DESIGN AND ANALYSIS OF MULTIVARIABLE PREDICTIVE CONTROL APPLIED TO AN OIL-WATER-GAS SEPARATOR: A POLYNOMIAL APPROACH

By

GIOVANI CAVALCANTI NUNES

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2001

To my beloved daughters Julia and Giovanna and to my mother Nisete.

ACKNOWLEDGMENTS

I would like to thank Professor Oscar Crisalle for his help. His insight and knowledge of Process Control were fundamental in the development of this dissertation. I consider myself a lucky person for finding such a wonderful advisor.

I would also like to thank professor Santiago for his immense help and support in solving all kinds of problems that naturally arise to a foreign student in graduate school.

I gratefully acknowledge the support received from my employee Petrobras for this work. The conditions were excellent. I especially thank the engineer and leader Jose Antonio de Figueiredo, head of the Division of Offshore Basic Engineering, at the time, to whom this thesis owes its existence. His energy and ability to motivate people are outstanding.

Professor Antonio Coelho, a colleague and a friend, is another person I must mention for his help and solidarity during the long hours of work in the control laboratory. Excitement and enthusiasm were always present when Antonio "Banderas" was around.

I would finally like to thank the big group of wonderful friends I had the chance to meet while in Gainesville. I would like to list all their names here but this is unfeasible. Kerry Alahar, however, must be mentioned for our friendship proved to be a special one. Our common interests, experience and burdens during graduate school created bonds that will last forever.

TABLE OF CONTENTS

page

ACKNOWLEDGMENTS iv

ABSTRACT vii

CHAPTERS

1 INTRODUCTION 1

2 LITERATURE REVIEW 7
2.1 History of Model Predictive Control 7
2.2 Stability 9
2.3 Model Predictive Control: An Optimal Control Problem 10

3 FORMULATION AND SOLUTION OF THE CONTROL PROBLEM 12
3.1 Objective Function of Multivariable Predictive Control 12
3.2 Derivation of the Predictive Control Law 16
3.3 Derivation of the Constant Forcing Function 19
3.4 Final Control Law Equation 21
3.5 Conclusion 22

4 ANALYSIS OF CLOSED-LOOP BEHAVIOR 23
4.1 Stability Analysis of Closed-loop Multivariable Systems 23
4.2 Closed-loop Transfer Function Matrices 24
4.3 MFD of Transfer Function Matrices 26
4.4 Poles and Zeros of a Transfer Matrix 28
4.5 Performance Analysis 38
4.6 Implementation 42
4.7 Conclusion 44

5 THREE-PHASE SEPARATOR 45
5.1 Description of the Separator 46
5.2 Rigorous Model 47
5.3 Simplified Model 48
5.4 Flow Equations 51
5.5 Differential Equations 51
5.6 Conclusion 52
5.7 Nomenclature 53

6 DESIGN OF A PREDICTIVE CONTROLLER FOR THE SEPARATOR............. 55
6.1 Discrete Linear Model of the Separator 55
6.2 Design of Predictive Controller........ 57
6.3 Case 1: First Tuning 58
6.4 Case 1: Second Tuning........ 67
6.5 Case 2 75
6.6 Conclusion........ 81

7 SIMULATION AND RESULTS 82
7.1 Results of Case 1 82
7.2 Results of Case 2 86
7.3 Conclusion........ 89

8 CONCLUSIONS 90

APPENDICES

A. DETAILED DERIVATION OF CONTROL LAW EQUATION........ 92
A.1 Derivation of Matrix Diophantine Equation 92
A.2 Derivation of Predictor Equation 95
A.3 Simplification of Predictor Equation........ 97
A.4 Derivation of Constant Forcing Response Equation 100
A.5 Substitution of Constant Forcing Response in the Control Law Equation........ 100

B. DERIVATION OF PRODUCT *KP* AND *KF*........ 104
B.1 Product of ***KP*** 104
B.2 Product of ***KF*** 106

C. OBTAINING A LEFT MATRIX DESCRIPTION 107

D. CLOSED-LOOP TRANSFER MATRICES........ 110
D.1 Disturbances in the Closed-loop........ 110
D.2 Closed-loop Transfer Matrices of Predictive Control 112

