UNIVERSITAT POLITÉCNICA DE CATALUNYA

ESCOLA TÈCNICA SUPERIOR D'ENGINYERIA INDUSTRIAL DE BARCELONA

MASTER THESIS

Multivariable control of a steam boiler

Author:
Abrahan A. RIERA CHIQUITO

Supervisor:
Dr. Vicenç PUIG CAYUELA

A thesis submitted in fulfillment of the requirements for the degree of Master in Automatic Control and Robotics

in the

Institut de Robòtica i Informàtica Industrial (IRI)
Departament d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial (ESAII)

September 14, 2017

Abstract

Multivariable control of a steam boiler
by Abrahan A. RIERA CHIQUITO

Keywords: IMC, LQR, Identification, State Space model, Transfer Function model, Steam Boiler, Industrial, CEA.

This thesis is devoted to apply a Multi-Input Multi-Output (MIMO) controller to a specific Steam Boiler Plant. The considered plant is based on the descriptions obtained from the input/output data of a referenced steam boiler in the Abbot combined cycle plant in Champaign, Illinois. The objective is to take all the useful input/output data from the steam boiler according to its performance and capability in different operation points in order to model the most accurate plant for control. The conceived case of study is based in a modification of a model proposed by Pellegrinetti and Bentsman in 1996, considering to be tested under a benchmark proposed by the Control Spanish Association (CEA).

Initially, taking into account only the input and output data of the system, black box modeling techniques were used to obtain different models of the plant. The first approach was to obtain a transfer function model to apply a Internal Model Controller (IMC). However the result was not as expected because the controller becomes considerable difficult to tune given the big quantity of poles and zeros of the resulting IMC controller. Hence this technique was dismissed.

On a second stage, it was obtained a model of the plant in state space representation to apply a Linear-Quadratic Regulator (LQR) technique to understand how the system behaves with this state space model design. Given that the description of the system in this form was more accurate the obtained results were better for this type of controller making it better suited to fulfill the needs of the plant.

This work covers all the steps followed to use the Internal Model Controller (IMC) and the Linear-Quadratic Regulator (LQR) techniques to study the behavior of a steam boiler system in an industrial environment. The obtained results are exposed and explained with the aim of describing which one of the two used methods is better suited for the control of the plant. Finally a budget and impact studies are presented to explain which could be the resources needed in order to apply this type of controllers effectively in a steam boiler plant, being able to extrapolate the obtained results to be applied to other type of processes in the same sector (heat exchangers, distillation columns, etc.).

Acknowledgements

This thesis represents the end of an enriching period for my personal and professional life. In this educational experience, there are people who deserve special recognition because without their valuable support and help this work would not have been possible.

First, I would like to thanks to my tutor: Vicenç Puig for his patience and support.

My thanks also go to my colleagues Michael, Eduardo, Esteban, Daniel and Victor for their remarkable friendship, wise advices and unconditional support.

And last but not least I want to express my special gratitude to my wife Maria, and to my parents and inlaws: Amparo, José, Elide and Jesús for their special support and caring. You all are the motor that drives me each day to be better.

Contents

Abstract **iii**

Acknowledgements **v**

1 Introduction **1**

1.1 Motivation . . . 1

1.2 Objectives and Scope . . . 2

1.2.1 Specific Objectives . . . 2

1.2.2 Scope of research . . . 2

1.3 Outline . . . 3

2 Background **5**

2.1 Industrial Steam Boilers . . . 5

2.1.1 Conditions of the analyzed model . . . 11

2.2 Modeling of a plant . . . 12

2.3 Internal model controller (IMC) . . . 15

2.3.1 IMC control scheme . . . 15

2.3.2 Internal stability and benefits . . . 17

2.4 Linear-quadratic regulator controller (LQR) . . . 17

2.4.1 Finite-horizon, discrete-time LQR . . . 18

2.4.2 Limitations . . . 18

3 Methodology **21**

3.1 Encrypted Model . . . 21

3.2 Identification of the Plant . . . 23

3.2.1 Transfer Function identification . . . 23

3.2.2 State Space identification . . . 27

3.3 IMC controller implementation . . . 30

3.4 LQR controller implementation . . . 31

4 Results **37**

4.1 IMC implementation results . . . 37

4.2 LQR implementation results . . . 38

4.2.1 Evaluation of controller with benchmark code . . . 47

5 Budget and Impact Studies **55**
5.1 Budget Study . . . 55
5.1.1 Hardware Resosurces . . . 55
5.1.2 Software Resources . . . 55
5.1.3 Human Resources . . . 55
5.1.4 General Resources . . . 56
5.1.5 Total Cost . . . 56
5.2 Impact Study . . . 56
5.2.1 Economical Impact . . . 56
5.2.2 Social Impact . . . 57
5.2.3 Environmental Impact . . . 57

6 Concluding Remarks **59**
6.1 Conclusions . . . 59
6.2 Contributions . . . 60
6.3 Future Work . . . 60

Bibliography **61**

List of Figures

2.1 Industrial Steam Generator Plant . . . 7
2.2 MIMO block of the boiler and its internal structure . . . 12
2.3 Types of Models . . . 13
2.4 Forward IMC control scheme . . . 16
2.5 LQR control scheme . . . 19

3.1 Dynamic inputs of the encrypted model . . . 22
3.2 Fix input Steam demand variation . . . 22
3.3 Controlled outputs of the encrypted model . . . 23
3.4 Simulated outputs of the transfer function model . . . 26
3.5 Parameters of the model in state space . . . 28
3.6 Simulated outputs of the state space model . . . 29
3.7 One step horizon of the state space model . . . 30
3.8 IMC implementation on Simulink . . . 31
3.9 LQR implementation on Simulink . . . 35

4.1 IMC results with a simple $G_c(s)$ . . . 38
4.2 SS model controlled with the LQR controller . . . 39
4.3 Open loop response of the plant . . . 40
4.4 Scenario 1: Close loop LQR controller without steam demand variation, $R = diag(1e-3)$ and $Q = diag(20)$ . . . 42
4.5 Scenario 2: Close loop LQR controller without steam demand variation, $R = diag(1e-3)$, $Q = diag(20)$ and $Rate = 0.1$ . . . 43
4.6 Scenario 3: Close loop LQR controller with steam demand variation, $R = diag(1e-3)$ and $Q = diag(20)$ . . . 45
4.7 Scenario 4: Close loop LQR controller with steam demand variation, $R = diag(1e-3)$, $Q = diag(20)$ and $Rate = 0.5$ . . . 46
4.8 Benchmark of K_3 without steam demand variation ($J = 31.2518$) . . . 48
4.9 Benchmark of K_4 without steam demand variation ($J = 33.3107$) . . . 49
4.10 Benchmark of K_6 without steam demand variation ($J = 29.0377$) . . . 50
4.11 Benchmark of K_3 with steam demand variation ($J = 83.7495$) . . . 51
4.12 Benchmark of K_4 with steam demand variation ($J = 65.6681$) . . . 52
4.13 Benchmark of K_6 with steam demand variation ($J = 75.5940$) . . . 53

List of Tables

2.1 Initial conditions of the Steam Boiler 12
2.2 Parameterizations of the system components (Mathworks, 2016) 14

3.1 Transfer function models parameters and FIT 24
3.2 Resulting transfer function models 25
3.3 Obtained Transfer Function model 25

5.1 Cost associated to hardware resources 55
5.2 Cost associated to software resources 56
5.3 Total cost associated to the thesis 56

List of Abbreviations

SBP	**S**team **B**oiler **P**lant
IMC	**I**nternal **M**odel **C**ontrol
LQR	**L**inear-**Q**uadratic **R**egulator
SISO	**S**ingle **I**nput - **S**ingle **O**utput
MIMO	**M**ulti **I**nput - **M**ulti **O**utput
TF	**T**ransfer **F**cuntion
SS	**S**tate **S**pace
PLC	**P**rogrammable **L**ogic **C**ontroller
SCADA	**S**upervisory **C**ontrol **A**nd **D**ata **A**cquisition

List of Symbols

$(a, b, ...)$	scalars	
	(Lower case letters)	
$(\mathbf{a}, \mathbf{b}, ...)$	column vectors	
	(Bold lowercase letters)	
$(\mathbf{A}, \mathbf{B}, ...)$	matrices	
	(Bold uppercase letters)	
$(\mathbb{A}, \mathbb{B}, ...)$	constraint sets	
	(Uppercase blackboard letters)	

Sets, Spaces and Set Operators

$\mathbb{R}$	set of real numbers	
$\mathbb{R}_{+0}$	set of non-negative real	
	numbers including zero	
$\mathbb{R}^n$	space of n-dimensional (column)	
	vectors with real entries	
$\mathbb{R}^{nxm}$	space of n by m matrices	
	with real entries	
$\mathbb{N}$	set of natural numbers	
	(non-negative integers)	$\mathbb{N}_+ := \mathbb{N} \setminus \{0\}$
$\mathbb{N}_{kj}$	set of consecutive	
	non-negative integers	$j, ..., k$
$(\subset) \subseteq$	(strict) subset	
$\setminus$	set minus	
x	cartesian product	$XxY = \{(x, y) \vert x\epsilon X, y\epsilon Y\}$

Model Theory

qf	fuel flow rate
qa	air flow rate
qfw	water flow rate
$QFCF$	maximum fuel flow rate
$QACA$	maximum air flow rate
$QCFW$	maximum water flow rate
CPi	constants for modeled equation
O_2	remaining oxygen
FAR	air to fuel mass ratio
$AIRO2$	rate of oxygen in the air

$TAIR$	time constant for air flow
VW	Volume of water inside the drum
VT	Total volume of the drum
x_i	states
u_i	inputs
y_i	outputs
c_{ij}	constants for the model

Systems and Control Theory

n_x	number of states	$n_x \epsilon +$
n_u	number of inputs	$n_u \epsilon +$
n_d	number of disturbances	$n_d \epsilon +$
x	state vector	$x \epsilon n_x$
u	control input vector	$u \epsilon n_u$
d	disturbance vector	$d \epsilon n_d$
X	set of admissible states	$X \subset n_x$
U	set of admissible control inputs	$U \subset n_u$

Chapter 1

Introduction

1.1 Motivation

Nowadays the efficient use of renewable energies is taking more importance inside the industry field, considering that the climatic change is a reality. Every year different countries meet in order to discuss new strategies that allow to stop the environmental damage caused by the industry. The creation of new technologies with the aim of using non-conventional different sources of energy has become an important factor to manage research resources.

There are already solutions developed in recent years that have been proved to be efficient and environmental friendly (e.g., sun energy, wind energy or tidal energy technologies). But the principal problem with this type of techniques is that the monetary cost is extremely high in comparison with the commonly used techniques.

Based on the economical fact just explained, the industries are having problems to apply this type of solutions. Hence, it is useless to create new technologies if they are not being implemented by the sectors that are affecting the environment.

This has driven many research groups around the world to focus in making the new energy generation technologies cost effective. In spite of that, it is important to find new ways of making more efficient the available methods that are found currently in the industrial sector.

One of the most popular systems in terms of energy generation is the case of the steam boiler. Steam boilers are used in a large type of industries given its capability of managing different pressures and temperatures, making possible the heat exchange necessary in different processes or even in electricity generation.

In terms of process control, the steam boilers are MIMO (Multi-Input Multi-Output) systems. Despite of that they are usually treated as single-input single-output (SISO) systems for the purposes of control design and implementation. In those cases, the tuning procedure is frequently carried out taking each loop individually. Even so, due to the complexity of the system the results obtained are often not as expected.

This problem does not only affect the boiler systems but a lot of other systems that belong to plants, which are equally described by MIMO systems. Additionally, in most of the cases these systems are non-linear, so they are not easily structured in the actual industrial controllers. The tuning of different loops and different variables that interact strongly with one another is the most difficult part when a plant is controlled automatically.

Given the lack of suitable MIMO tuning tools for industrial controllers, the goal is to be able to implement the technologies and new techniques in MIMO control on the actual industry plants. It will be a big step forward in industrial control to find the correct path to accomplish the interaction between all the variables, having a more detailed behavior of the systems. The accomplishment of the goal is difficult even when there are enough computational resources. The objective of this thesis is to show how the control of a MIMO system as the steam boiler can be developed and which could be the steps to follow in order of implement this type of controllers in the industrial environment.

1.2 Objectives and Scope

Considering the already explained popularity facts of SISO systems on industry, the main objective of this thesis is to implement a MIMO controller on a Steam Boiler plant proving that this type of approach could be implemented on a real plant.

1.2.1 Specific Objectives

In order to fulfill the mentioned objective, the following specific objectives are defined:

1. To obtain the model of a steam boiler system from a set of input/output data.
2. To validate this model based on the physical states and its correlation with the set of data.
3. To design and implement a MIMO controller suitable for the obtained process model.

1.2.2 Scope of research

The available data set describes the inputs and outputs of a real steam boiler. Based on this data set we proceed to use system identification techniques to obtain the transfer function (TF) model and the state space (SS) model of the steam boiler. Once the models have been obtained, the validation process compares the relation between the inputs and outputs of the plant, and it is observed how much this relation fitted with the actual relation found in the data set. When the FIT of a model was above 70% then the model was considered to be valid.

Once the TF and SS models were validated we proceed to design two different MIMO controllers based on the Internal Model Control (IMC) theory and the Linear-Quadratic Regulator (LQR) theory.

The obtained controllers were implemented in a realistic simulation of the plant in closed loop. Analyzing the obtained results, it was discussed which one of this two MIMO control approaches could be applied in a real industrial boiler plant, based on the performance of the controller and also its tuning capability.

1.3 Outline

This master thesis is structured in the following way:

- **Chapter 2: Background**

  This chapter reviews the control theory background, by defining the used methods and referencing the previous works developed in other relevant cases that allows to fulfill the current objectives.

- **Chapter 3: Methodology**

  In this chapter the objective is to describe the followed steps in order to fulfill the proposed objectives (model the steam boiler system, validate the model, design the MIMO controllers and implement the control strategies on the steam boiler plant simulation).

- **Chapter 4: Obtained results**

  This chapter presents and analyzes the obtained results of the modeled systems and the controller implementation. Also it is discussed which of the two strategies evaluated is more applicable into a real industrial boiler plant.

- **Chapter 5: Budget and impact**

  The impact of this thesis is discussed. The economic impact that this work has over a conventional project of a steam boiler inside an industry is described. Other social and environmental impacts are also discussed when applicable.

- **Chapter 6: Conclusions and contributions**

  Finally, an overall analysis is performed. It is analyzed if the principal objective of the thesis has been reached. If not, the possible causes and possible future work are described. Also it is discussed the contribution to the field of steam boilers control.

Chapter 2

Background

This chapter describes the steam boiler process and explains all process variables (states, input and outputs variables, etc.) needed to design the controller. After presenting the process model, the conditions of the model are described and finally the considered multivariable control approaches (IMC and LQR) are reviewed.

The first step is to understand how the system works. A case of study with data taken from a real plant will be analyzed. This data represents the significant dynamics of the boiler at Abbott Power Plant in Champaign, Illinois in the United States, in the normal regimes as in the feasible abnormal ones. This plant has been proposed as a benchmark problem to be solved in the Control Engineering Contest (CIC2016), organized by the Grupo Temático de Ingeniería de Control del Comité Español de Automática (CEA).

The model is encrypted, which means that the simulated plant (as a real one) has to be identified from input and output data, as a black box model, selecting the most convenient modeling method and then applying the controller according to the characteristics and parameters of the plant.

The useful information to be extracted from the plant are the variables that affect the dynamics of the system and how they are related between them. After this, every variable input or output can be tagged as measured, unmeasured, disturbance or control variable of the plant. With this information, the controllers can be designed according to how the model interacts with the system.

2.1 Industrial Steam Boilers

The steam boilers are capable to exchange in an efficient way the heat that is produced in some parts of the plant, even if the heat is concentrated inside the boiler or if it is produced by some other processes outside the boiler system.

The boilers are one of the basic elements of a plant and they have a lot of variations. Nowadays, most of the boilers are based on the classic model but it is not a secret that the technology is going faster and the boilers efficiency should improve along the years.

The objectives of the boilers is to find the range of exchange where the heat can be taken in the best way. This means finding the correct conditions of pressure and temperature. Every boiler is different, and they have different ways to be controlled. In the industrial field the controllers of any important system as this one needs to be robust and capable to be adapted to small changes at least.

A boiler incorporates a burner in order to burn the fuel and generate heat. The generated heat is transferred to water to make steam, known as the boiling process. The higher the burner temperature, the faster the steam production. The saturated steam thus produced can then either be used immediately to produce power via a turbine and alternator (Steingress,2001).

The boiler model is developed on the basis of physical laws, previous efforts in boiler modeling, known physical constants, plant data, and heuristic adjustments. The resulting fairly accurate model is nonlinear, fourth order, and include inverse response, time delays, measurement noise models, and a load of disturbance component (Pellegrinetti & Bentsman,1996).

The methods to obtain the correct model of any steam boiler plant are not readily found in open literature and are often specific to a particular system. This is particularly true on industrial environments where the signals given by any system can not be predicted until they are working on place because they can be very affected depending on the conditions of where the system is located, the standard work of the company and the users.

The boiler model which is used as a base on this chapter contains non-linearities, noise as the normal plant, time delays and disturbances that are going to be managed (Pellegrinetti & Bentsman,1996).

The behavior of the model represents the significant dynamics of the boiler at Abbott Power Plant in Champaign, IL in the United States, in normal regimes as in feasible abnormal ones. This model represents a complete boiler that predicts process response in terms of the measured outputs like drum pressure, drum water level and oxygen excess, to controllable input as air and fuel rates, steam demand rate and flow rate of water, and also the disturbances and noises (Pellegrinetti & Bentsman,1996).

In order to follow the equations that describes the steam boiler plant behavior, the Figure 2.1 presents an scheme of the conventional industrial steam generator plant:

The first group of equations relates the control of the steam demand valve position (u_1) to the input flow rates for the fuel valve (u_2), air valve (u_3) and feed water flow valve (u_4) rates respectively:

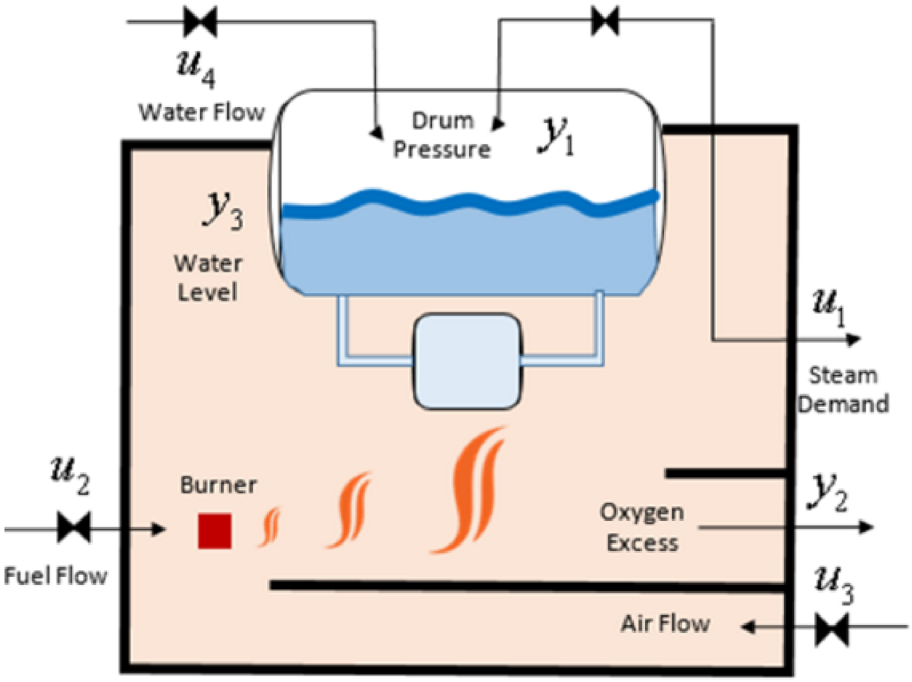


FIGURE 2.1: Industrial Steam Generator Plant (Blanco, 2016).

$$qf = QFCFu_2 \tag{2.1}$$

where qf is the fuel flow rate, $QFCF$ is the maximum fuel flow rate of the system, u_2 is the valve position from 0% to 100%.

$$qa = QACAu_3 \tag{2.2}$$

where qa is the air flow rate, $QACA$ is the maximum air flow rate of the system, u_3 is the valve position from 0% to 100%.

$$qfw = QCFWu_4 \tag{2.3}$$

where qfw is the water flow rate, $QCFW$ is the maximum water flow rate of the system, u_4 is the valve position from 0% to 100%.

