Richalet, J. *Practique de la Commande Predictive*, Hermes, 1993a

73. Richalet, J., Rault, A., Testud, J.L. and Papon, J,, "Industrial Applications of Model Predictive Control", *Automatica*, Vol. 29, pp. 1251-1274,1993b.

74. Robinson, B.D. and Clarke, D.W., "Robustness Effects of a Prefilter in Generalised Predictive Control", *IEE Proc.-D*, Vol. 138, No. 1, pp. 2-8, January, 1991.

75. Rohrs, C.E., Valavani, L., Athans, M. and Stein, G., "Robustness of Continuous-Time Adaptive Control Algorithms in the Presence of Unmodeled Dynamics", *IEEE Trans. on Automatic Control*, Vol. AC-30, No. 9, pp. 881-889, September, 1985.

76. Rossiter, J.A., Kouvaritakis, B., Dunnett, R.M., "Application of Generalised Predictive Control to a Boiler-Turbine Unit for Electricity Generation", *IEE Proc. D*, Vol. 138, No. 1, pp 59-67, Janurary, 1991.

77. Rovnak, J.A. and Corlis, R., "Dynamic Matrix Based Control of Fossil Power Plants", *IEEE Trans. on Energy Conversion,* Vol. 6, No. 2, pp. 320-326, 1991.

78. Rumelhart, D.E., Hinton, G.E. and Williams, R.J., "Learning Internal Representations by Error Propogation", *Parallel Distributed Processing*, Vol.1 (Eds. D.E. Rummelhart and J.L. McClelland), MIT Press, 1986

79. Sato, Y., Nomura, M. and Matsumoto, H., "Steam Temperature Prediction Control for Thermal Power Plant", *IEEE Trans. on Power Apparatus and Systems*, Vol. PAS-103, No. 9, September, 1984

80. *SIMULINK User's Guide,* The MATHWORKS Inc., 1993

81. Smith, O.J.M, *Feedback Control Systems*, McGraw-Hill, New York, 1958.

82. Soeterboek, A.R.M., Peis, A.F., Pels. A.F., Verbruggen, H.B. and Van Langen, G.C.A., "A Predictive Controller for the Mach Number in a Transonic Wind Tunnel", *IEEE Control Systems Magazine,* 1991.

83. Soeterboek, A.R.M, *Predictive Control - A Unified Approach*, Prentice-Hall, 1992..

84. Sommer, St., "Model-Based Predictive Control Methods based on Non-Linear and Bilinear Parametric System Descriptions, *Advances in Model-Based Predictive Control*, Oxford University Press, pp. 192-204, 1994.

85. Sun Demin, Xian Hui, Cao Shuhua and Peng Lixing, "Adaptive PID Controller with Parameter Optimization for Main Steam Pressure Control of an Industrial Boiler of 20T/H", *Chinese Journal of Automation*, Vol. 4, No. 1, pp 85-90.

86. Tada, H.A., *Operations Research: An Introduction*, Collier MacMillan, 1987

87. Tysso, A. "Modelling and Parameter Estimation of a Ship Boiler", *Automatica*, Vol. 17, No. 1, pp. 157-166, 1981.

88. Uchida, M., Hirosaki, T., Toyoda, Y. and Nakamura, H., "Practical Model Reference Adaptive Feedforward Control System of a Variable Pressure Boiler by the Computing Network", *Electrical Engineering in Japan*, Vol. 108, No. 5, pp 112-120, 1988.

89. Unbehauen, H., "The Load Dependent Multivariable Steam Temperature Control System in a Boiler", *Automatica,* Vol. 5, pp. 421-432, 1969

90. Unbehauen, H., Keuchel, U. and Kocaarslan, I. "Real-Time Adaptive Control of Electrical Power and Enthalpy for a 750 MW Once-Through Boiler", *IEE Int. Conf. Control,* Edinburgh, U.K., March 1991.

91. Unbehauen, H. and Kocaarslan, I. "Experimental Modelling and Adaptive Power Control of a 750 MW Once-Through Boiler", *Automatic Control World Congress*, Tallin, USSR, pp 32-37, August, 1990

92. Ydstie, B.E., "Extended Horizon Adaptive Control", *IFAC World Congress*, Budapest, Hungary, 1984.

93. Werbos, P.J., "Beyond Regression: New Tools for Prediction and Analysis in the Behavior Sciences: Ph.D Thesis", Harvard University, Committee on Applied Mathermatics, 1974.

94. Wu, G., Peng. L. and Sun, D. "Application of Stair-Like Generalised Predictive Control to Industrial Boiler", *"Proc. of the IEEE Int. Symp. on Industrial Electronics*, Vol. 1, Xian, China, May. 1992.

95. Zengqiang, C., Zhuzhi, Y., Yumei, L., Ping, H., Wemin, W. and Baomin, G., "Weighting Predictive Self-Tuning Control of Industrial Boiler", *Chinese Journal of Automation*, Vol. 5, No. 1, pp. 65-72, 1993.

Appendix A. Nomenclature for Chapter 4

ρ	=	Density	$kg\ m^{-3}$
c	=	Specific Heat at Constant Pressure	$kJ\ (kg^{-1}\ K^{-1}$
h	=	Enthalpy	$kJ\ kg^{-1}$
k	=	Heat Transfer Coefficient	
m	=	Mass Flow Rate	$kg\ s^{-1}$
v	=	Velocity	$m\ s^{-1}$
x	=	Steam Quality	
A	=	Area	m^2
C	=	Calorific Value	$kJ\ kg^{-1}$
D	=	Diameter	m
L	=	Length	m
M	=	Mass	kg
N	=	Number of Pipes	
P	=	Pressure	Pa
Q	=	Heat Transfer Rate	$kJ\ s^{-1}$
T	=	Temperature	°C
V	=	Volume	m^3

Subscripts

air	=	Air
d	=	Drum
ds	=	Drum Steam
dw	=	Drum Water
do	=	Downcomers
e	=	Economiser
fu	=	Fuel
g	=	Gas
gm	=	Gas-Metal
l	=	Load
m	=	Metal
mf	=	Metal-Fluid
r	=	Risers
s	=	Superheater

Sub-subscripts

i	=	inlet
o	=	outlet