E. OBTAINING MATRIX ***K*** 113

LIST OF REFERENCES 115

BIOGRAPHICAL SKETCH 118

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

DESIGN AND ANALYSIS OF MULTIVARIABLE PREDICTIVE CONTROL
APPLIED TO AN OIL-WATER-GAS SEPARATOR: A POLYNOMIAL APPROACH

By

Giovani Cavalcanti Nunes

August 2001

Chairman: Tim Anderson
Major Department: Chemical Engineering

This dissertation uses a polynomial-operator technique to study stability and performance of unconstrained multivariable predictive control. A simple and direct way to determine stability of the closed-loop system is developed.

It is shown that to guarantee stability of the closed-loop two transfer matrices must be stable. These are represented as fraction descriptions, a ratio of "numerator" and "denominator" polynomial matrices from which the poles can be determined. Because both transfer matrices possess the same denominator matrix the position of the roots of its determinant give sufficient conditions for stability of the closed-loop. Furthermore it is shown that if a coprime fraction description is done for the process transfer matrix then it is necessary and sufficient to check if the roots of the determinant of the denominator matrix lie inside the unit circle. This technique avoids the inversion of transfer matrices which is a numerically difficult task allowing the tuning of multivariable systems with many inputs and outputs.

Performance is also studied and it is proven that the system has zero offset response to step changes in the reference, a property known to be valid for the single-input single-output case. For systems with an equal number of inputs and outputs the "inversion of the plant" is also proven. In this case the weights on the input are zero and the solution to the optimization problem results in a controller that inverts the plant and the output matches the reference.

The use of a multivariable predictive control for an oil-water-gas separator is studied. The nonlinear model of the plant is linearized around a steady state and different predictive controllers are designed for it. The controller responds positively to changes in the parameters and performance objectives are pursued. Results show agreement with the simulations done for the linear model, it is concluded that predictive control is a successful control strategy for oil-water-gas separators.

This method becomes an important tool for the analysis of predictive controllers, allowing the study of the effect of tuning parameters on the behavior of the system when the constraints are removed.

CHAPTER 1
INTRODUCTION

The process industry is characterized by ever tighter product quality specifications, increasing productivity demands, new environmental regulations and fast changes in the economical market. Over the last two decades predictive control has proven to be a very successful controller design strategy, both in theory and practice. The main reason for this acceptance is that it provides high performance controllers that can easily be applied to difficult high-order and multivariable processes. Process constraints are handled in a simple and systematic way. However a general stability and robustness theory is lacking.

Predictive control is a class of control strategies based on the explicit use of a process model to generate the predicted values of the output at future time instants, which are then used to compute a sequence of control moves that optimize the future behavior of a plant. Predictive control is rather a methodology than a single technique. The difference in the various methods is mainly the way the problem is translated into a mathematical formulation.

The explicit use of a model is the main difference between predictive control and the classical PID controller. Its advantage is that the behavior of the controller can be studied in detail, simulations can be done and performance can be evaluated. One of the drawbacks is the need of an appropriate model of the process. The benefits obtained are affected by the discrepancies existing between the real process and the model. According to some researchers 80% of the work done is in modeling and identification of the plant.

The results almost always show that the effort is paid back in short time. Another drawback is that although the resulting control law is easy to implement and requires little computation, its derivation is more complex than that of the PID. If the process dynamics do not change, the derivation can be done beforehand, but in the adaptive control case all the computation has to be carried out at each sampling time. When constraints are considered, the amount of computation is even higher.

Predictive control was pioneered simultaneously by Richalet *et al*. (1978) and Cutler and Ramaker (1980). The use of finite-impulse response models and finite-step response models, which are easily obtained for open loop stable processes, partly explains the acceptance in the process industry.