The differential equation for the drum pressure depends on the variable x_4 which corresponds to an exogenous variable, the fuel flow rate u_2, and the water flow rate u_4 and is given as:

$$\dot{x_1} = -CP1x_4x_1^{9/8} + CP2u_2 - CP3u_4 + CP4 \tag{2.4}$$

$$\dot{x_1} = -0.00558x_4x_1^{9/8} + 0.0280u_2 - 0.01348u_4 + 0.02493$$

where $CP1$, $CP2$, $CP3$, $CP4$ are some empirical parameters, $\dot{x_1}$ is the drum pressure differential equation, x_1 is the pressure of the steam.

The oxygen level equation assumes complete combustion and in steady state can be represented as

$$O_2 = \frac{100(qa - qfFAR)}{(qa + qf)AIRO2} \tag{2.5}$$

where O_2 is the oxygen excess, FAR is the relation of air needed to consume all the fuel, $AIRO2$ is the percentage of oxygen in the air.

FAR is the air to fuel mass ratio for complete combustion (stoichiometric relation), and $AIRO2$ is the mass ratio of air to oxygen in atmospheric air (generally near the 0.2). Assuming a first order lag with time constant $TAIR$, the state equation of the oxygen can be expressed as

$$\dot{x_2} = [O_2 - x_2]\frac{1}{TAIR} \tag{2.6}$$

$$\dot{x_2} = [O_2 - x_2]\frac{1}{6.492}$$

where $\dot{x_2}$ is the oxygen excess rate, x_2 is the permitted oxygen excess.

But due to the nonlinear behaviour, a linear equation could not describe very well how the system evolves being necessary to create a new nonlinear relation to define better the behaviour of the constant FAR described as follows:

$$FAR = FA1O_2 + FA2 \tag{2.7}$$

$$FAR = 0.3106290O_2 + 16.2983$$

where $FA1$, $FA2$ are experimental parameters characterizing the relation fuel-air.

In some models as in this case, the steam demand is already defined and it will be treated as a measured disturbance input.

$$qs = (x_4CQS1 - CQS2)x_1 \tag{2.8}$$

$$qs = (x_4 0.855663 - 0.18128)x_1$$

where qs is the desired steam flow rate, $CQS1$, $CQS2$ are experimental parameters characterizing the steam flow rate.

The load level was computed from the steam flow rate and pressure. Then, the steady state and the states varying on time are the following

$$\dot{x_4} = -(x_4 - CD11u_1 - CD12)\frac{1}{TD1} \tag{2.9}$$

$$x_4 = CD11u_1 + CD12) \quad (2.10)$$

where $CD1$, $CD2$ are experimental parameters, $TD1$ is the time constant for the steam rate, x_4 is the variable produced from the steam flow and pressure data, $\dot{x_4}$ the rate related with the steam flow.

There are other complementary equations that help to define the rest of the states, for example the density of the steam with a constant temperature will be only dependent of the pressure inside the boiler and the density of the liquid has very low variations

$$rsh = CS1x_1 + CS2 \quad (2.11)$$

where rhs is the density inside the drum, $CS1$, $CS2$ are experimental parameters.

The volume of water inside the drum depends on this density, the evaporation flow rate (msd), the volume of water in the drum (vwd), the steam quality (a_1) and the energy given (ef)

$$ef = CU11qf + CU12 \quad (2.12)$$

$$ef = 37633qf + 174$$

where ef is the normal evaporation flow, $CU11$, $CU12$ are experimental parameters characterizing the evaporation flow.

$$a_1 = \frac{\frac{1}{x_3} - VW}{\frac{1}{rsh} - VW} \quad (2.13)$$

where a_1 is the relation for evaporation of water, VW is the volume of water.

$$msd = \frac{KBef - Rqfw + qsK}{1 + K} \quad (2.14)$$

where K is the quality of steam, R is the constant for quality of the stream.

$$vwd = VWVTx_3 + CVWD1a_1 + 0.159msd \quad (2.15)$$

where VT is the total volume of the drum, vwd is the volume of water in the drum.

The constants of the system are volume of the drum (VW, VT), the quality of the steam (K), etc. The 3rd state represents the density of the fluid (liquid and steam).

$$\dot{x_3} = \frac{QCFWu_3 - qs}{VT} \quad (2.16)$$

where $\dot{x_3}$ is the fluid density.

The outputs provide the proper scaling to match the Abbot boiler to the particular boiler. So, it is needed to do several conversions of units to adapt it to the international system:

$$y_1 = SCPx_1 \tag{2.17}$$

where y_1 is the pressure in PSI.

$$y_2 = x_2 \tag{2.18}$$

where y_2 is the oxygen level in %.

$$y_3 = SCWCXW1(vwd - CXW2) \tag{2.19}$$

where y_3 is the water level in inches.

$$y_4 = qs \tag{2.20}$$

where y_4 is the steam flow rate.

It is well known that when a model is more complex, it is supposed to be more faithful to the essentials of the plant dynamics but in terms of control, the objective is to use the simplest possible model but with the closest behavior to the reality.

In the Pellegrinetti and Bentsman model (Pellegrinetti & Bentsman,1996) the description of the state-space nonlinear model is described as follows:

$$\dot{x_1}(t) = c_{11}x_4(t)x_1(t) + c_{12}u_1(t - \tau_1) - c_{13}u_3(t - \tau_3) \tag{2.21}$$

$$\dot{x_2}(t) = c_{21}x_2(t) + \frac{c_{22}u_2(t - \tau_2) - c_{23}u_1(t - \tau_1) - c_{24}u_1(t - \tau_1)x_2(t)}{c_{25}u_2(t - \tau_2) - c_{26}u_1(t - \tau_1)} \tag{2.22}$$

$$\dot{x_3}(t) = -c_{31}x_1(t) + c_{32}x_4(t)x_1(t) + c_{33}u_3(t - \tau_3) \tag{2.23}$$

$$x_4(t) = -c_{41}x_4(t) + c_{42}u_1(t - \tau_1) + c_{43} \tag{2.24}$$

$$y_1(t) = c_{51}x_1(t - \tau_4) \tag{2.25}$$

$$y_2(t) = c_{61}x_2(t - \tau_5) \tag{2.26}$$

$$y_3(t) = c_{70}x_1(t - \tau_6) + c_{71}x_3(t - \tau_6) + c_{72}x_4(t - \tau_6)x_1(t - \tau_6) \tag{2.27}$$

$$+c_{74}u_1(t-\tau_3-\tau_6)+\frac{[c_{75}x_1(t-\tau_6)+c_{16}][1+c_{77}x_3(t-\tau_6)]}{x_3(t-\tau_6)[x_1(t-\tau_6)+c_{78}]}+c_{79}$$

$$y_4(t)=[c_{81}x_4(t-\tau_1)+c_{82}]x_1(t-\tau_1) \quad (2.28)$$

2.1.1 Conditions of the analyzed model

The model considered in this thesis is a modification of the one proposed by Pellegrinetti and Bentsmann (Pellegrinetti & Bentsman,1996). The physical and chemical basis should be the same but there are some things that were changed in order to adapt it to the benchmark considered in the control contest. This contest is organized by the Grupo Temático de Control de CEA and is part of the Control Engineering Contest (CIC2016). The objective suggested by the Department of Informatics and Automatics of the Universidad Nacional de Educación a Distancia (UNED) and directed by Fernando Morilla and Carlos Rodríguez, was to apply an effective controller to a modified Boiler plant that should follow in general the same conditions as the original.

The units are expressed according to the international system, being transformed the English units inside the model. The variables are based in a scale from 0 to 100 representing the percentage where the variables (inputs and outputs) are moving. For the inputs, it is a percentage representation of a normal control valve position, because they are totally open when its state is 100% and totally closed when this same state is 0%. For the outputs, the percentage represents the maximum and minimum value allowed by the system.

The fuel is described as a single component fuel, increasing the uncertainty. The pressure is supposed to be constant at 2.24 MPa (22.4 bar) and the steam demand is 150000 lb/hr (68038.86 Kg/h).

The variables to control are: drum pressure, level of the water in the drum and oxygen excess. These levels will be specified and need to be maintained despite the variation mainly on the steam demand. The desired steam pressure must be maintained as much the temperature grows (Comité Español de Automática,2016). Considering that there is a mixture of water and steam inside the boiler, it is interesting that more water input does not affect the temperature allowing to keep exchanging heat produced by the fuel into more steam and maintain the inside boiler pressure. This feature stabilizes the water temperature around the boiling point and contributes to the fact that there are no extremely hot areas inside the boiler due to safety measures.

The studied plant (Astrom & Bell, 1987) was based in a real one located in Illinois. The Abbott Plant and the boiler number two was the one measured to be modeled and the results of this model offered a nonlinear combustion equation and a model for excess of oxygen, including stoichiometric air-to-fuel ratio to combustion.

Furthermore, the boiler plant is a MIMO system with several internal interactions, every input affects several outputs. If the fuel input increases, the heat will cause the temperature of the water to increase, the mixture between water and steam will tend to steam and with this, the pressure inside the boiler is going to rise too, in contrast to the excess oxygen and water level that will decrease inside the drum.

2.2 Modeling of a plant

The plant has been described with an internal structure that cannot be studied given that the code is encrypted, so the method to obtain the model is to get the data from inputs and outputs, and save them with the plant parameters during some tested time (see Figure 2.2). There are some characteristics inside a plant that makes it unique and to identify it is necessary a detailed analysis of the input/output data (see Table 2.1).

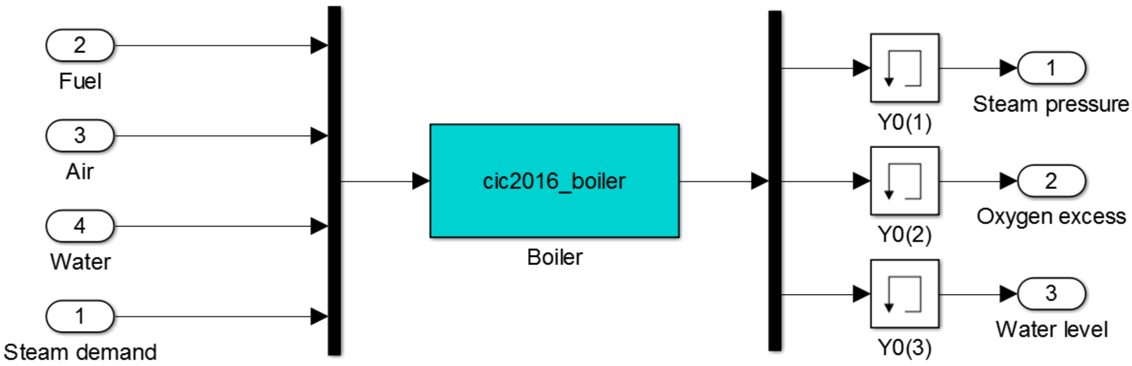


FIGURE 2.2: MIMO block of the boiler and its internal structure (Comité Español de Automática, 2016)

TABLE 2.1: Initial conditions of the Steam Boiler

Variable	Name	%	Type of Variable
Fuel Rate (2)	u2	40.59	Measured input
Air Rate (3)	u3	63.07	Measured input
Water Flowrate (4)	u4	35.06	Measured input
Drum Pressure (1)	y1	40.51	Measured output
Oxygen Excess (2)	y2	37.77	Measured output
Water Level (3)	y3	44.41	Measured output
Steam Demand (1)	u1	37.86	Measured disturbance

Figure 2.3 shows the type of models that can be identified using the MATLAB identification toolbox that includes both static and dynamic models. Alternatively, models can be white or black box. In the case of the white box type, a model is based on physical

laws and every detail is taken into account to have a good accuracy.

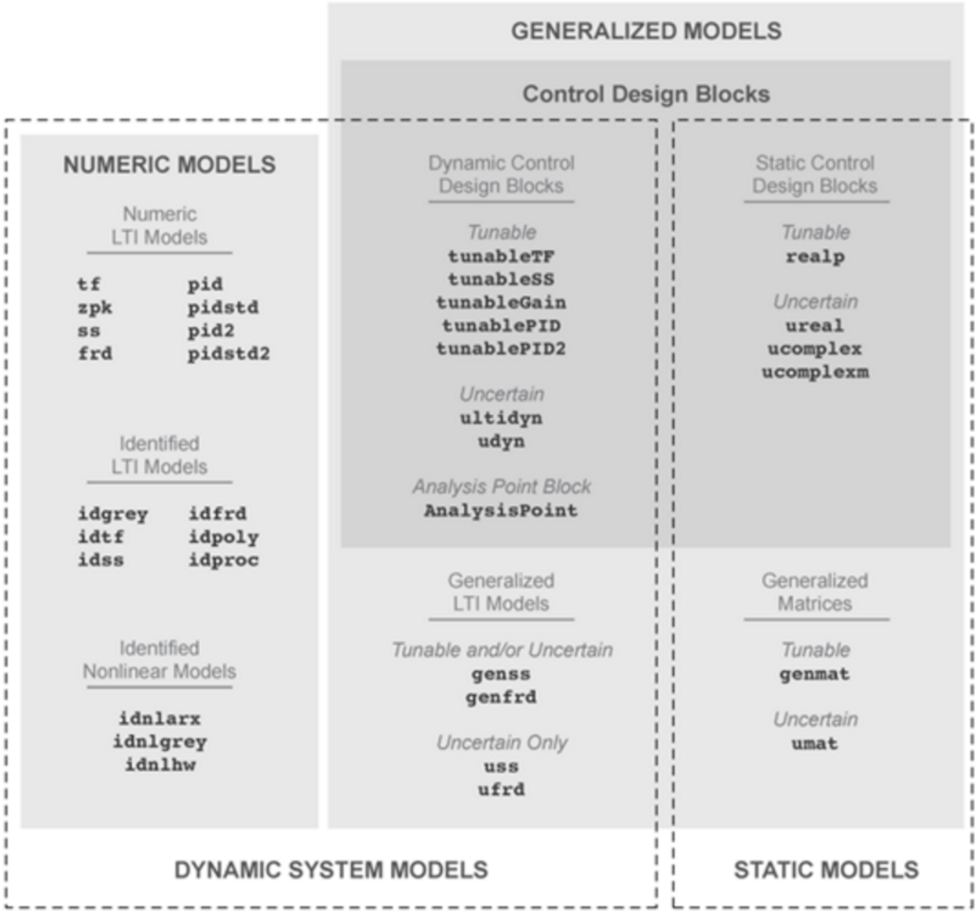


FIGURE 2.3: Types of Models (Mathworks, 2016).

The black box approach its applied, where it does not matter how the model works inside but the dynamics is tried to adjust as much as possible to the plant behavior considering only the information of the inputs and outputs. The majority of the numerical models assume a linear time-invariant (LTI) for the system.

Black-box modeling is usually a trial-and-error process, where parameters are estimated considering different structures and selecting the one that achieves best fitting between real and estimated data. Typically, black-box approach starts with the simple linear model structure and progress to more complex structures. It might also be chosen a model structure because it is familiar with this structure or because specific application needs.

In this case it is considered that the application needs the transfer function structure in order to be able to work with the IMC theorem and a state space structure in the case of the LQR theorem. This two types of models structures are used to find the controller that is applied to the plant.

The various linear model structures provide different ways to parameterize the transfer functions G and H. Being G the relation between the measured input and the measured output and H the relationship between the disturbances at the output and the measured output. When using input-output data to estimate a LTI model, you can configure the structure of both G and H, according to Table 2.2.

TABLE 2.2: Parameterizations of the system components (Mathworks, 2016)

Model Type	G and H functions
State space model	Represents an identified state-space model structure, governed by the equations: where the transfer function between the measured input u and output y is,and the noise transfer function is
Polynomial model	Represents a polynomial model such as ARX, ARMAX and BJ. An ARMAX model, for example, uses the input-output equation, so that the measured transfer function G is, while the noise transfer function is, The ARMAX model is a special configuration of the general polynomial model whose governing equation is: The autoregressive component, A, is common between the measured and noise components. The polynomials B and F constitute the measured component while the polynomials C and D constitute the noise component.
Transfer function model	Represents an identified transfer function model, which has no dynamic elements to model noise behavior. This object uses the trivial noise model H(s) = I.
Process model	Represents a process model, which provides options to represent the noise dynamics as either first- or second-order ARMAX process (that is, where C(s) and A(s) are harmonic polynomials of equal degree). The measured component, G(s), is represented by a transfer function expressed in pole-zero form.

Then , the industrial technology is very important to choose the best technique to model because not all the models can be applied. The most common controllers in the industry are the PLCs (Programmable Logic Controllers) and most of them manipulate the variables based on PID controllers, this means that they use SISO descriptions of the system and single loops to control real plants.

It is well known that the plants are very difficult to be described entirely as a SISO system and besides of it, linear. The real industry is not like that, a simple system has multiple nonlinear inputs and/or outputs, but the best way to implement and insert the required controller in this kind of equipment is through transfer functions, being this the reason why the interaction between all the outputs with all the inputs will be identified and analyzed with this method.

2.3 Internal model controller (IMC)

An internal model is a process that simulates the response of the system in order to estimate the outcome of a system disturbance. It stands in contrast to classical control, in that the classical feedback loop fails to explicitly model the controlled system (although the classical controller may contain an implicit model).

The internal model theory of control argues that the system is controlled by the constant interactions of the “plant” and the “controller.” The plant is the body part being controlled, while the internal model itself is considered part of the controller. Information from the controller, such as information from the Central Nervous System (CNS), feedback information, and the efference copy, is sent to the plant which moves accordingly.

Internal models can be controlled through either feed-forward or feedback control. Feed-forward control computes its input into a system using only the current state and its model of the system. It does not use feedback, so it cannot correct for errors in its control. In feedback control, some of the output of the system can be fed back into the system’s input, and the system is then able to make adjustments or compensate for errors from its desired output.

2.3.1 IMC control scheme

In their simplest form, IMC models takes the input (u) and output (y) of the plant ($G_p(s)$) to the model of the same plant ($G_{mp}(s)$) . The system input (u) to the model is an efference copy, as seen in Figure 2.4. The output from that model is then compared with the actual outputs of the system, including the possible disturbances ($w(s)$). The actual (plant value) and predicted (model value) output may differ due to noise introduced into the system by either internal (e.g. body sensors are not perfect, sensory noise) or external (e.g. unpredictable forces from outside the plant) sources. If the actual and predicted outputs differ, the difference can be feedback as an input into the entire system again so that an adjusted set of the control inputs can be formed in the controlled ($G_c(s)$) to create a corrected output.

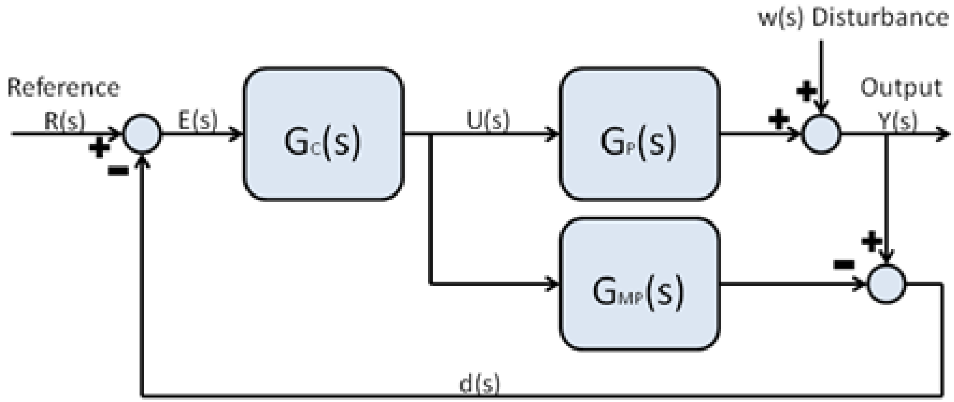


FIGURE 2.4: Forward IMC control scheme

where,

$$Y(s) = \frac{G_p(s)G_c(s)}{1 + G_c(s)(G_p(s) - G_{mp}(s))}R(s) + \frac{1 - G_{mp}(s)G_c(s)}{1 + G_c(s)(G_p(s) - G_{mp}(s))}w(s) \tag{2.29}$$

and having into account that the sensitivity function

$$S(s) = \frac{1}{1 + G_p(s)G_c(s)}$$

and the complementary sensitivity function

$$T(s) = \frac{G_p(s)G_c(s)}{1 + G_p(s)G_c(s)}$$

are $S(s) + T(s) = 1$, then we have:

$$Y(s) = T(s)R(s) + S(s)w(s) \tag{2.30}$$

In the case of plant/model mismatch ($G_p(s)$ = $G_{mp}(s)$), this functions simplify to:

$$\bar{T}(s) = G_{mp}(s)G_c(s)$$

$$\bar{S}(s) = 1 - \bar{T}(s) = 1 - G_{mp}(s)G_c(s)$$

$$G_c(s) = G_{mp}(s)^{-1}\bar{T}(s)$$

which lead to the following expressions for the input/output relationships between $Y(s)$, $U(s)$, $E(s)$ and $R(s)$, $d(s)$ and $w(s)$.

$$Y(s) = \bar{T}(s)R(s) + (-\bar{T}(s))w(s) - \bar{T}(s)\bar{d} \tag{2.31}$$

$$U(s) = G_c(s)R(s) + G_c(s)w(s) - G_c(s)\bar{d} \tag{2.32}$$

$$E(s) = (1 - \bar{T}(s))R(s) + (1 - \bar{T}(s))w(s) - (1 - \bar{T}(s))\bar{d} \tag{2.33}$$

Now, having this definition we can define the classical feedback controller $C(s)$ in terms of $G_c(s)$:

$$C(s) = \frac{G_c(s)}{1 - \bar{T}(s)} \tag{2.34}$$

2.3.2 Internal stability and benefits

If we count with a perfect model ($G_p(s) = G_{mp}(s)$), the IMC system is internally stable if and only both $G_p(s)$ and $G_{mp}(s)$ are stable.

Assuming that $G_p(s)$ is stable and $G_p(s) = G_{mp}(s)$. Then the classical feedback system with controller according to equation (2.34) is internally stable if and only if $G_c(s)$ is stable.

The IMC structure thus offers the following benefits with respect to classical feedback:

- No need to solve for roots of the characteristic polynomial $1+G_p(s)C(s)$; one simply examines the poles of $G_c(s)$.
- One can search for $G_c(s)$ instead of $C(s)$ without any loss of generality.

2.4 Linear-quadratic regulator controller (LQR)

The theory of optimal control is concerned with operating a dynamic system at minimum cost. The case where the system dynamics are described by a set of linear differential equations and the cost is described by a quadratic function is called the LQ problem. One of the main results in the theory is that the solution is provided by the linear-quadratic regulator (LQR), a feedback controller whose equations are given below.

The settings of a (regulating) controller governing either a system are found by using a mathematical algorithm that minimizes a cost function with weighting factors supplied by the operator. The cost function is often defined as a sum of the deviations of key measurements, such as desired process pressure or temperature, from their desired values. The algorithm that finds those controller settings that minimize undesired deviations. The magnitude of the control action itself may also be included in the cost function.

The LQR algorithm reduces the amount of work done by the control systems engineer to optimize the controller. However, the engineer still needs to specify the cost function parameters, and compare the results with the specified design goals. Often this means that controller construction will be an iterative process in which the engineer judges the "optimal" controllers produced through simulation and then adjusts the parameters to produce a controller more consistent with design goals.

The LQR algorithm is essentially an automated way of finding an appropriate state-feedback controller. As such, it is not uncommon for control engineers to prefer alternative methods, like full state feedback, also known as pole placement, in which there is a clearer relationship between controller parameters and controller behavior. Difficulty in finding the right weighting factors limits the application of the LQR based controller synthesis.

There exist different sets of equations for the LQR problem depending on the type of system. On this case we are focused con the case of the Finite-horizon, discrete-time LQR.

2.4.1 Finite-horizon, discrete-time LQR

When having a discrete system described by:

$$x_{k+1} = Ax_k + Bu_k$$

with a performance index defined as:

$$J = \sum_{k=0}^{N} (x_k^T Q x_k + u_k^T R u_k + 2x_k^T N u_k) \tag{2.35}$$

the optimal control sequence minimizing the performance index is given by:

$$u_k = -K_k x_k \tag{2.36}$$

where

$$K_k = (R + B^T S_k B)^{-1}(B^T S_k A + N^T) \tag{2.37}$$

and S_k is found iteratively backwards in time by the dynamic Riccati equation:

$$S_{k-1} = A^T S_k A - (A^T S_k B + N)(R + B^T S_k B)^{-1}(B^T S_k A + N^T) + Q \tag{2.38}$$

from terminal condution $P_N = Q$. Note that u_N is not defined, since x is driven to its final state by $Ax_{N-1} + Bu_{N-1}$.

2.4.2 Limitations

The problem data must satisfy:

- The pair (A,B) is stabilizable.
- $R > 0$ and $Q - NR^{-1}N^T \geq 0$
- $(Q - NR^{-1}N^T, A - BR^{-1}N^T)$ has no unobservable mode on the unit circle.

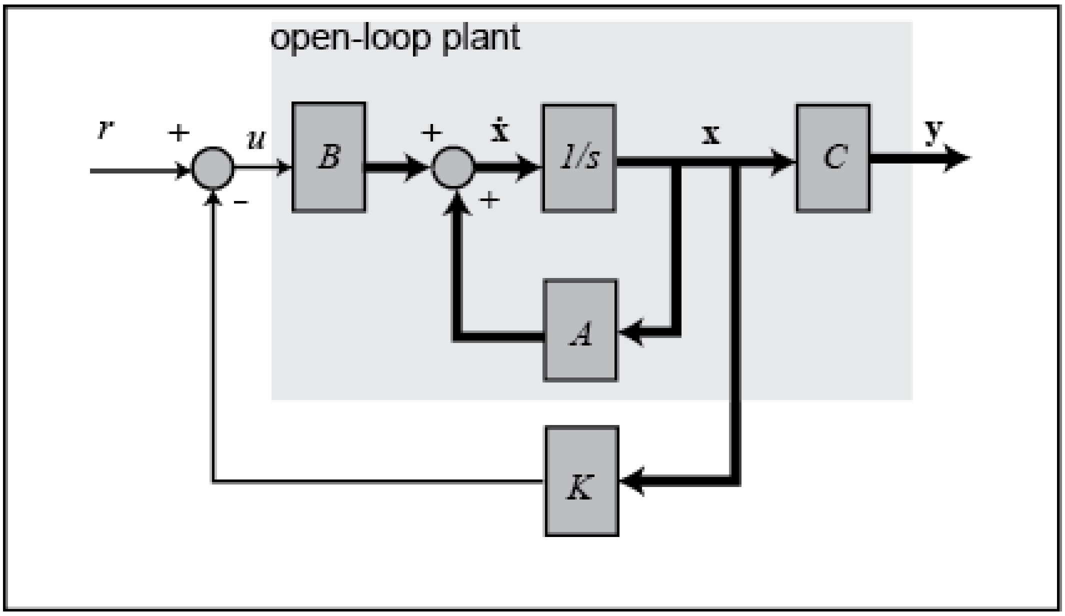


FIGURE 2.5: LQR control scheme

Chapter 3

Methodology

This chapter is devoted to describe the followed steps that allows to fulfill the stated objectives. First the established conditions for the model are discussed and the modeling of the system is made using two different techniques (transfer function and state space) in the identification toolbox of MATLAB. Once the models was generated the controller design was applied. In the case of the transfer function model a IMC strategy was followed. And in the case of the state space model a LQR strategy have been chosen.

3.1 Encrypted Model

The considered boiler model is a MIMO system with three dynamic inputs that can be manipulated from 0% to 100% in order to modify the fuel rate (u_2), air rate (u_3) and flowrate of water (u_4), as it can be seen in Figure 3.1, and one fix input that establish the steam demand rate (u_1) that the plant is requiring. There also exists a limitation in the ratio of the three dynamic inputs that can be modified according to the convenience, describing the common restrictions of the industrial actuators. The boiler model is encrypted and the use of tools for its control is limited according the rules established in the control competition (Comité Español de Automática, 2016).

The input variables changes their reference in several moments observing an strong relation between the steam demand and the rates the other three inputs. The steam demand change during 20 minutes of the simulation time (see Figure 3.2) causing important changes on the references of the fuel ratio, air ratio and water ratio.

It is easy to see in Figure 3.1 how the fuel and air flow are much coupled, knowing that in order to have good combustion in the heating process the burner combine the fuel ratio with the air ratio normally in a relation of 1:10. The water flow variable needs more time to stabilize, even when the reference controlled is evaluated.

There are three output defined as the drum pressure (y_1), the oxygen excess (y_2) and the water level (y_3). This outputs are controlled (see Figure 3.3) to obtain as resultant the correct quantity of steam demanded.

The water level seems hard to stabilize because involves some delays, but it is the less important in order to control. The drum pressure and oxygen excess are noted to change during the first minutes of the simulation, being this physically related to the change of state of the water during the heating process. The generated heat is transferred to water to make steam, the process of boiling. This produces saturated steam at a rate which vary according to the pressure above the boiling water.

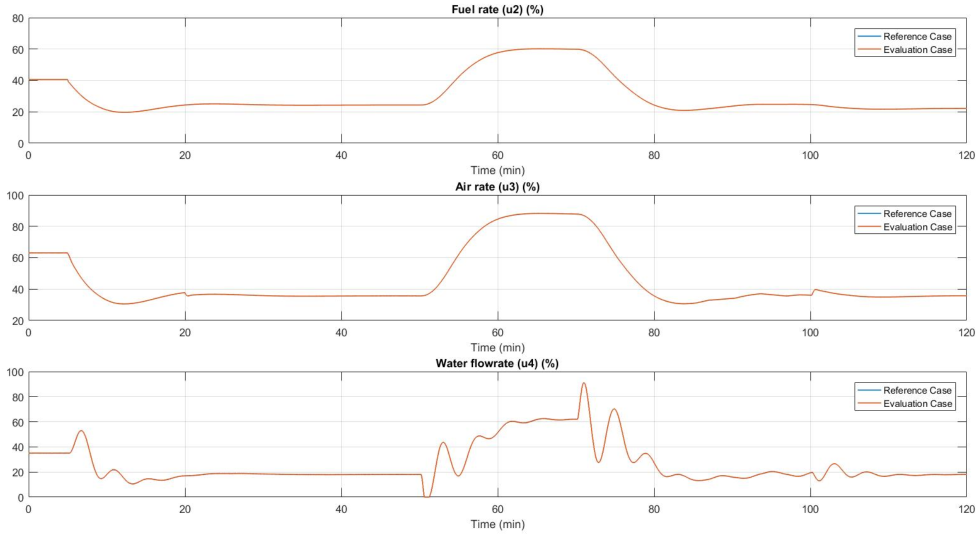


FIGURE 3.1: Dynamic inputs of the encrypted model

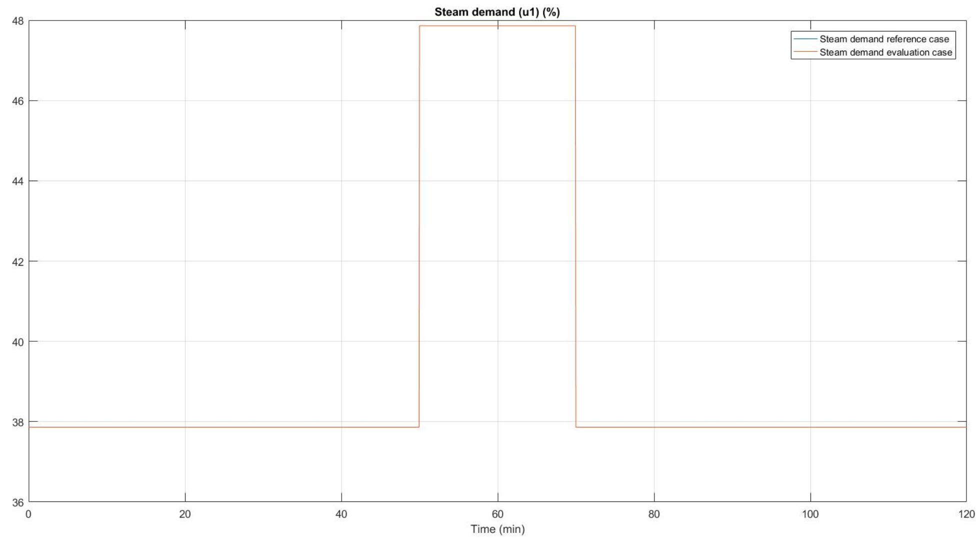


FIGURE 3.2: Fix input Steam demand variation

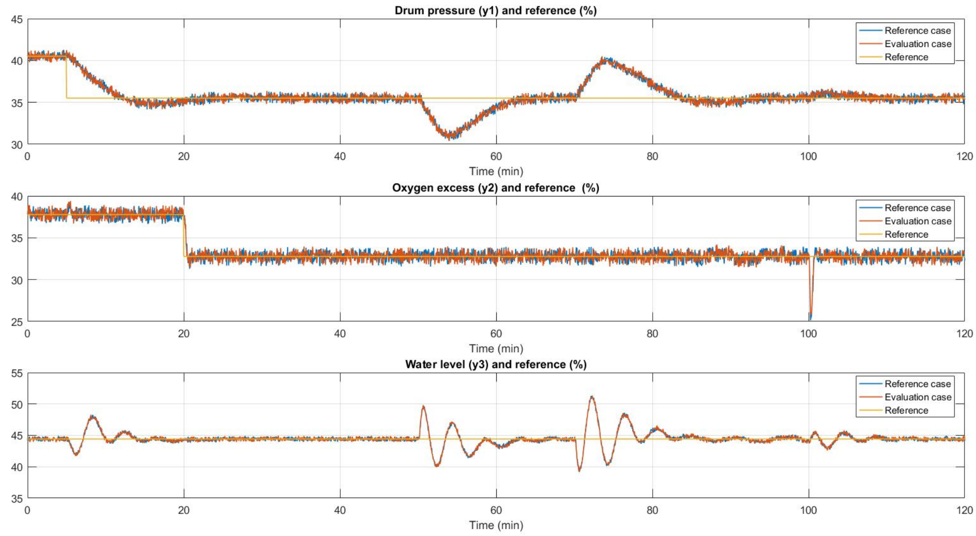


FIGURE 3.3: Controlled outputs of the encrypted model

3.2 Identification of the Plant

The physical equations that describes the system are not used. Instead we use the input/output data obtained from the simulations of the given plant to applicate black box modeling techniques to identify the plant. For this purpose it was necessary to use the identification toolbox available the MATLAB program.

The identification of the plant was defined in two phases: the transfer function identification and the state space identification. Each one of them was applied to different types of controllers. IMC works with the first method and the LQR works with state space models. There are some characteristics inside a plant that makes it unique and to identify it is necessary a detailed analysis of the input/output data.

The model has to describe the behavior of the plant as accurate as possible but also have to be simple to be useful for control design purposes. There exists some references about how control oriented models were constructed for a classic boiler and are used to develop the new models for the considered boiler used as case study (Grosso, 2012).

The boiler plant behaves as a MIMO system that presents a lot of internal interaction, every input affects several outputs.

3.2.1 Transfer Function identification

To obtain a black box model of the plant, the procedure is to take the data from the inputs and outputs of the system and study the dynamics of the states. To define the dynamic, is well known that there are interactions between all the inputs and outputs, so the idea is to define one transfer function for each one of these interactions.

Having 3 inputs and 3 outputs, the system becomes in a 3x3 transfer function matrix. In order to obtain a more accurate response for the model to be identified, it was necessary to divide the matrix into several transfer functions that afterwards shall be concatenated. The parameters were different for each interaction as it can be seen on Table 3.1. The Table 3.2 shows the obtained transfer functions, the Final Prediction Error (FPE) and Mean-Squared Error (MSE) for the estimated models.It is important to note that these two values (FPE and MSE) the closer to zero the difference between the estimator and what is estimated is minimal.

TABLE 3.1: Transfer function models parameters and FIT

Transfer Function Models			
TF	**Poles**	**Zeros**	**FIT**
G_{11}	4	0	62.81
G_{12}	3	1	65.86
G_{13}	3	1	56.55
G_{21}	4	0	58.61
G_{22}	4	0	61.29
G_{23}	3	1	59.07
G_{31}	3	0	52.71
G_{32}	3	0	53.54
G_{33}	3	1	54.33

The transfer functions was constructed until the fit was according to the expectation of these cases over the 60%. To find a fitting of this level in every simulated output it has to be adapted to the needs of each dynamics, that is how the table was constructed were the zeros and poles were chosen and changed until they fit to the behavior of the system. It is notorious that the fit of the outputs related with the third control variable (water level) have a less accurate FIT, but in the dynamic it can be seen how the model at least stills fit with the set point. This is due some delays responses and oscillations that all the inputs have with respect the output. In fact, to understand better what is happening we proceed to estimate the delay matrix between each input/output interaction (delay given in time samples), obtaining as a result the following:

$$n_k = \begin{pmatrix} 0 & 0 & 38 \\ 2 & 0 & 1 \\ 34 & 34 & 23 \end{pmatrix}$$

After obtain all the interactions between inputs and outputs, a final system ($G_{mp}(s)$) is constructed by concatenating all the obtained transfer functions. We proceed to create the transfer function that correlates all the individual interactions into a MIMO matrix.

TABLE 3.2: Resulting transfer function models

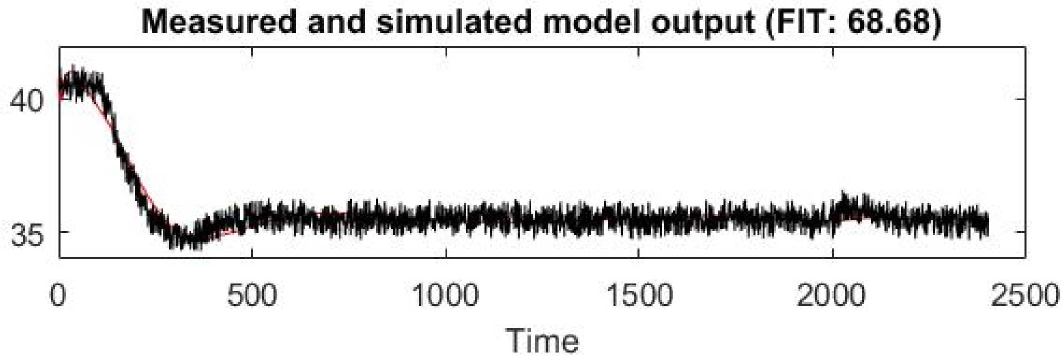
Resulting transfer function Models		
TF	**FPE**	**MSE**
$G_{11} = \frac{2.551e{-}10}{s^4+0.1341s^3+4.509e{-}5s^2+2.352e{-}7s+1.74e{-}10}$	0.2175	0.2158
$G_{12} = \frac{6.277e{-}5s+1.808e{-}7}{s^3+0.06405s^2+0.0003181s+1.815e{-}7}$	0.1831	0.1839
$G_{13} = \frac{8.626e{-}6s+2.731e{-}8}{s^3+0.8619s^2+0.002998s+}$	0.2131	0.2117
$G_{21} = \frac{1.096e{-}10}{s^4+0.01111s^3+4.181e{-}5s^2+1.071e{-}7s+8.133e{-}11}$	0.6703	0.6652
$G_{22} = \frac{1.171e{-}10}{s^4+0.0007584s^3+4.015e{-}5s^2+1.093e{-}7s+1.281e{-}10}$	0.5946	0.5902
$G_{23} = \frac{2.995e{-}6s+1.353e{-}8}{s^3+0.006938s^2+1.311e{-}5s+7.449e{-}9}$	0.6642	0.6598
$G_{31} = \frac{17.757}{(1+44.752s+143.26s^2)*(1+3.507e5s)}$	0.06427	0.06368
$G_{32} = \frac{1.2348}{(1+47.032s)(1+5862.4s)(1+0.82246s)}$	0.06275	0.06217
$G_{33} = \frac{1+210.48s}{(1+2*0.0595*1.1859s+(1.1859s)^2)(1+4065s)}$	0.06014	0.05953

The fittings in general are described in the Table 3.3. The Figure 3.4 shows the obtained fit for each output of the model. Over this model is going to be applied the Internal model controller to the plant.