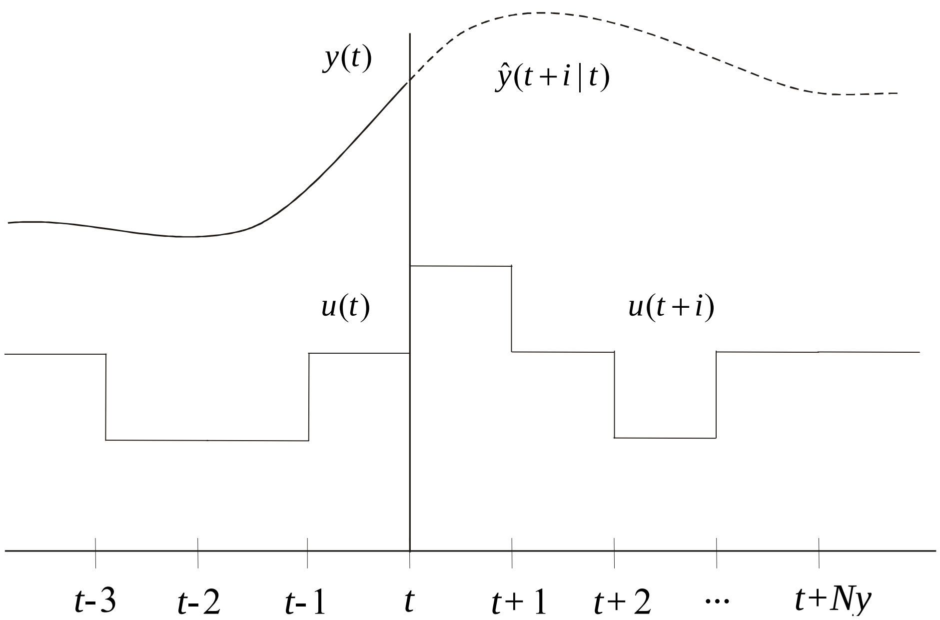

Figure 1: Predictive control strategy

The methodology of all predictive controllers consists in predicting, at each time *t*, the future outputs for a determined horizon Ny, called the prediction horizon. This prediction of the outputs, $\hat{y}(t+i\,|\,t)$ for i = 1… Ny, is based on the model of the process and depends on the known values of past inputs and outputs up to instant t.

The set of future control signals is calculated by optimizing a given criterion (called objective function or performance index) in order to keep the process as close as possible to the reference trajectory $r(t+i)$, which can be the set point or an approximation of it. Different algorithms present different forms of objective functions, which usually take the form of a quadratic function of the errors between the predicted output signal and the predicted reference trajectory. Some algorithms use the states of the process as opposed to the outputs. The control effort is included in the objective function in most cases. Weights are used to adjust the influence of each term in the equation. The solution to the problem is the future control sequence that minimizes the objective function equation. For that, a model is used to predict future outputs or states.

A typical objective function equation of a single-output single-input process is

$$J = \sum_{i=Ny1}^{Ny2} \phi(i)(r(t+i) - \hat{y}(t+i\,|\,t))^2 + \sum_{i=1}^{Nu} \lambda(i)\Delta u(t+i-1)^2$$

It is a quadratic function of future inputs, $u(t+i)$, and the error between future values of reference $r(t+i)$ and predicted outputs, $\hat{y}(t+i\,|\,t)$. Weights ϕ and λ are used to adjust the influence of the error and inputs respectively.

The sequences of predicted outputs and future inputs are limited to horizons $Ny2$ and Nu respectively. The limitation on the sequence of inputs, $u(t+i)$ from i=1 to Nu, comes from the assumption that the control action is constant after Nu steps ahead. The prediction horizon, on the other hand, limits the sequence of predicted output considered in the objective function equation. The control horizon has to be smaller than the prediction horizon. Weights and horizons are tuning parameters of the controller.

The optimization of the objective function requires a prediction of the future outputs. The predicted output is the addition of two signals:

$$\hat{y}(t+i\,|\,t) = y^0(t+i\,|\,t) + \boldsymbol{G}\Delta u(t+i)$$

the constant forcing response and forced response. The constant forcing response, $y^0(t+i\,|\,t)$, corresponds to the prediction of the evolution of the process under the consideration that future input values will be constant. The forced response, $\boldsymbol{G}\Delta u(t+i)$ where $\boldsymbol{G}$ is the *dynamic matrix* of the process, corresponds to the prediction of the output when the control sequence is made equal to the solution of the minimization of the objective function.