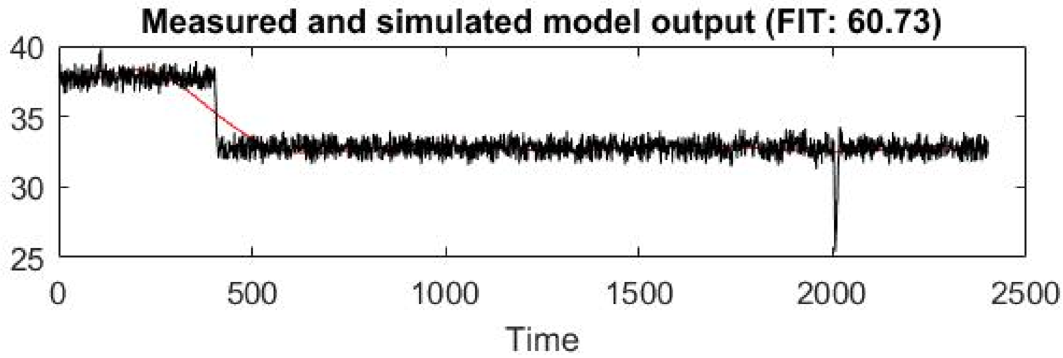
$$G_{mp}(s) = \begin{pmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{pmatrix}$$

TABLE 3.3: Obtained Transfer Function model

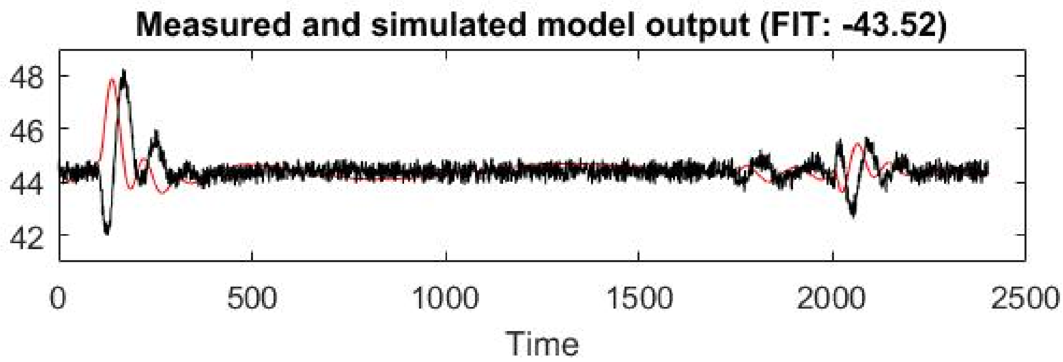
$G_{mp}(s)$	**FIT (y_1)**	**FIT (y_2)**	**FIT (y_3)**
	68,68	60,73	-43,52



(a) Output y_1 with FIT=68.68%



(b) Output y_2 with FIT=60.73%



(c) Output y_3 with FIT=-43.72% (given a notable delay)

FIGURE 3.4: Simulated outputs of the transfer function model

3.2.2 State Space identification

The data obtained from inputs and outputs can be analyzed also to obtain a model in state space configuration, being this type of model more useful to use with the majority of the controllers. The first thing is to define how many outputs is going to have the model to be controlled, because it is not necessary to have the same number of variables than the real plant. Knowing that the steam demand is more alike a disturbance than a measured variable, then the idea was to obtain the model with the variables that change and interact with each other in the control process.

Then the number of outputs is going to be the same but the number of inputs is reduced from four to three in comparison with the original model discussed in Chapter 2. The number of states chosen was three and is related with the number of inputs and outputs that were taken into account, having three inputs (water, fuel and air ratios) and three outputs (pressure of the boiler, water level and oxygen excess).

The parameters are resumed in the Figure 3.5, but in general is a system with three states, with discrete time based on three seconds, without feedtrough or matrix D and the estimation method is the Prediction Error Minimization, so the system model are going to be focused on prediction with estimated initial conditions.

To obtain a trustful model was planted as an objective. The idea was to have only one state space model that describe all the system and then analyze if it needs feedback or not. The obtained discrete model is described by the following equations and matrices:

$$x(k+1) = Ax(k) + Bu(k) + Ke(k)$$

$$y(k) = Cx(k) + e(k)$$

$$A = \begin{pmatrix} & x_1 & x_2 & x_3 & x_4 \\ x_1 & 0.9969 & 0.004051 & -0.002129 & -0.01645 \\ x_2 & 0.009545 & 0.9736 & 0.006316 & 0.0439 \\ x_3 & -0.0168 & 0.01784 & 0.983 & -0.05528 \\ x_4 & -0.02555 & -0.01602 & 0.0327 & 0.3595 \end{pmatrix}$$

$$B = \begin{pmatrix} & u_1 & u_2 & u_3 & u_4 \\ x_1 & 8.158e-5 & -6.203e-5 & 0.0002221 & -0.0002128 \\ x_2 & -0.0003333 & -4.029e-5 & -0.000361 & 0.0004944 \\ x_3 & 0.0004419 & -0.000276 & 0.0007114 & -0.0006746 \\ x_4 & 0.0005295 & -0.002673 & 0.01076 & -0.01027 \end{pmatrix}$$

$$C = \left(\begin{array}{ccccc} & x_1 & x_2 & x_3 & x_4 \\ y_1 & 105.9 & 10.05 & -22.15 & -0.8343 \\ y_2 & 95.47 & 61.56 & 36.32 & -0.8572 \\ y_3 & 105.3 & -19.33 & 9.102 & 7.121 \end{array}\right)$$

$$K = \left(\begin{array}{cccc} & y_1 & y_2 & y_3 \\ x_1 & 0.0006927 & 0.0003269 & 0.002617 \\ x_2 & -9.53e-5 & 0.002949 & -0.006793 \\ x_3 & -0.0005207 & 0.0028 & 0.003629 \\ x_4 & -0.006058 & 0.0004706 & -0.01429 \end{array}\right)$$

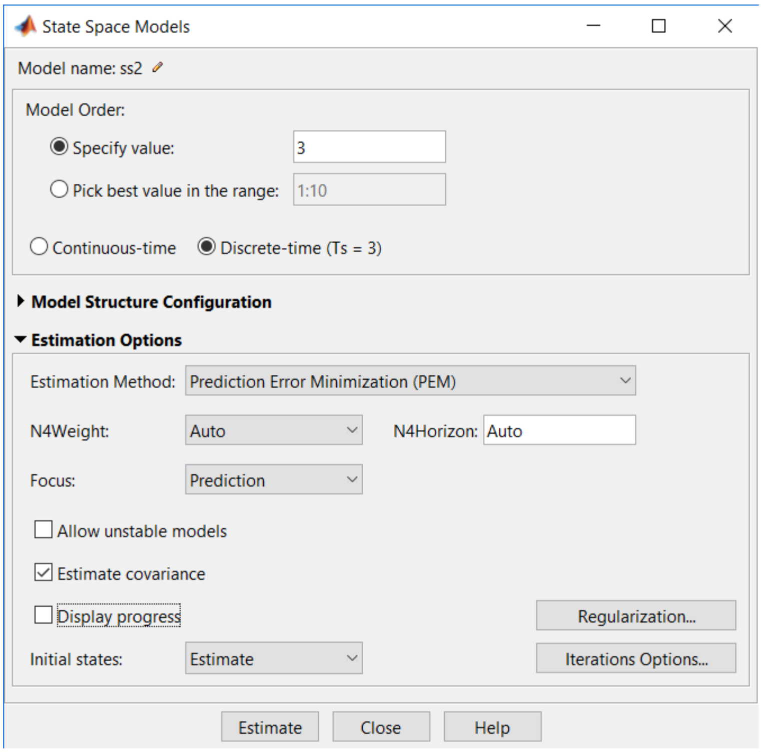
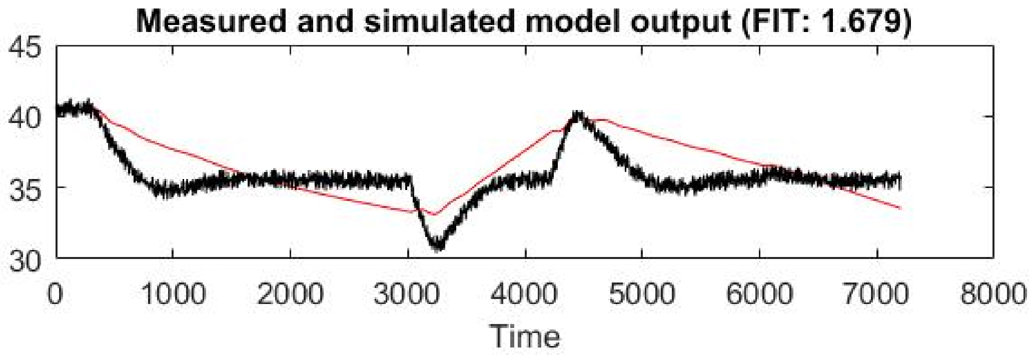


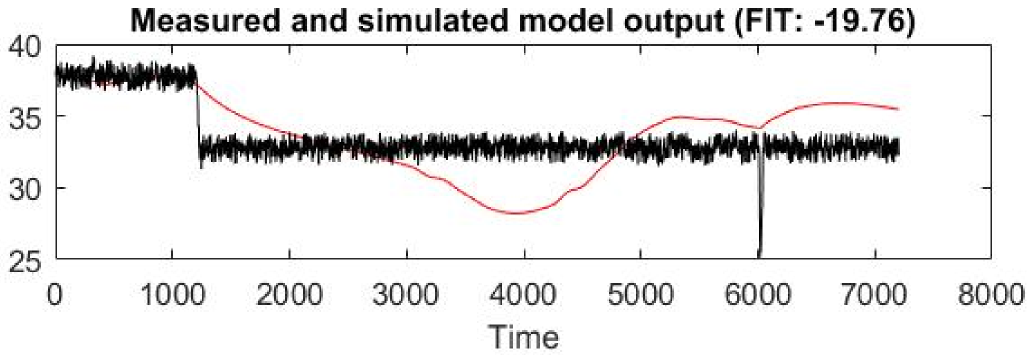
FIGURE 3.5: Parameters of the model in state space

With simulated output, the model is not so accurate compared with the outputs of the system because the fit value is around -300%, so the model could not be validated correctly because it does not satisfy the fitting expectations. In the Figure 3.6 this relation can be observed.

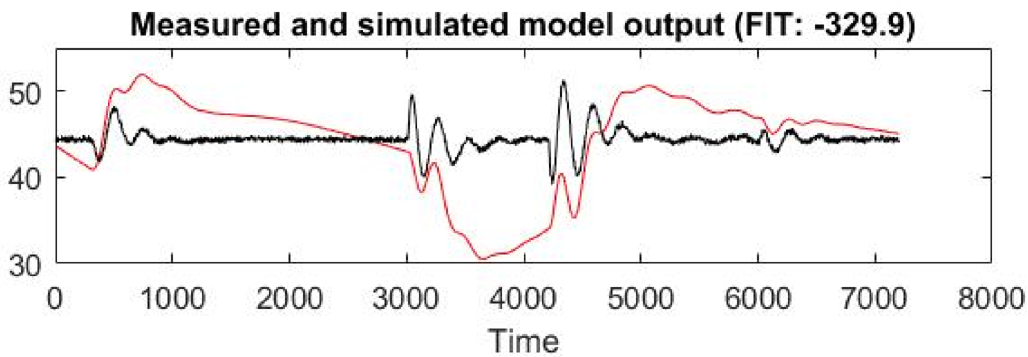
The next step is to simulate how the model behaves with a close loop. When the horizon of the close loop is shorter it means that the feedback is happening in a shorter time, and the deviations can be reduced in every step. So, taking into account that the system can have a feedback with every step, the simulation of the model to be compared with the system is as it can be seen on the Figure 3.7.



(a) Simulated output y_1 with FIT=1.679%



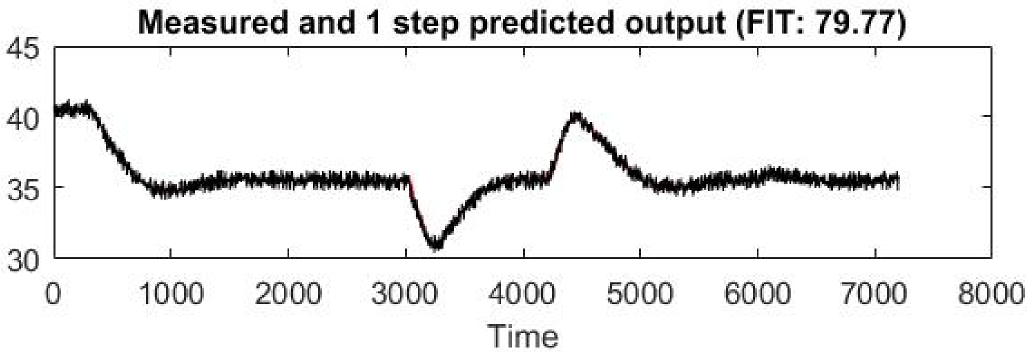
(b) Simulated output y_2 with FIT=-19.76%



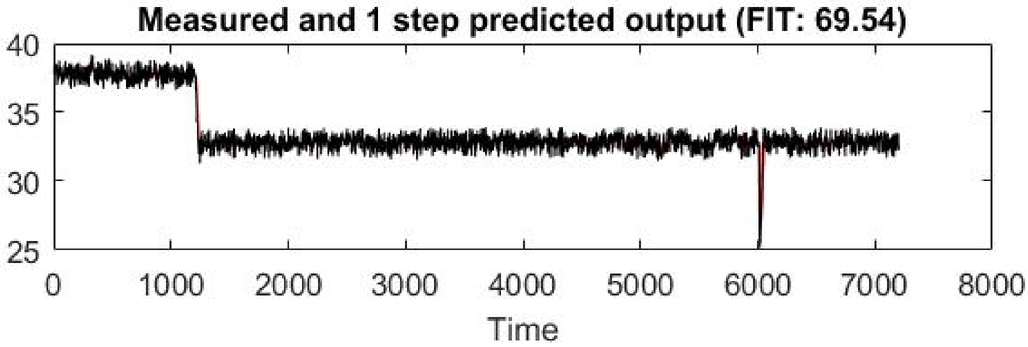
(c) Simulated output y_3 with FIT=-329.9%

FIGURE 3.6: Simulated outputs of the state space model

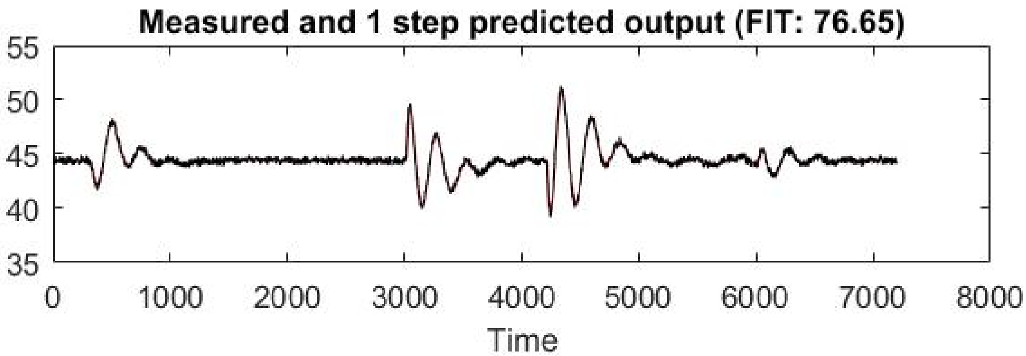
The model in state space with the variation of the important variable setpoints (steam demand, oxygen excess and drum pressure) with close loop behaves as expected. The fitting was over the 70% and can be validated to be used as the model to be implemented in the LQR controller as is going to be explained in the next section.



(a) Simulated output y_1 with FIT=79.77%



(b) Simulated output y_2 with FIT=69.54%



(c) Simulated output y_3 with FIT=76.65%

FIGURE 3.7: One step horizon of the state space model

3.3 IMC controller implementation

Once the plant was identified as a transfer function model, we proceed to implement the IMC control strategy. Based on the theory explained on the Chapter 2 we know from the Figure 2.4 that the forward IMC implementation are based basically on introduce the same inputs ($u(s)$) of the plant (G_p) to the identified model (G_{mp}). Both outputs ($y(s)$) are compared and the difference ($d(s)$) between them are the feedback that compares its value with the references inputs ($R(s)$) given the error input ($E(s)$) to the controller ($G_c(s)$) that creates a corrected output.

Based on the equation (2.34), we design the IMC controller in a Matlab script fixing the $G_c(s)$ parameter as the parameter that will give the tuning capability to the controller. Once the controller was found we proceed to implement the same structure noted in the Figure 2.4 in the available plant model on Simulink. This implementation can be observed in the Figure 3.8. During the implementation the only thing to take into account was the fact that the plant is a MIMO plant with 4 inputs and 3 outputs. Nevertheless, for identification purposes the obtained transfer function is a 3 inputs by 3 outputs MIMO model. The missing input corresponds to the steam demand that could be denoted as a measured disturbance. So, given the fact that our controller is only prepared to give 3 outputs, corresponding to the control signal for the fuel rate, air rate and water rate respectively, the steam demand is added as an additional input to the plant, affecting directly the behavior of the system outputs, but controllable given the fact that is a known value.

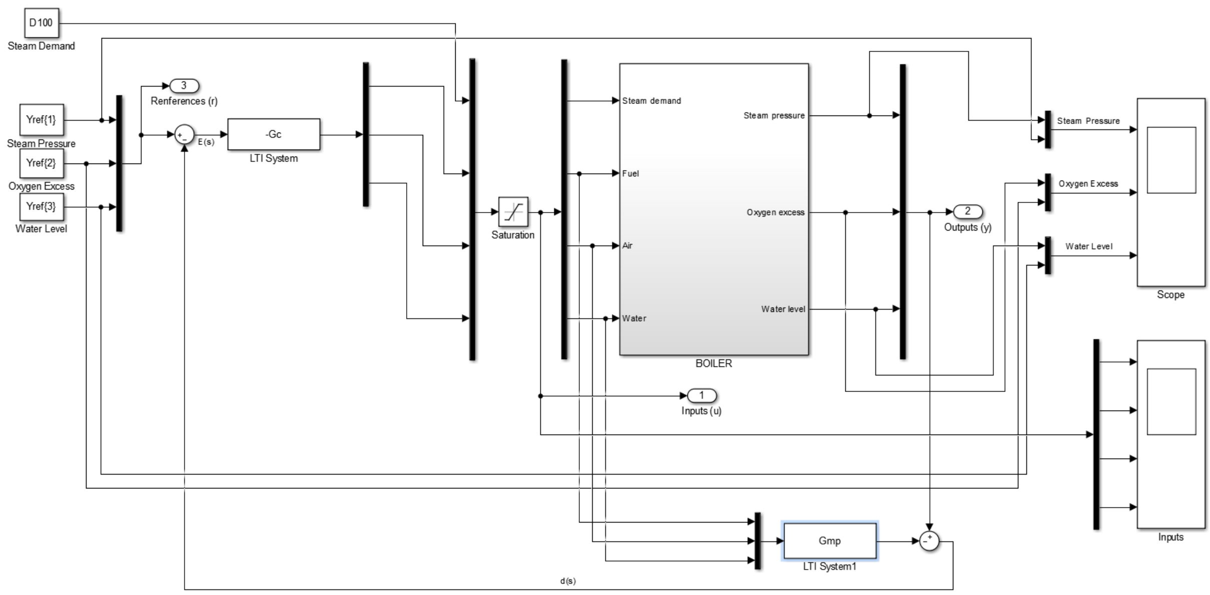


FIGURE 3.8: IMC implementation on Simulink

3.4 LQR controller implementation

Now in the case of the LQR controller implementation, we base the design of the controller on the formulas described in the point 2.4.1 for a finite-horizon discrete-time LQR. Having into account that the evaluation data have a time sampling of $ts = 3$ the identified model in state space is discrete and counts with a very good FIT (over 70%). This allow us to apply the cost function reduction using the Ricatti equation (2.38). The controller its designed according to the equation (2.37) and the close loop to the system takes the form of the equation (2.36), as it can be observed on the Figure 2.5.

The design of the controller is applied in a Matlab script using the instruction **dlqr**, that needs to have as inputs the A and B matrices of the state space model and the Q, R and N matrices of the LQR equations. Knowing that there is no feedforward on this system, the matrix N is set to zero directly. In the case of the parameters Q and R can be used as design parameters to penalize the state variables and the control signals. The larger these values are, the more these signals are penalized. Basically, choosing a large value for R means we are trying to stabilize the system with less (weighted) energy. On the other hand, choosing a small value for R means that the penalization of the control signal is not so important. Similarly, choosing a large value for Q means that we are trying to stabilize the system with the least possible changes in the states and small Q implies less concern about the changes in the states.

Since there is a trade-off between the two, in this case we iterates the values of Q and R but in opposite manner until finding a value of K that stabilizes the system. Performing several test to see how the closed loop poles were affected and given the constant changes of the physical states of the system it was decided to give less weight to the control actions R and try to stabilize the system making more important the weights of the Q matrix.

The Matlab instruction **dlqr** returns as outputs the controller K_k, the infinite horizon solution S_k and the closed-loop eigen values according to $e = \mathbf{eig}(A - B * K)$. As its known, we are only interested on found a controller K_K that in closed loop makes the system stable. This is only possible if the closed loop eigen values (e) are positioned in the left-half plane of the root locus, meaning that the poles are pushing the system to stability. So, in order to find this values we alter iteratively the values of R and Q until find the K_k that makes the system stable in close-loop. The used Matlab code was:

```
%% Calculates LQR by iterations
clear all; clc; close all;
% Load the model
load('modelo_bueno3x3x3.mat')
% Check controllability of the loaded model
co = ctrb(ss3);
controllability = rank(co);
% Initialize the arrays
K_all = [];
R_all = [];
Q_all = [];
% Calculates all the possible controllers that makes
% the system stable
```

```
if controllability == 3
    for i = 1e-3:1e-5:1;
        for j = 1:200;
            Q = diag([j,j,j]);
            R = diag([i,i,i]);
            % Calculate the controller K, and the poles Ek
            [K,S,Ek] = dlqr(ss3.A,ss3.B,Q,R);
            % Store all the K that makes the closed
            % loop system poles stables
            if Ek(1) < 0 && Ek(2) < 0 && Ek(3) < 0
                K_all = [K_all, K];
                R_all = [R_all, diag(R)];
                Q_all = [Q_all, diag(Q)];
                Ek_all = [Ek_all, Ek];
            end
        end
    end
end
%% Check for the minimum poles and find position in the arrays
[M,I] = min(Ek_all(:));
[I_row, I_col] = ind2sub(size(Ek_all),I);
%% Store better results to evaluate in simulation
K_best = [K_all(:,I_col*3-2), K_all(:,I_col*3-1), K_all(:,I_col*3)];
save('K_to_evaluate', 'K_best');
```

This code returns six different controllers for the following values of R and Q.

$$R = \begin{pmatrix} 1.10^{-3} & 0 & 0 \\ 0 & 1.10^{-3} & 0 \\ 0 & 0 & 1.10^{-3} \end{pmatrix}$$

$$Q_1 = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} Q_2 = \begin{pmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{pmatrix} Q_3 = \begin{pmatrix} 11 & 0 & 0 \\ 0 & 11 & 0 \\ 0 & 0 & 11 \end{pmatrix}$$

$$Q_4 = \begin{pmatrix} 13 & 0 & 0 \\ 0 & 13 & 0 \\ 0 & 0 & 13 \end{pmatrix} Q_5 = \begin{pmatrix} 19 & 0 & 0 \\ 0 & 19 & 0 \\ 0 & 0 & 19 \end{pmatrix} Q_6 = \begin{pmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{pmatrix}$$

where the obtained controllers gave as result the following values:

$$K_1 = \begin{pmatrix} 6.3210^{+05} & 4.4510^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4410^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix} K_2 = \begin{pmatrix} 6.3210^{+05} & 4.4410^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4310^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix}$$

$$K_3 = \begin{pmatrix} 6.3210^{+05} & 4.4410^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4310^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix} K_4 = \begin{pmatrix} 6.3210^{+05} & 4.4410^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4310^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix}$$

$$K_5 = \begin{pmatrix} 6.3210^{+05} & 4.4410^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4310^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix} K_6 = \begin{pmatrix} 6.3210^{+05} & 4.4410^{+03} & -4.2210^{+04} \\ 9.7310^{+05} & 1.4310^{+04} & -5.7510^{+04} \\ 8.3210^{+05} & 1.0210^{+04} & -4.3110^{+04} \end{pmatrix}$$

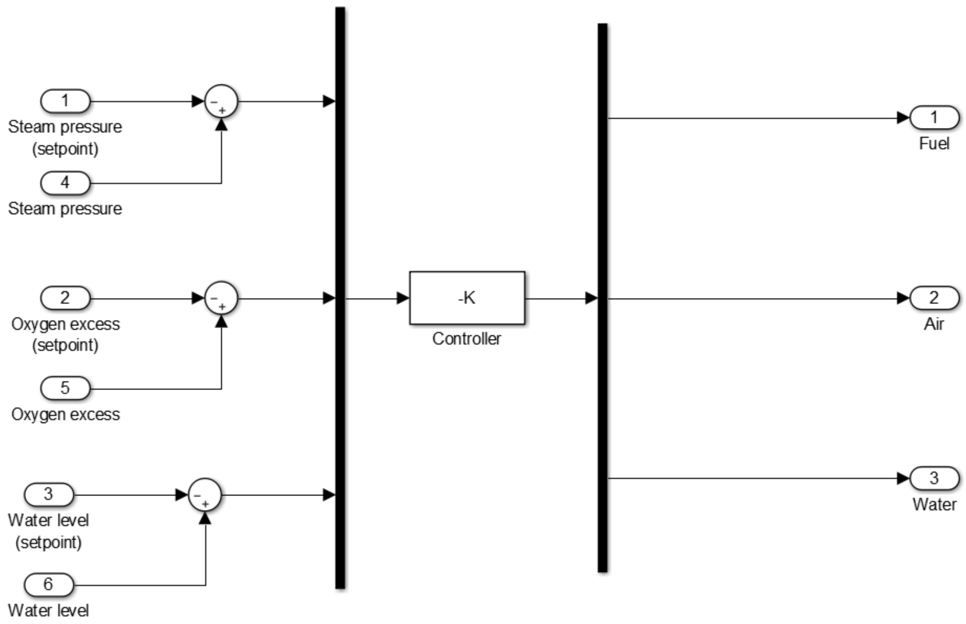
$$Ek_1 = \begin{pmatrix} -1.4910^{-11} \\ -4.7610^{-17} \\ -1.5610^{-15} \end{pmatrix} Ek_2 = \begin{pmatrix} -3.8910^{-13} \\ -3.9110^{-16} \\ -1.7210^{-15} \end{pmatrix} Ek_3 = \begin{pmatrix} -1.6110^{-12} \\ -1.1610^{-15} \\ -6.8210^{-18} \end{pmatrix}$$

$$Ek_4 = \begin{pmatrix} -5.6110^{-12} \\ -1.8710^{-16} \\ -1.8710^{-16} \end{pmatrix} Ek_5 = \begin{pmatrix} -2.0310^{-12} \\ -2.9410^{-15} \\ -8.4310^{-17} \end{pmatrix} Ek_6 = \begin{pmatrix} -7.7110^{-13} \\ -8.5910^{-16} \\ -1.4610^{-15} \end{pmatrix}$$

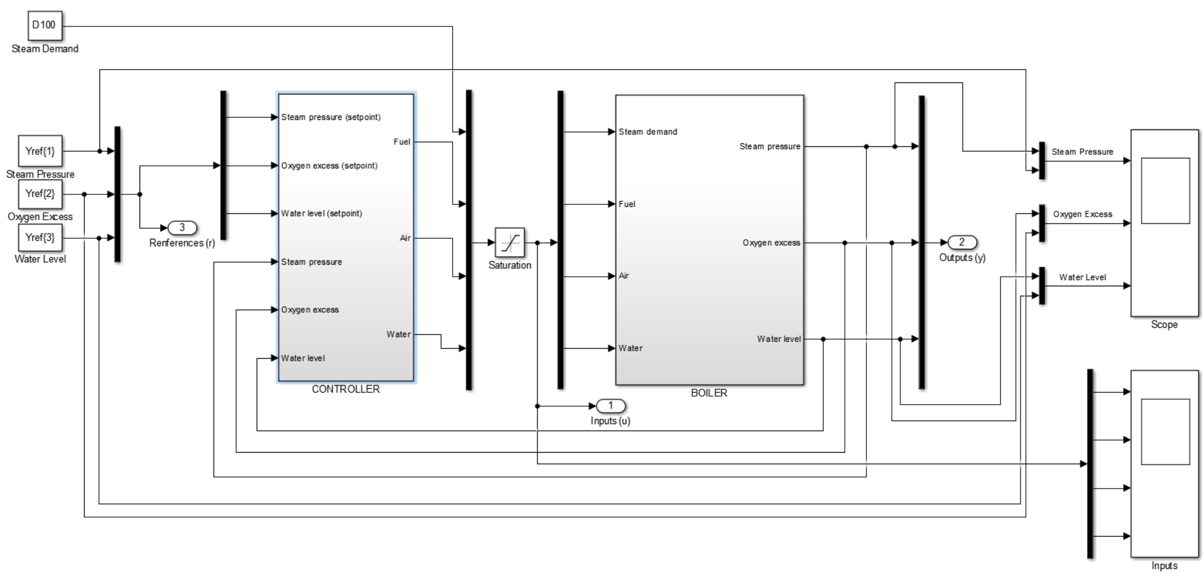
where K is the controller matrix, Ek are the poles of the closed loop system.

Once the controller have been found, what remains is to implement it in the plant using the Simulink model available. We modify this model in order to replace the existing controller with our K_k. As it can be seen in the Figure 3.9a, inside the controller block we replace the existing block for our K_k and also introduce to the controller the differences between the references inputs (r) and the feedback outputs (y) of the plant.

The controller is a 3 inputs by 3 outputs matrix, meanwhile the plant is a MIMO system with 4 inputs and 3 outputs. As in the IMC case, the first input to the plant (the steam demand) is a measurable disturbance that affects to all the dynamic of the system. However, this perturbation can be managed by only the control inputs of the plant, that are the fuel rate, the air rate and the water rate. So, as it can be observed in the Figure 3.9b, the steam demand is introduced directly to the plant.



(a) Block of LQR controller



(b) LQR applied to the plant

FIGURE 3.9: LQR implementation on Simulink

Chapter 4

Results

This chapter is devoted to explain the obtained results when the two different controllers were applied to the system using the methods described in the Chapter 3. A comparison between the identification methods and the multivariable controllers is performed.

4.1 IMC implementation results

During the implementation of the IMC controller several problems were encountered. First of all, the identified transfer function model is very complex. As it have been explained in the Section 3.3.1, the easiest form to identify the MIMO transfer function was to work with a battery of different transfer functions that relates the inputs and outputs individually. Each obtained model have unique characteristics and different polynomial orders. Given this differences, the number of poles of the resulting general transfer function are quite high.

The IMC controller is designed as a transfer function and to be able to control effectively a high order system the pole placement shall be performed differently for each interaction. There is no direct connection between each input and output of the plant. Being a MIMO system there exist a strong link between the majority of inputs/outputs and the IMC controller for this particular case does not count with easy pole placement capability.

When trying to apply this method inside the real plant SIMULINK implementation, any solver could reach results until the final time step. Given the complexity of the model, the data delay, measured and unmeasured disturbances and uncertainties, the resulting IMC controller is also very complex. In a finite time simulation of the system with the IMC controller implemented, the SIMULINK program reproduce errors given by singularities encountered on certain steps.

However, in order to prove that a simpler transfer function controller could be implemented inside the real plant simulation using the IMC model, we simulate a controller with 3 poles placed in -1. The obtained results can be observed in the Figure 4.1.

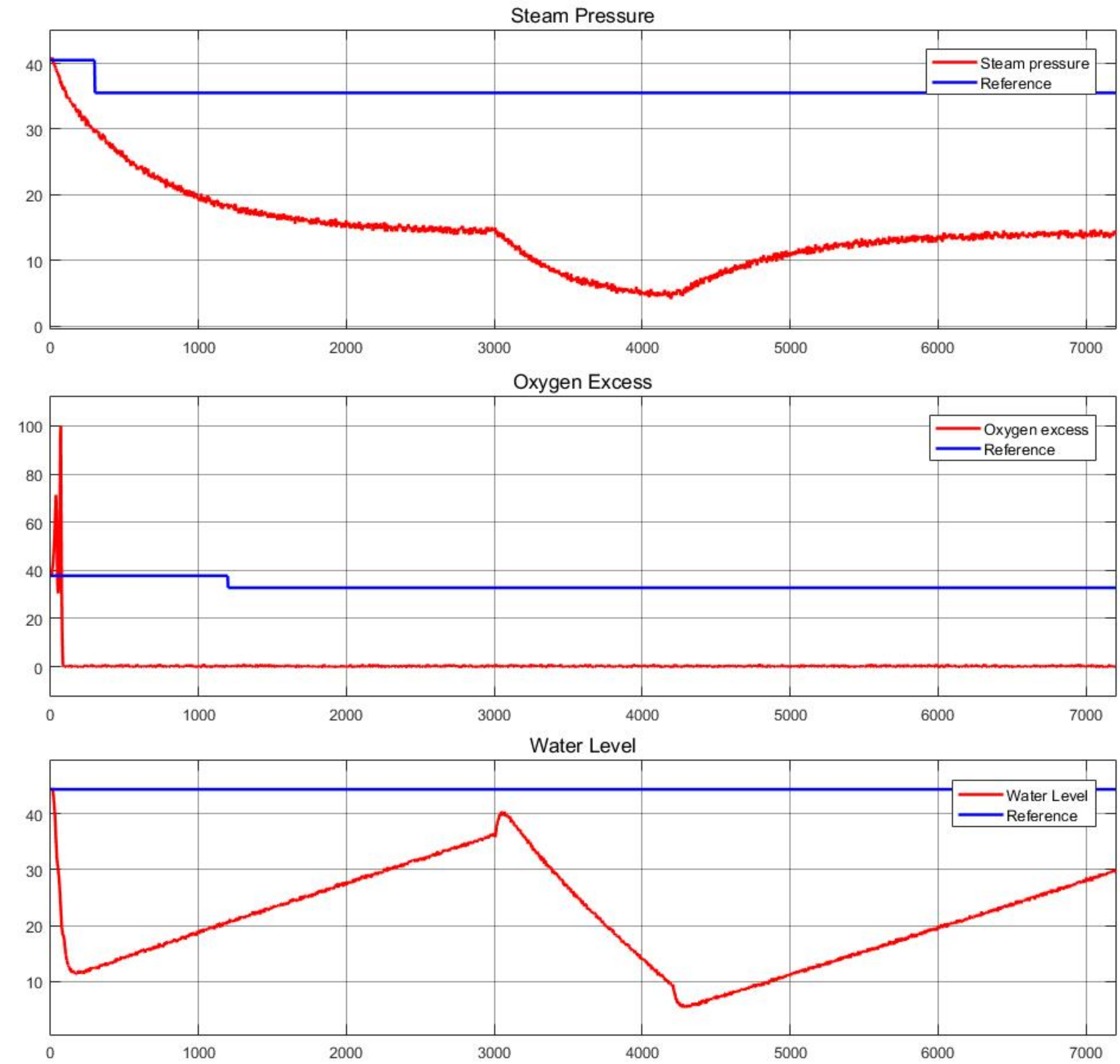


FIGURE 4.1: IMC results with a simple $G_c(s)$

As it can be seen, the IMC strategy is feasible only when the controller complexity is low. For this particular case, given the complexity of the identified transfer function model, the obtained IMC controller is also very complex and not easily tunable. Moreover, this type of controller have made the SIMULINK simulation impossible given the singularities found on certain steps no mattering which solver has been used.

4.2 LQR implementation results

Once the IMC implementation have failed we pass to try a different controller approach. Seeing that the state space model was easily identified and have a very good FIT to the input/output data of the plant simulation, as is explained in the point 3.3.2, an

LQR controller was applied.

The first step was to verify that the obtained controller stabilize the model. Using the **dlrq** command in Matlab, as it was described in the Section 3.5, the resulting controllers (K_1 to K_6) also is accompanied for the eigen values vectors (Ek_1 to Ek_6) of the close loop system. By varying the values of the R and Q matrices, that weights the control actions of the controller and the variations in the states, we iteratively calculate different K until the close loop eigen values are negative, meaning that the model is being controlled till the stability. In the Figure 4.2 is observed the step response of the state space model when this obtained K makes the model stable in close loop.

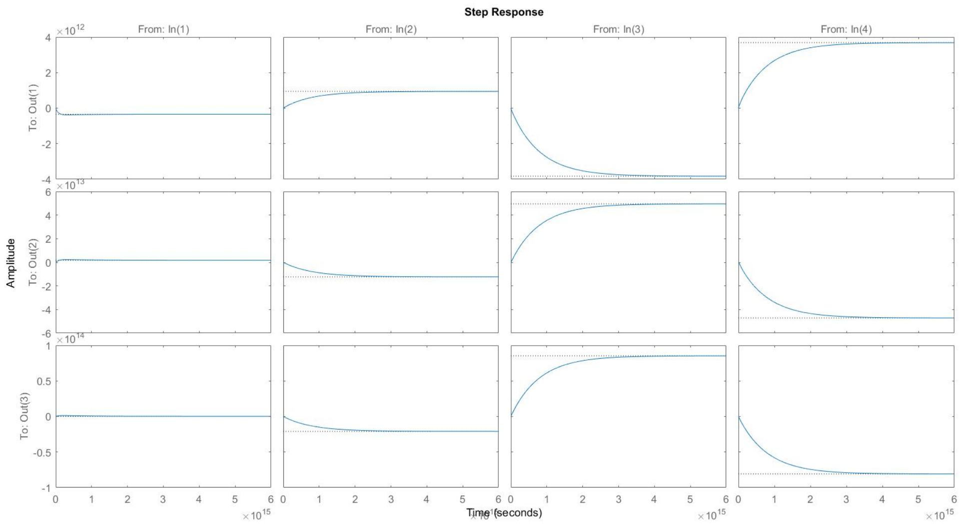


FIGURE 4.2: SS model controlled with the LQR controller

Now to begin to understand how the plant behaves first in open loop and be able to keep a good tracking of the application of the actual controller, we simulate in SIMULINK the response of the plant with a null control action, that is the same as establish the Q and R values of the controller as the identity matrix. The obtained results can be observed in the Figure 4.3, and as it can be noted, the response of the plant in open loop is approximately as expected due to the fact that almost all the physical interactions match.