The expression for the predicted outputs can be substituted in the objective function and the solution to the minimization problem leads to the desired control sequence. Once the control sequence has been obtained only the first control move is implemented. Subsequently the horizon is shifted and the values of all sequences are updated and the optimization problem is solved once again. This is known as the Receding Horizon Principle and is adopted by all predictive control strategies. It is not advisable to implement the entire sequence over the next Nu intervals because it is impossible to perfectly estimate the unavoidable disturbances that cause the actual output to differ from the predictions made. Furthermore the operation might decide to change the set point over the next Nu intervals.

The various predictive control algorithms only differ themselves in the model used to represent the process, the model for the noise and the objective function to be minimized. The models used can be Impulse/Step response models, transfer function models or state space models.

- Impulse/Step response models: Predictive control has part of its roots in the process industry where the use of detailed dynamical models is uncommon. Getting reliable dynamical models of these processes on the basis of physical laws is difficult and the first models used were Impulse/Step response models. The models are easily obtained by rather simple experiments and give good results. The disadvantage is that it requires a large amount of parameters.

- Transfer function models: For some processes good models can be obtained on the basis of physical laws or by parametric system identification. In this case a transfer function model is preferred. Less parameters are used than in the Impulse/Step response models.

- State Space models: These models are the most general description for a linear time invariant system and being so are the most elaborate and difficult to generate.

Generalized Predictive Control (GPC) is a class of predictive control, proposed by Clarke *et al*. (1987). It provides an analytical solution (in the absence of constraints) and can deal with unstable and non-minimum phase plants. It uses transfer function model for the plant leading to lower-order representations.

Different transfer function models are used by GPC such as Controller Auto-Regressive Integrated Moving Average (CARIMA), Controller Auto-Regressive Moving Average (CARMA), etc. In this dissertation a Deterministic Auto-Regressive Moving Average (DARMA) model is considered

$$\boldsymbol{A}(q^{-1})\boldsymbol{y}(t) = \boldsymbol{B}(q^{-1})\boldsymbol{u}(t-1) \tag{1.1}$$

which can be obtained by a left matrix fraction description (see Appendix C) of the transfer function model.

Most industrial plants have many outputs and manipulated variables. In certain cases a manipulated variable mainly affects the corresponding controlled variable and each of the input-output pairs can be considered as a single-input single-output (SISO) plant and controlled by independent loops. In cases where the interaction between the different variables is not negligible the plant must be considered as a multiple-inputs and multiple-outputs (MIMO) process. These interactions may result in poor performance and even instability if the process is treated as multiple SISO loops.

In practice all processes are subject to restrictions and these can be considered in the objective function equation as constraints on the inputs and outputs. In many industrial plants the control system will operate close to the boundaries due to operational conditions. This has lead to the widespread belief that the solution to the optimization problem lies at the boundaries. On the other hand the design of multivariable predictive controllers requires the specification of horizons and weights for all inputs and outputs. Therefore a big number of parameters must be selected, a task that may prove challenging even for systems of modest size. Badly tuned predictive controllers tend to take the plant to extreme and sometimes unstable conditions, frequently unnoticed because of physical constraints such as valve opening, maximum flow rate, etc., masking the instability problem and fooling operational groups into concluding that the plant has been optimized.

Many different predictive control algorithms have treated multivariable systems in some way or another but none has established the mathematical basis for a comprehensive analysis of the system, such as performance or stability.

In this dissertation a practical method for analyzing the stability of classical multivariable predictive control systems is pursued. The focus is on unconstrained systems in order to allow the identification of the destabilizing choices of tuning parameters, avoiding the masking effect introduced by the presence of hard constraints.

CHAPTER 2
LITERATURE REVIEW

Model predictive control (MPC) has had an enormous impact in the process industry over the last 15 years. It is an effective means of dealing with multivariable constrained control problems and the many reports of industrial applications confirm its popularity.

2.1 History of Model Predictive Control

The development of modern control concepts can be traced back to the work of Kalman in the early 1960s with the linear quadratic regulator (LQR) designed to minimize an unconstrained quadratic objective function of states and inputs. The infinite horizon endowed the LQR algorithm with powerful stabilizing properties. However it had little impact on the control technology development in the process industries. The reason for this lies in the absence of constraints in its formulation, the nonlinearities of the real systems, and above all the culture of the industrial process control community at the time, in which instrument technicians and control engineers either had no exposure to optimal control concepts or regarded them as impractical. Thus the early proponents of MPC for process control proceeded independently, addressing the needs of the industry.

In the late 1970s various articles reported successful applications of model predictive control in the industry, principally the ones by Richalet *et al.* (1978) presenting Model Predictive Heuristic Control (later known as Model Algorithmic Control (MAC)) and those of Cutler and Ramaker (1980) with Dynamic Matrix Control (DMC). The common theme of these strategies was the idea of using a dynamic model of the process (impulse response in the former and step response in the later) to predict the effect of the future control actions, which were determined by minimizing the predicted error subject to operational restrictions. The optimization is repeated at each sampling period with updated information from the process. These formulations were algorithmic and also heuristic and took advantage of the increasing potential of digital computers at the time. Stability was not addressed theoretically and the initial versions of MPC were not automatically stabilizing. However, by focusing on stable plants and choosing a horizon large enough compared to the settling time of the plant, stability is achieved after playing with the weights of the cost function.

Later on a second generation of MPC such as quadratic dynamic matrix control (QDMC; Garcia, Morshedi, 1986) used quadratic programming to solve the constrained open-loop optimal control problem where the system is linear, the cost quadratic, the control and state constraints are defined by linear inequalities.

Another line of work arose independently around adaptive control ideas developing strategies essentially for mono-variable processes formulated with transfer function models (for which less parameters are required in the identification of the model) and Diophantine equation was used to calculate future input. The first initiative came from Astron *et al*. (1970) with the Minimum Variance Control where the performance index to be minimized is a quadratic function of the error between the most recent output and the reference (i.e. the prediction horizon Ny=1). In order to deal with non-minimum phase plants a penalized input was placed in the objective function and this became the Generalized Minimum Variance (GVM) control. To overcome the limitation on the horizon, Peterka (1984) developed the Predictor-Based Self-Tuning Control. Extended Prediction Self–Adaptive Control (EPSAC) by De Keyser *et al*. (1985) proposes a constant control signal starting from the present moment while using a sub-optimal predictor instead of solving a Diophantine equation. Later on the input was replaced by the increment in the control signal to guarantee a zero steady-state error. Based on the ideas of GVM Clarke *et al*. (1987) developed the Generalized Predictive Control (GPC) and is today one of the most popular methods. A closed form for the GPC is given by Soeterboek. State-space versions of unconstrained GPC were also developed.

2.2 Stability

Stability has always been an important issue for those working with predictive control. Due to the finite horizon stability is not guaranteed and is achieved by tuning the weights and horizons. Mohtadi proved specific stability theorems of GPC using state-space relationships and studied the influence of filter polynomials on robustness improvement. However a general stability property for predictive controllers, in general, with finite horizons was still lacking. This lead researchers to pursue new predictive control methods with guaranteed stability in the 1990s. With that purpose a number of design modifications have been proposed since then including the use of terminal constraints (Kwon *et al*., 1983; Meadows *et al*. 1995), the introduction of dual-mode designs (Mayne and Michalska, 1993) and the use of infinite prediction horizons (Rawlings and Muske, 1993), among others. Clarke and Scattolini (1991) and Mosca *et al*. (1990) independently developed stable predictive controllers by imposing end-point equality constraints on the output after a finite horizon. Kouvaritakis *et al*. (1992) presented a stable formulation for GPC by stabilizing the process prior to the minimization of the objective function. Many of these techniques are specialized for state-space representations of the controlled plant, and achieve stability at the expense of introducing additional constraints and modifying the structure of the design. Practitioners, however, avoid changing the structure of the problem and prefer to achieve stability by tuning the controller. For that a good doses of heuristics is used.