For instance, in the case of no control action, for a variable steam demand that acts as measurable perturbation, the water level increase till 100% meanwhile the oxygen excess decrease until 0%. This actions are completely justifiable knowing that in the actual steam boiler tank it could not be oxygen when is fully filled with water. However, in the case of the steam pressure we observe a little perturbation. For a fix steam demand the pressure inside the boiler stabilize in a value that could correspond to the same pressure that can be encountered in the water inlet. This is due to the fact that the water fill completely the tank and pressures equilibrates. In the interval when the steam demand change the pressure drops, and this is also reasonable knowing that probably the output valve that normally lets out steam opens a bit more causing differences between the internal pressure and the output pressure that also equilibrate. This results allow us to validate the plant model given by the CEA.

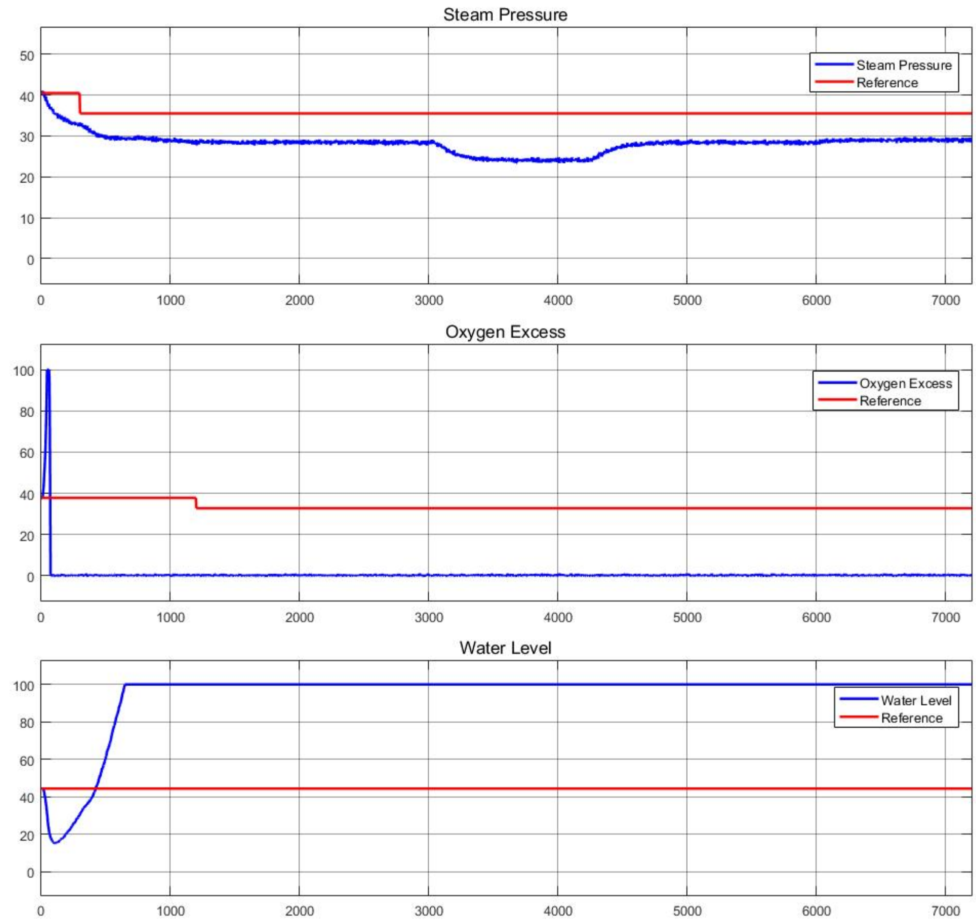


FIGURE 4.3: Open loop response of the plant

Continuing with the application of the LQR controller, we fix the values of R and Q to those values that makes the model stable. Based on the values of the closed loop poles for each obtained K (as is shown in Section 3.4) we pass to choose the controller K that place the poles in the minimum value on the left half plane of the root locus. In this case the one that it was selected is the K_6 that was obtained for $R = diag(1e{-3})$ and $Q = diag(20)$.

The small value of R is related with less control action penalization, meanwhile the large value of Q denotes more penalization to the states variation.

To apply it to the plant model that have been already validated we pass to simulate different scenarios to see which is the plant behavior comparing the outputs (y) against the controlled inputs (u).

- **Scenario 1: Fix steam demand, control action no penalized and states variation penalized**

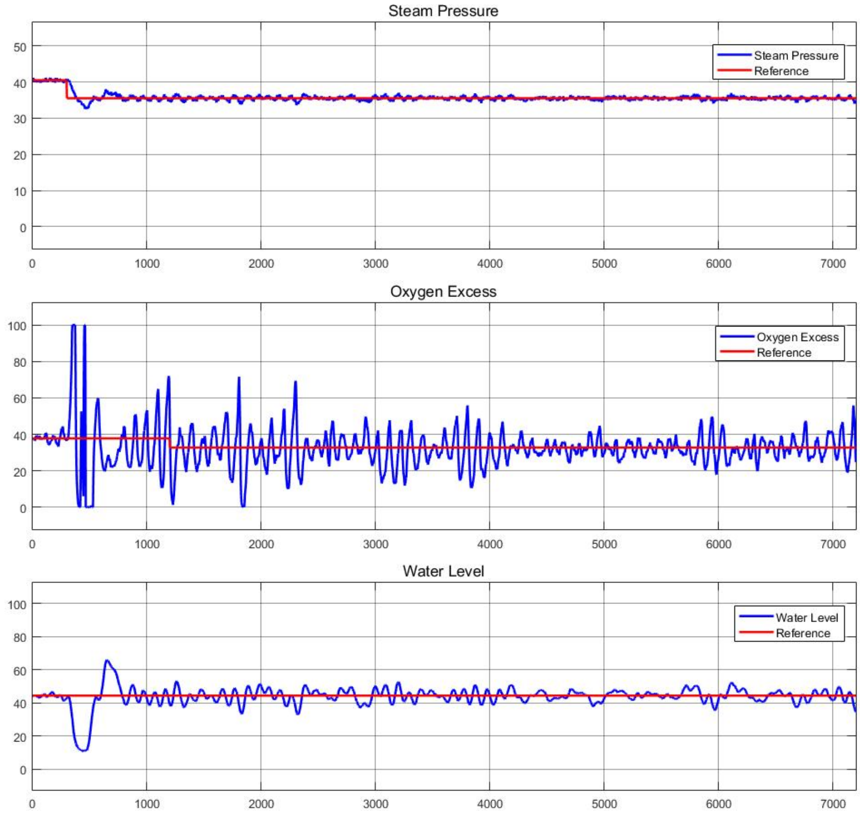
  The first scenario was to test the response of the plant being controlled for a stable controller and fixing the steam demand to avoid abrupt changes for the measurable perturbation in the complete time. The obtained results are shown in the Figure 4.4. As is observed the control of the plant is very accurate for each one of the three variables. The references signals was followed and a remarkable steady state is reached at least for two of the three outputs. This was one of the best obtained results and prove that the LQR controller is very effective for this plant when the steam demand is fixed. However the action of the controller is observed to be very aggressive, which lead us to the second scenario.

- **Scenario 2: Fix steam demand, control action penalized and states variation penalized**

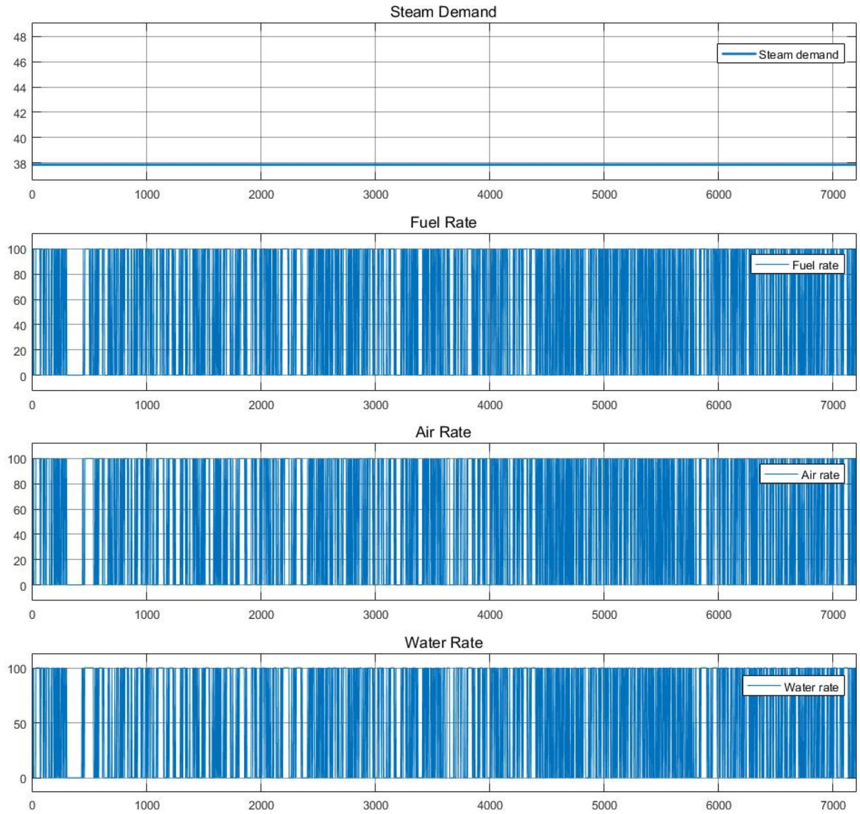
  Observing that the input signals to the plant that comes from the controller (control actions) are very aggressive and in most parts fully saturated between the 0 and 100% values, we try to make it smoother in order to see how the system will behave. There exist two possibilities, being the first one to penalize the control actions using a large value in R. However by making different tests the model does not stabilize when the control action is being penalized, probably because the important link between all the inputs/outputs that makes the controller to be as fast as possible, hence to increase the value of R is discarded.

  The second possibility is to penalize directly inside the simulation the output of the controller using a rate limiter that restrict higher variations of the control actions between fixed values of rising slew rate and falling slew rate. This actually makes the output of the controller smother and also, when the value of the steam demand is fixed, makes the behavior of the outputs to stabilize even more than in the previous case. The obtained results, for a case when the rising and falling rates equal to 0.1, can be observed in the Figure 4.5.

  Its important to denote that even when the response of the system is very good when it reach the steady state (robust response), this penalization to the control action introduce a delay to the outputs making impossible to stabilize the values of the steam pressure and the oxygen excess according to the first reference values. The cost of this type of control its that the plant takes more time to stabilize having an overdamped response. Hence, the classical trade-off between robustness and performance is observed.

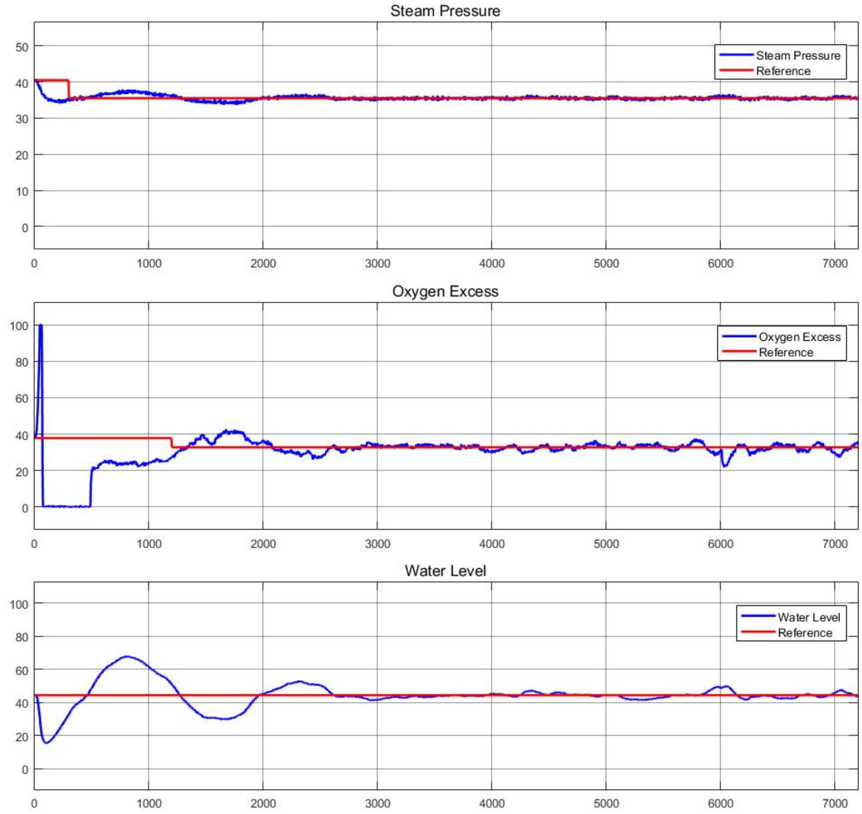


(a) Outputs

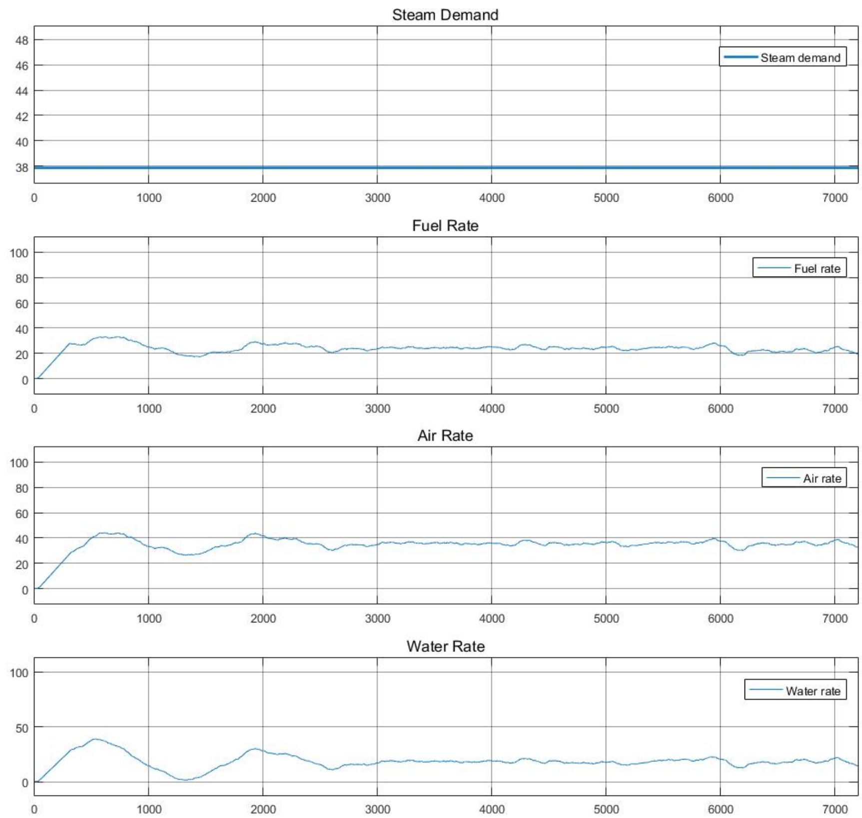


(b) Inputs

FIGURE 4.4: Scenario 1: Close loop LQR controller without steam demand variation, $R = diag(1e-3)$ and $Q = diag(20)$



(a) Outputs



(b) Inputs

FIGURE 4.5: Scenario 2: Close loop LQR controller without steam demand variation, $R = diag(1e-3)$, $Q = diag(20)$ and $Rate = 0.1$

- **Scenario 3: Variable steam demand, control action no penalized and states variation penalized**

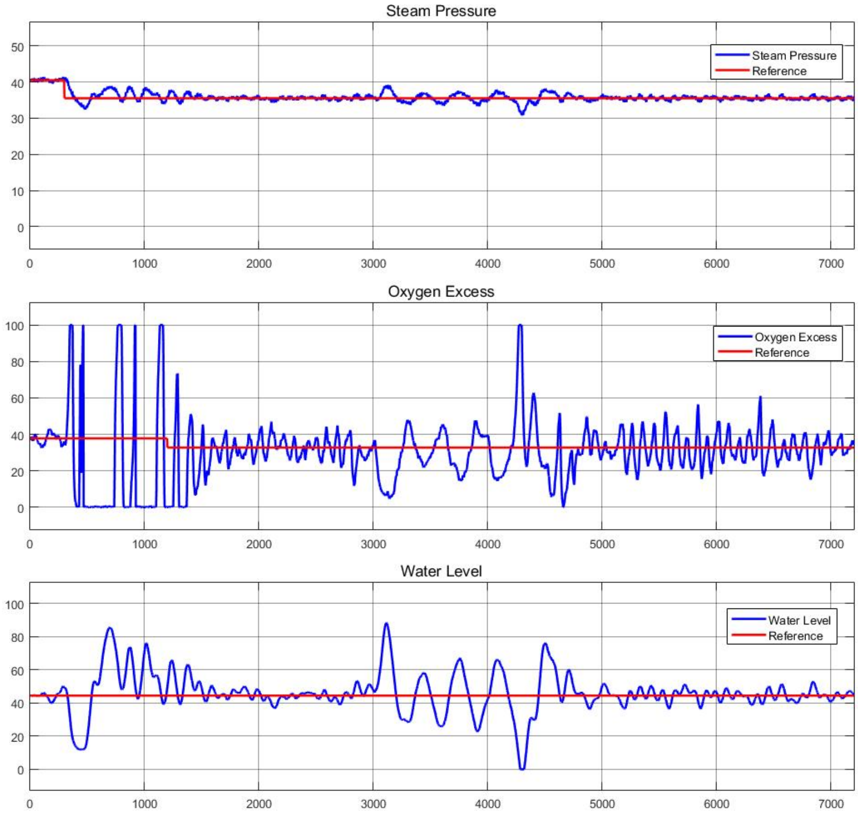
  The third scenario consist now in testing the response of the plant when the steam demand vary. For this first case we want to see how the system reacts only by applying the controller that stabilize the model, not penalizing the control action. The Figure 4.6 shows the obtained results, being notable that the steam pressure is always easily controlled. However, despite the fact that the control action is pretty aggressive (as it can be seen in the Figure 4.6b), the oxygen excess and the water level have some issues to stabilize in the beginning of the simulation. This could be provoked by the beginning values of the controller. The output of the controller always begins in zero meanwhile the references for the outputs of the system have a predetermined value different from zero. Also the initial conditions are set to values that differs the outputs of the controller (u) and in order to stabilize the outputs (y) the control actions have aggressive reactions switching between 0 to 100% with a high frequency.

  Even do, the response of the system when the demand vary in the time 3000 is actually pretty good. The rapid control action makes the system stabilize quickly and the steady state is reached again once the steam demand returns to its initial value. This behavior could be said that is practical in terms of long time of use of the steam boiler, which is the standard almost in every industrial plant. However, it could have some problems when the required working time of the system is lower. The best method to guarantee a rapid stable response could be to be able to introduce an initial value to the output of the controller that match with the necessary initial conditions of the plant.

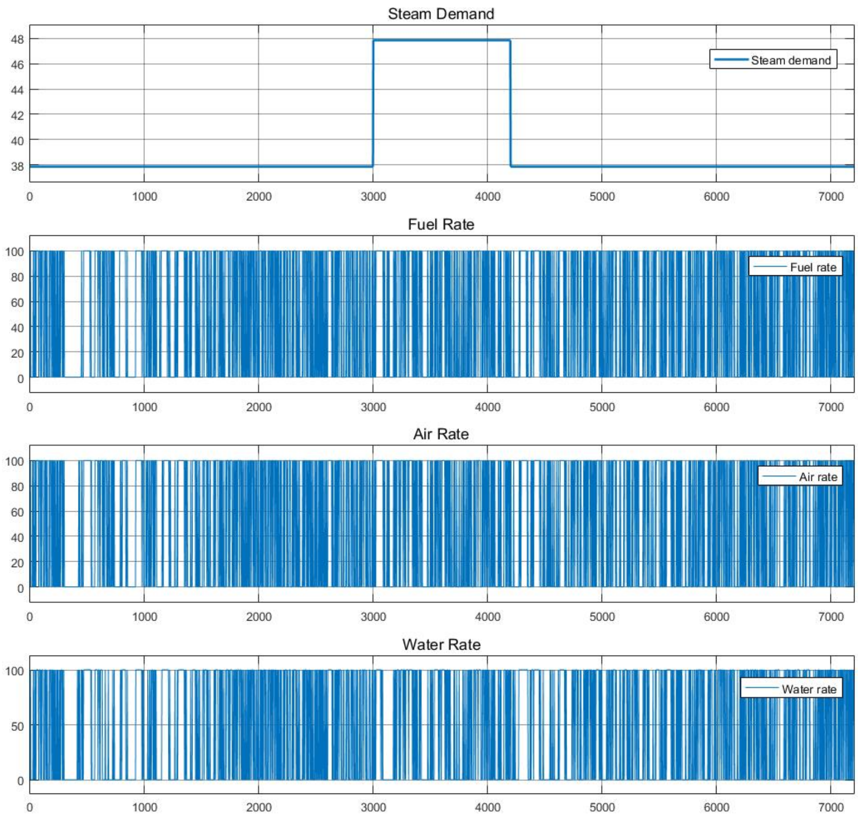
- **Scenario 4: Fix steam demand, control action no penalized and states variation penalized**

  The fourth and last scenario was to see if penalizing the control actions, as in the scenario 2, the system have a better performance or if its more robust. We rate the output of the controller with a rising and falling rates equal to 0.5 obtaining the results shown in the Figure 4.7. The same rate value used for the scenario 2 does not have a good response given the fact that if the control actions are penalized in mayor scale, the perturbations inside the plant and the steam variation makes the response of the system uncontrollable. There must be a trade-off between the rapid response of the controller and its smoothness.

  As its observed in the Figure 4.7 the response of the system improves a little bit more. There is a good performance when the steam variation occurs and the steady state is reached once this value change again.

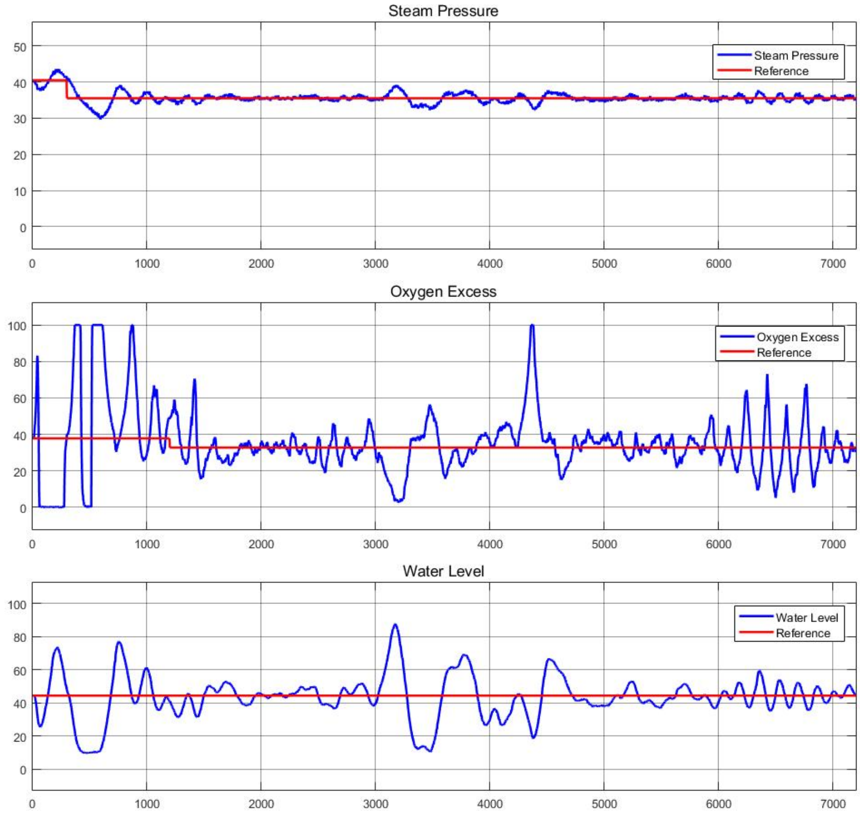


(a) Outputs

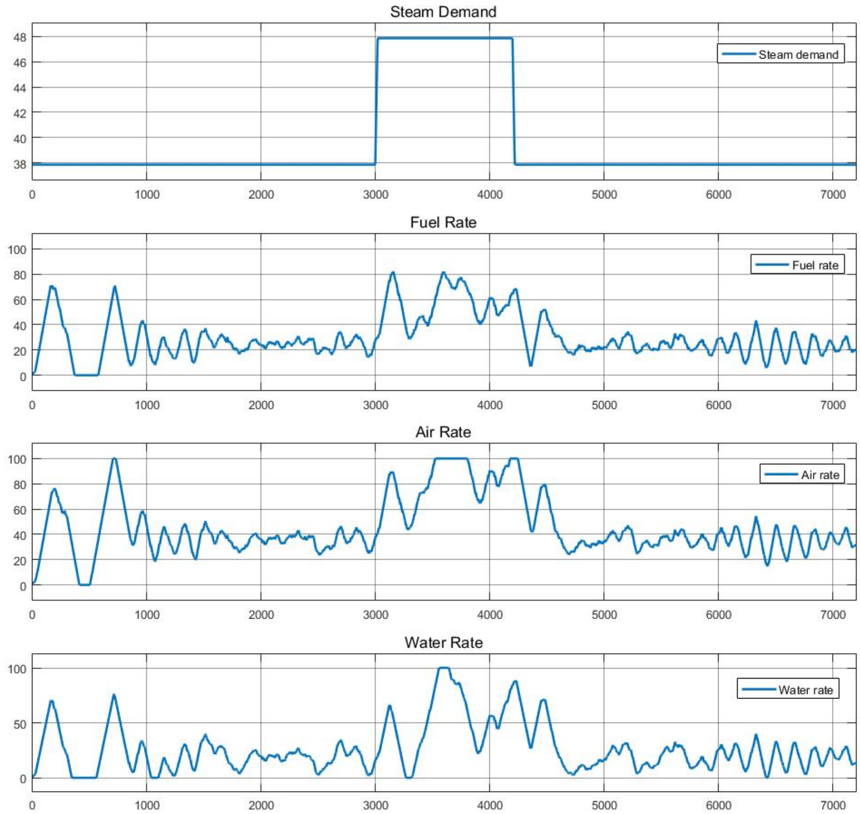


(b) Inputs

FIGURE 4.6: Scenario 3: Close loop LQR controller with steam demand variation, $R = diag(1e-3)$ and $Q = diag(20)$



(a) Outputs



(b) Inputs

FIGURE 4.7: Scenario 4: Close loop LQR controller with steam demand variation, $R = diag(1e-3)$, $Q = diag(20)$ and $Rate = 0.5$

4.2.1 Evaluation of controller with benchmark code

To finalize the results analysis we proceed to perform a comparison between the obtained closed loop controller response when applied to the simulation, against the results available from the original data.

A benchmark code proposed by the CEA was used, where the the input/output data of the LQR simulations was introduced to obtain the results of the adjustment of the designed controller with respect to the one that it was proposed as reference. The evaluation is performed by minimizing a cost function J.

The first step it was to decide which one of the previously explained scenarios was the more correct one in terms of correct control. By seeing the obtained results we opt to use the second one, given the fact that even when the closed loop response is overdamped, the system behaves correctly reaching the steady state without problem.

Knowing the method to be compared we proceed to run the code comparison for the best three obtained controllers based on the LHP closed loop pole placement. The ones elected was the K_3, K_4 and K_6.

Also we want to study if the minimization of the cost function is better for the case of steam demand variation or for the case of no steam demand variation. In order to do this for each controller we simulate the response of the system for both scenarios and storage the corresponding data to perform the benchmarking.

In Figures 4.8, 4.9 and 4.10 we can the results of the benchmark performed to each controller in the case of no steam demand variation. In this case the obtained minimization of the cost function J was:

$$K_3(1) \rightarrow J = 31.2518$$

$$K_4(1) \rightarrow J = 33.3107$$

$$K_6(1) \rightarrow J = 29.0377$$

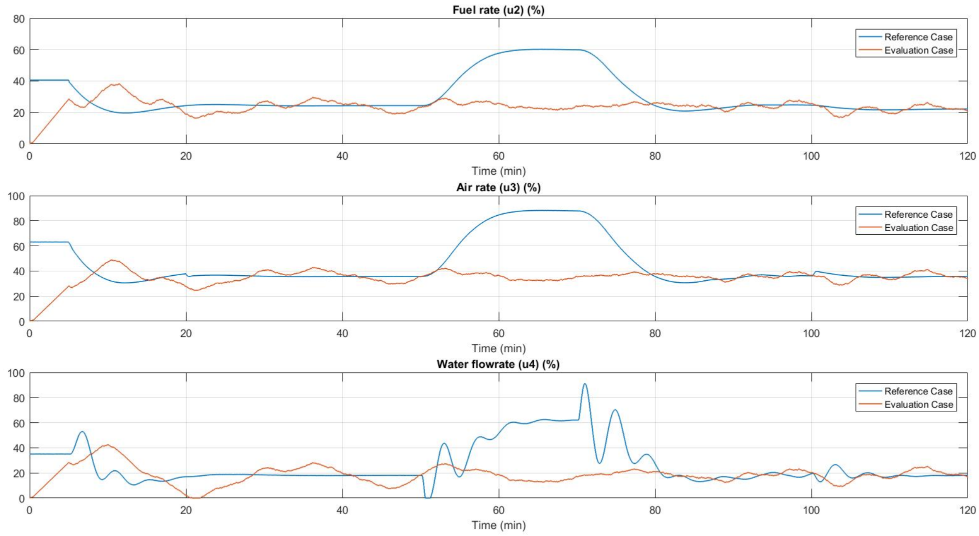
In Figures 4.11, 4.12 and 4.13 we can the results of the benchmark performed to each controller in the case of steam demand variation. In this case the obtained minimization of the cost function J was:

$$K_3(1) \rightarrow J = 83.7495$$

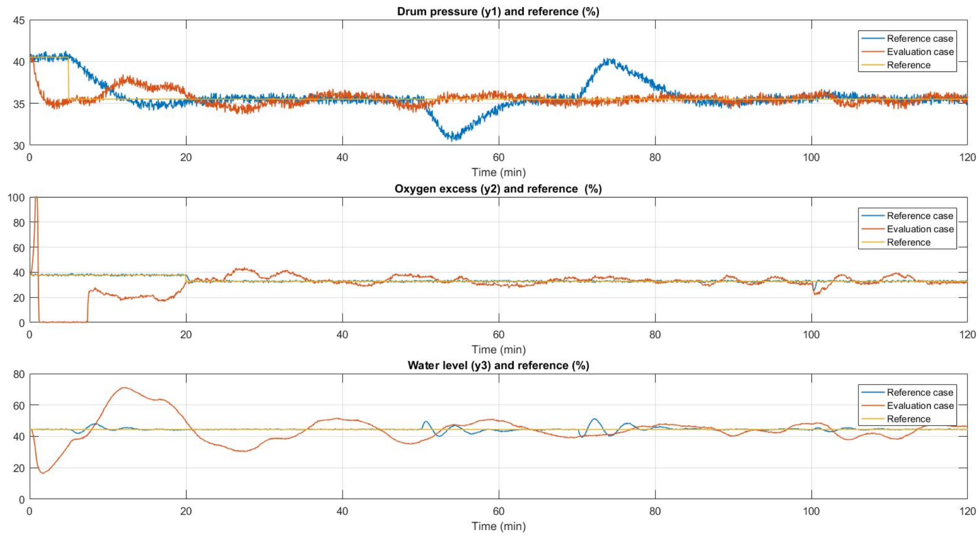
$$K_4(1) \rightarrow J = 65.6681$$

$$K_6(1) \rightarrow J = 75.5940$$

With the obtained results it can be concluded that the controller that proves to be more adapted to the performing of the referenced data given by the CEA it would be the K_4. However, even when the LQR controller is prove to be able to stabilize the plant, it is quite notable that the values of the minimization of the cost function is not the more desirable.

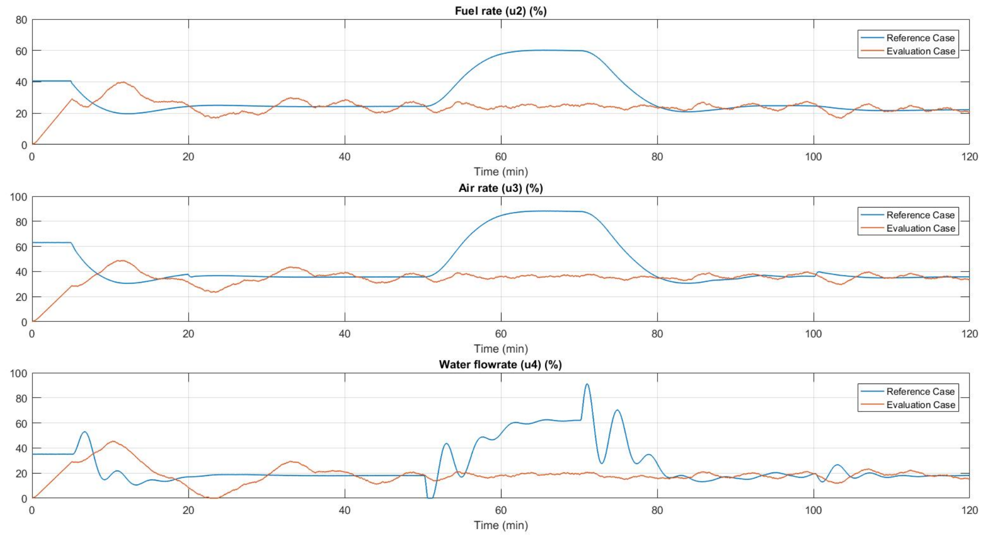


(a) Inputs of K_3 compared to inputs of reference data

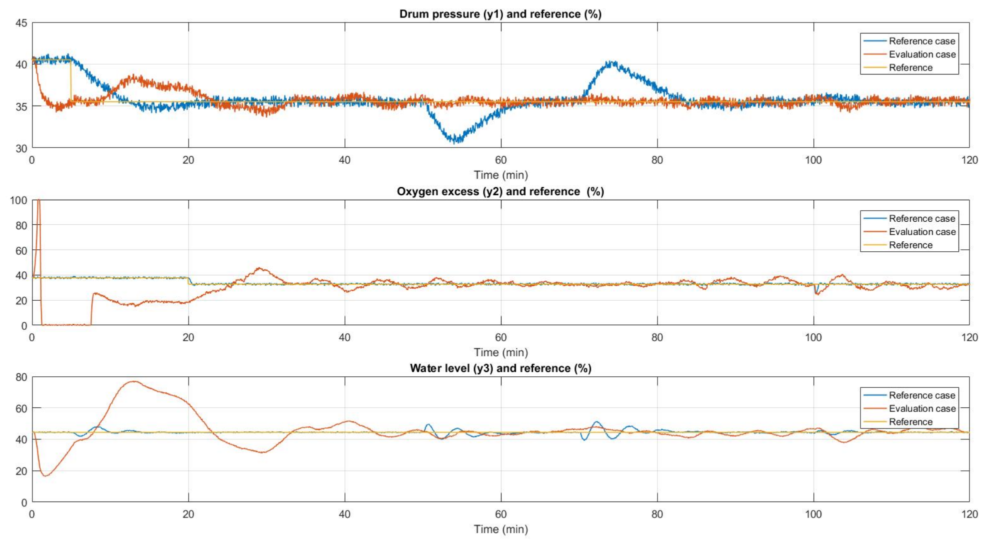


(b) Outputs of K_3 compared to inputs of reference data

FIGURE 4.8: Benchmark of K_3 without steam demand variation ($J = 31.2518$)

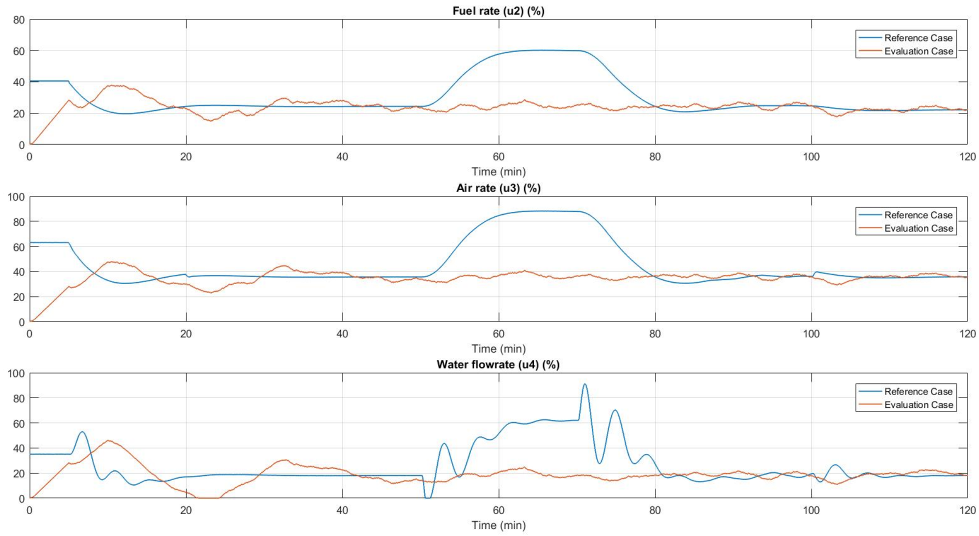


(a) Inputs of K_4 compared to inputs of reference data

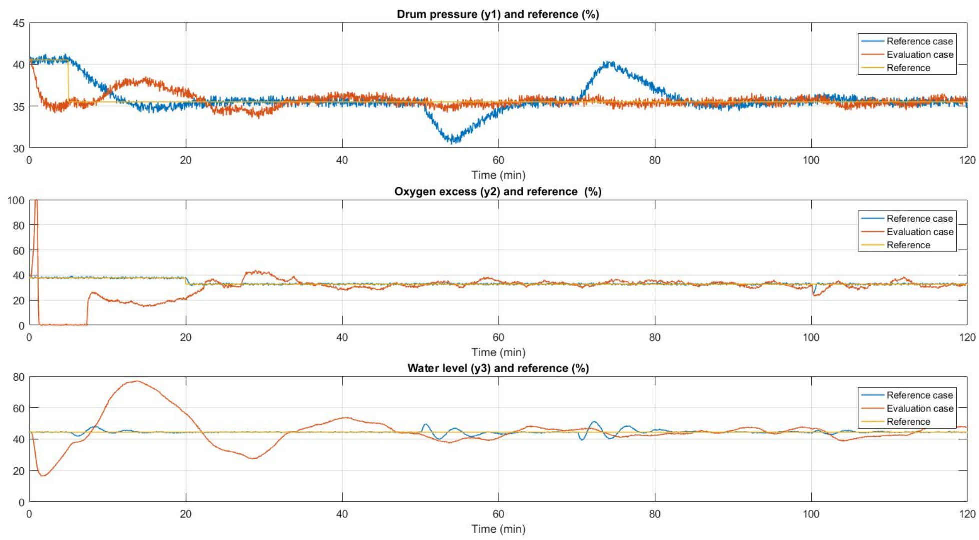


(b) Outputs of K_4 compared to inputs of reference data

FIGURE 4.9: Benchmark of K_4 without steam demand variation ($J = 33.3107$)

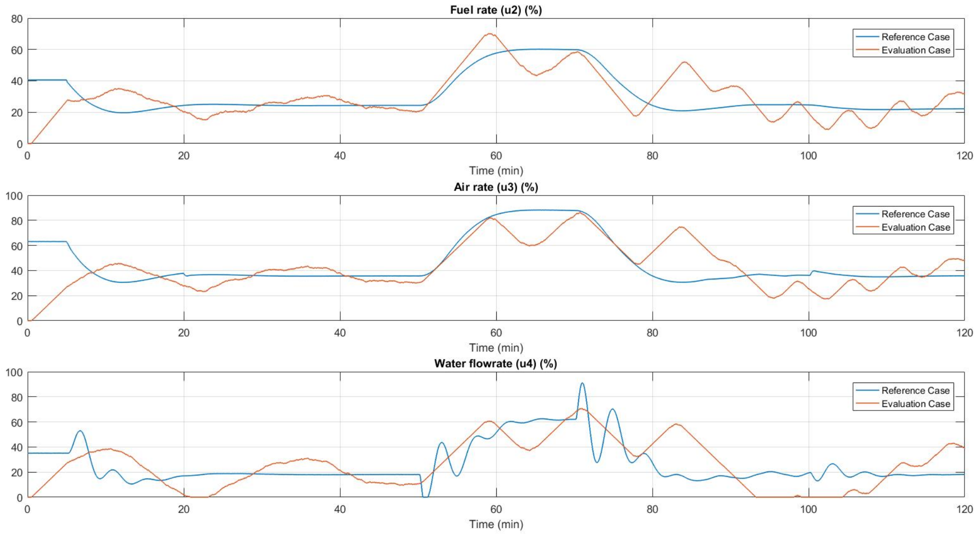


(a) Inputs of K_6 compared to inputs of reference data

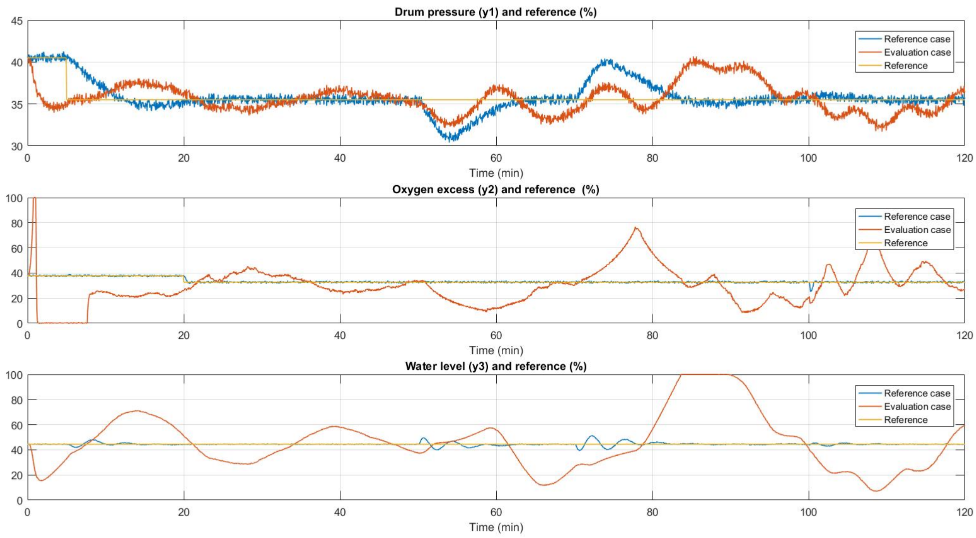


(b) Outputs of K_6 compared to inputs of reference data

FIGURE 4.10: Benchmark of K_6 without steam demand variation ($J = 29.0377$)

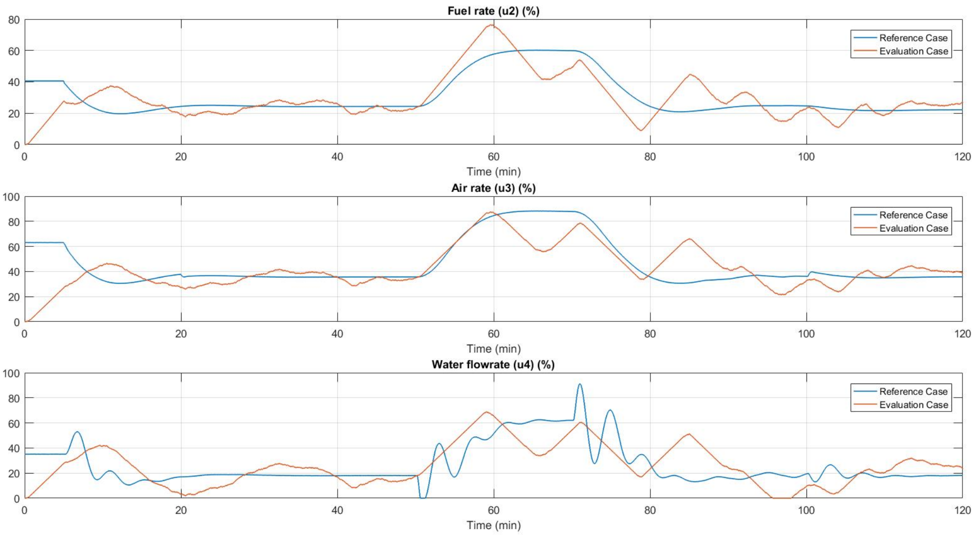


(a) Inputs of K_3 compared to inputs of reference data

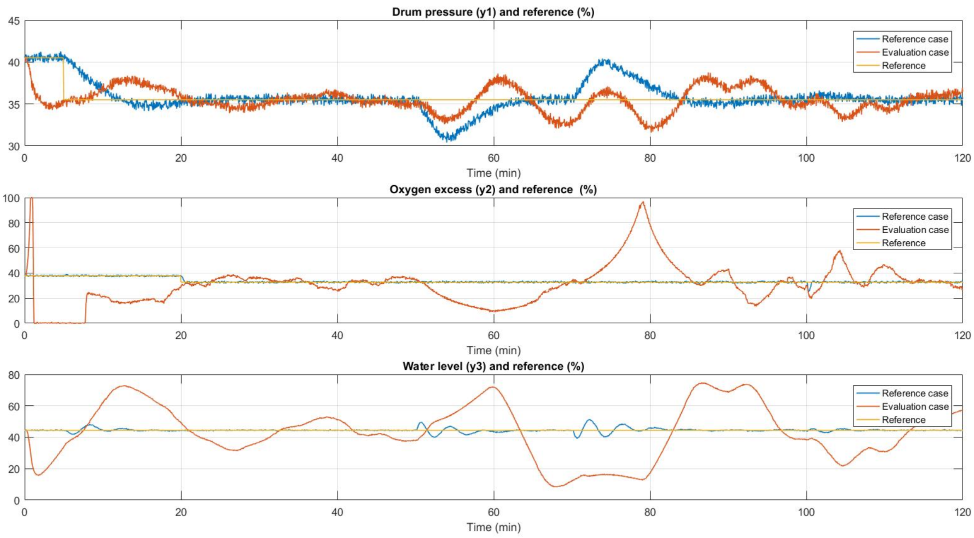


(b) Outputs of K_3 compared to inputs of reference data

FIGURE 4.11: Benchmark of K_3 with steam demand variation ($J = 83.7495$)

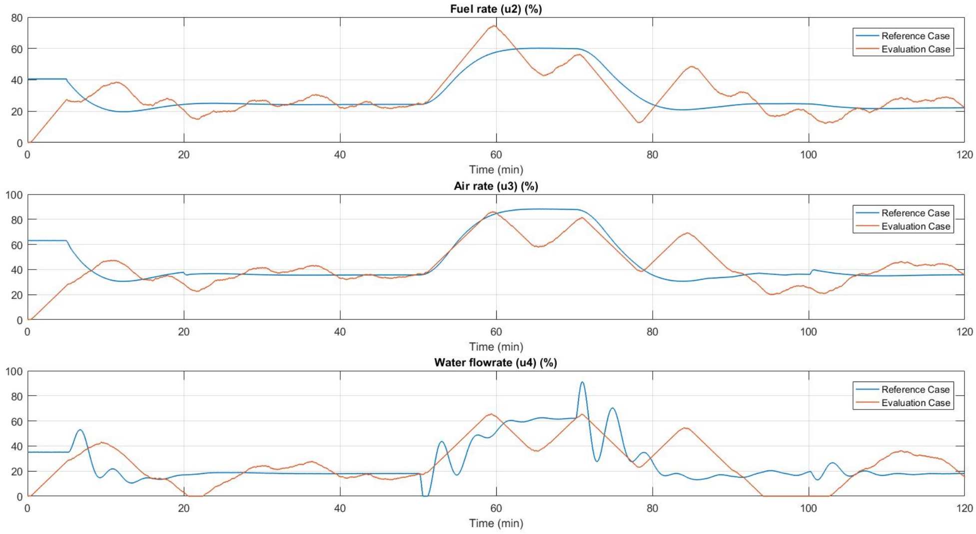


(a) Inputs of K_4 compared to inputs of reference data

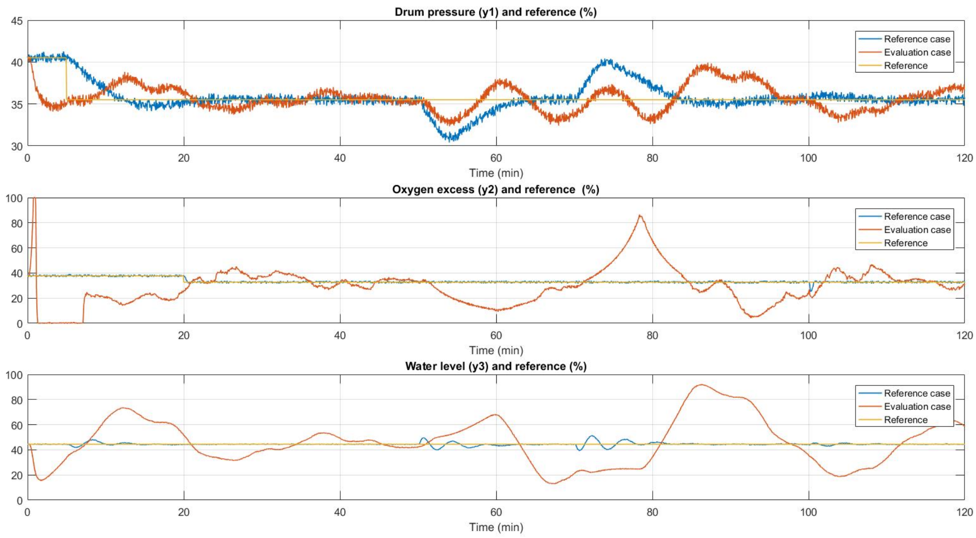


(b) Outputs of K_4 compared to inputs of reference data

FIGURE 4.12: Benchmark of K_4 with steam demand variation ($J = 65.6681$)



(a) Inputs of K_6 compared to inputs of reference data



(b) Outputs of K_6 compared to inputs of reference data

FIGURE 4.13: Benchmark of K_6 with steam demand variation ($J = 75.5940$)

Chapter 5

Budget and Impact Studies

This chapter covers the budget analysis of the work and a discussion about the impact of the work from an economical and environmental point of view.

5.1 Budget Study

In this section will be presented the associated costs of the work for each one of the different factors: hardware, software, human resources and general expenses.

5.1.1 Hardware Resosources

The hardware resources are a personal computer. It is considered a media of 12 hours of work per week, and 30 weeks for the project and the value unit will be Euro (EUR). The amortization will be based on 40 working hours per week and 52 weeks a year.

TABLE 5.1: Cost associated to hardware resources

Resource	Unit Price	Amortization	Price/hour	Hours of use	Amortization
PC	800 EUR	4 years	0.10 EUR	360	36.00 EUR
Total	1000 EUR	-	-	-	36.00 EUR

5.1.2 Software Resources

The software resources are an Operative System and other developed software of statistical analysis and transcription. It is considered 30 weeks for the project and the value unit will be Euro (EUR). The cost will be based on 52 weeks a year. The browsers on the World Wide Web and the documents viewers are free. And the main software for simulations, programing and results manipulation is MATLAB 2014a full version.

5.1.3 Human Resources

As a student doing thesis and needing no external human resources, the cost is 0 EUR. The minimum salary per hour according to the UPC for a student doing practices is 8 EUR per hour but these tasks can be divided in a private company between a Project Manager, a Tester and a Programmer during 360 hours. As an approximation, the HHRR cost for students is 3200 EUR and professional salary could be around 14000 EUR.

TABLE 5.2: Cost associated to software resources

Resource	Unit Price	Valid for	Price/week	Weeks of use	Cost
Ubuntu	0 EUR	Inf	0.00 EUR	30	0.00 EUR
Overleaf	0 EUR	Inf	0.00 EUR	30	0.00 EUR
MATLAB Student	69 EUR	4 years	0.33 EUR	30	10.00 EUR
MATLAB Coder	7 EUR	4 years	0.03 EUR	30	1.00 EUR
MATLAB Report	7 EUR	4 years	0.03 EUR	30	1.00 EUR
Simulink Opt	7 EUR	4 years	0.03 EUR	30	1.00 EUR
Simulink Report	7 EUR	4 years	0.03 EUR	30	1.00 EUR
Ident Toolbox	7 EUR	4 years	0.03 EUR	30	1.00 EUR
Browsers	0 EUR	Inf	0 EUR	30	0 EUR
Acrobat Reader	0 EUR	Inf	0 EUR	30	0 EUR
Total	230 EUR	-	-	-	15.00 EUR

5.1.4 General Resources

These expenses are generally the use of the working space, the energy consumed by the tools, etc. It is estimated that the cost for all the project is around 300 EUR.

5.1.5 Total Cost

Finally a margin can be considered in case of contingences and it is around the 20%.

TABLE 5.3: Total cost associated to the thesis

Category	Value
Hardware resources	36.00 EUR
Software resources	15.00 EUR
Human Resources	2880.00 EUR
General expenses	300.00 EUR
Subtotal	3231.00 EUR
Margin (20%)	646.20 EUR
Total costs	3877.20 EUR

5.2 Impact Study

5.2.1 Economical Impact

The improvement in the new techniques and implement them in the industry have been bringing a lot of economic benefits along the history and this is not an exception. If the changes presented in this thesis with the MIMO systems are implemented, the interaction between the variables can be more detailed and in this way the time and the quality of the tuning will improve a lot.

The Steam boiler is only an example of what can be done with a MIMO controller in MIMO systems, and this can be perfectly reproduced for other system inside big industrial plants. So, this can help in the fault detection, the time of the professionals involved in the project and the quality of the final product.

5.2.2 Social Impact

The society can be beneficiated with the application of these technologies in long term. After all the changes and improvements were applied in the future, it is supposed that the renewable energy and the exploitation of it will give a better quality of life to the people around this technological wave.

5.2.3 Environmental Impact

The implementation of more efficient controllers in each sector of the industry will optimize the processes and the evolution of the results will be more friendly with the environment, cheaper and easy to apply.

The energy is one of the big concerns of the actual world community and it has a lot of improvement margin in the industrial sector, so if the optimization is applied in a lot of MIMO systems inside a plant the final result should be an exploitation of all the resources and consequently the improvement in the environmental field of a region.

The application of all the technology available in the right moment should slow down the production of carbon dioxide and with this increase the quality of life of the future generations.

Chapter 6

Concluding Remarks

6.1 Conclusions

This thesis has proposed a MIMO controller strategy to control a boiler plant with the implementation of IMC using a transfer function model and LQR using a state space model. This two approaches can be a transition between the actual industrial controllers (PLCs), which use SISO control strategies, and the MIMO techniques.

The plant was identified in two different ways and both methods was validated. Based on the obtained models the IMC and LQR controllers was designed in order to implement both individually in the given plant. The IMC control have problems to be implemented given the complexity of the transfer function model that have been identified. The resulting MIMO transfer function have an elevated order, and adding the data delay, disturbances and uncertainty, the IMC controller have a lot poles that needs to be placed one by one. There is no direct connection between each input and output of the plant. Being a MIMO system there exist a strong link between the majority of inputs/outputs and the IMC for this particular case does not count with easy tuning capability.

On the contrary, the LQR control strategy has proven to be more efficient given its easy tuning capability. It turns out that regardless of the values of Q and R, the cost function has a unique minimum that can be obtained by solving the Algebraic Riccati Equation. Knowing this the parameters Q and R are used as design parameters to penalize the state variables (in the case of Q) and the control signals (in the case of R). Particularly, given the trade off between both parameters, we only focus our attention on tuning the R parameters and we obtain good results.

Through this thesis, each chapter already presents important conclusions about the proposed methods to identify and control and the system by itself, nevertheless, some final comments are remarked below.

1. The plant can be modeled as battery of transfer functions to implement an IMC controllers and the identification tool offer this option.

2. The IMC does not offer the simpliest controller to implement in the system due to the big amount of dynamics described by the internal model in this case.

3. The state space model was the simplest model to apply a controller, which in this case is the LQR.

4. The LQR behaves as a normal industrial controller, checking when the setpoints are not satisfied and depending on the weights, the correction is more aggressive or not.

6.2 Contributions

The main contributions of this thesis are listed below:

1. The analysis of the application of MIMO controllers to industrial plants and suggest to take that step forward in the implementation of these techniques.

2. To design an IMC and LQR controller and see how useful are for the industrial plants.

6.3 Future Work

Some of the future work that can be implemented thanks to the point of view of this thesis are listed below:

1. To create an user interface for MIMO systems and show that they work as same as the SISO systems.

2. Keep implementing new types of controllers that can use transfer functions in order to compare them to the actual ones.

3. Apply the LQR into the actual industrial and software interfaces to see how they behave and adapt it to the reality.

Bibliography

[1] Adkins, C. (1968). Equilibrium Thermodynamics. London: McGraw-Hill.

[2] Astrom, K., & Bell, R. (1987). Dynamic Models for Boiler-Turbine-Alternator Units: Data Logs and Parameter Estimation for a 160 MW Unit. Lund. Sweden: Department of Automatic Control. Lund Institute of Technology.

[3] Bird, B., Stewart, W., & Lightfoot, E. (2007). Transpor Phenomena. New York: Jhon Wiley & Sons .

[4] Lanco, M. (2016).Model Predictive Fuzzy Control of a Steam Boiler. Master in Automatic and Robotics Departament d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial (ESAII). Institut de Robòtica i Informàtica Industrial. Universitat Politécnica de Catalunya.

[5] Bordons, C. (2000). I Curso de Especialización en Automática. Universidad de Sevilla.

[6] Comité Español de Automática. (2016). Concurso de Ingeniería de Control 2016. Control de una Caldera de Vapor. Madrid: Ingeniería de Control.

[7] Fernández, I., & Rodríguez, C. (2010). Control de una Caldera. Almería: Grupo Temático de Control de CEA.

[8] Grosso, J. (2012). A Robust Adaptive Model Predictive Control to enhance the Management of DrinkingWater Networks subject to Demand Uncertainty and Actuators Degradation. Barcelona: Universidad Poliétcnica de Cataluña.

[9] Mathworks. (2016, July). MATLAB & Simulink - Black-Box Models. From http://es.mathworks.com/help/ident/ug/black-box-modeling.html

[10] Mathworks. (2016, July). Specifying Constraints - MATLAB. From http://es.mathworks.com/help/mpc/ug/specifying-constraints.html#buj077i

[11] Mathworks. (2016, July). Types of model objects - MATLAB & Simulink. From http://es.mathworks.com/help/ident/ug/types-of-model-objects.html

[12] Mathworks. (2016, July). Linear-Quadratic Regulator - MATLAB & Simulink. From http://es.mathworks.com/help/control/ref/lqr.html

[13] Pellegrinetti, G., & Bentsman, J. (1996). Nonlinear Control Oriented Boiler Modeling - A Benchmark Problem for Controller Design. IEEE Transactions on Control Systems Technology, Vol. 4.

[14] Philipson, & Schuster. (2009). Modeling by Non Linear Differential Equations: Dissipative and Conservative Processes. World Scientific Publishing.

[15] Reina, A. (2016, July). Study. From http://study.com/academy/lesson/heat-of-vaporization-definition-equation.html

[16] Rivera, Daniel. (1999). Internal Model Control: A comprehensive View. College of Engineering and Applied Sciences Arizona State University, Tempe, Arizona.

[17] Swevers, Jan. (2006) Internal model control (IMC). Systems and Control Theory. KU Leuven. Netherlands.

[18] Steingress, F. (2001). Low Pressure Boilers. American Technical Publishers.

[19] Zhu, F. (2014). Energy and Process Optimization for the Process Industries. Manchester: Wiley.