2.3 Model Predictive Control: An Optimal Control Problem

Recently a theoretical basis for MPC has started to emerge. Researchers have revisited the LQR problem arguing that model predictive control essentially solves standard optimal control problems with receding horizon, ideas that can be traced back to the 1960s (Garcia *et al*., 1989). MPC is characterized as a mathematical programming problem since the slow dynamics of process industry plants allow on-line solution to open-loop problems in which case the initial state is the current state of the system being controlled. Determining the feedback solution, on the other hand, requires the solution of the Hamilton-Jacobi-Bellman (a dynamic programming problem) differential or difference equation which turns out to be more difficult. Riccati equation appears as a particular case to some optimal control problems. It is shown (Mayne *et al*., 2000) that the difference between MPCs approach and the use of dynamic programming is purely one of implementation. This line of research has an early representative in the work of Kwon and Pearson (1983) and Keerthi and Gilbert (1988) and has recently gained popularity through multiple advocates, such as the work of Muske and Rawlings (1993).

More recently two major approaches to the control problem are very popular. The first one employs the optimal cost function, for a fixed horizon, as a Lyapunov function and the second approach takes advantage of the monotonicity property of a sequence of the optimal cost function for various horizons. Note that for linear systems the presence of hard constraints makes the design of the controller a nonlinear problem so that the natural tool for establishing stability is Lyapunov theory.

Obtaining a formulation which can relate the tuning parameters of MPC to properties such as stability and performance is the major goal of this line of work and recent advances show promising results. Solving the dynamic programming problem, however, is not practical and this can be accounted for the resistance of the industry to adopt these new methods.

CHAPTER 3
FORMULATION AND SOLUTION OF THE CONTROL PROBLEM

In this chapter the predictive control law is derived for an unconstrained MIMO system with m controlled outputs and p inputs. The problem of minimizing the objective function is solved analytically using least squares. The formulation is done in vector matrix format. A final expression, function of future values of reference and output, is reached. Diophantine equation is used to generate the predicted values of the output.

3.1 Objective Function of Multivariable Predictive Control

Unconstrained multivariable predictive controllers are designed to minimize the objective function:

$$\begin{aligned} J = & \sum_{i=1}^{Ny_1} \phi_1(i)(r_1(t+i) - \hat{y}_1(t+i\,|\,t))^2 + \sum_{i=1}^{Ny_2} \phi_2(i)(r_2(t+i) - \hat{y}_2(t+i\,|\,t))^2 + \dots \\ & + \sum_{i=1}^{Ny_m} \phi_m(i)(r_m(t+i) - \hat{y}_m(t+i\,|\,t))^2 + \sum_{i=1}^{Nu_1} \lambda_1(i)\Delta u_1(t+i-1)^2 \\ & + \sum_{i=1}^{Nu_2} \lambda_2(i)\Delta u_2(t+i-1)^2 + \dots + \sum_{i=1}^{Nu_p} \lambda_p(i)\Delta u_p(t+i-1)^2 \end{aligned} \tag{3.1}$$

where:

$\hat{y}_j$ is the jth output

r_j is the desired trajectory for the jth output

u_ℓ is the ℓth input

ϕ_j are the weights on the error between the desired trajectory and the output

λ_ℓ are the weights on the control signal

Ny_j are the prediction horizons

Nu_ℓ are the control horizons

The general aim of the objective function is to make the future output, on the considered horizon, follow a desired reference signal and, at the same time, penalize the control effort (Δu) in order to avoid unfeasible inputs. Many formulations of predictive control consider an initial and a final prediction horizon to account for the dead times of the process. In this dissertation the initial prediction horizon considered is always 1 and if dead time is to be considered then the weights can be appropriately set to zero.

Similarly to the SISO case, where the prediction horizon has to be greater than the control horizon, the sum of all prediction horizons has to be greater than the sum of all control horizons; $Ny \geq Nu$ where

$$Ny \triangleq \sum_{j=1}^{m} Ny_j \text{ and } Nu \triangleq \sum_{\ell=1}^{p} Nu_\ell$$

Note that no control horizon should be greater than the max prediction horizon:

$$Nu_\ell \leq \max Ny_j \text{ for all } j$$

The many different indices required in the derivations that follow makes notation an important issue. The following rules are applied hereon:

- polynomials are written in upper case letters
- vectors in bold lower case letters
- matrices in bold upper case letters and in some exceptional cases bold lower case letters.

Vectors and matrices are created to represent the time expansion of the equations. Whenever the sequence of values from 1 to i has to be represented in a vector then $t + i$ will be written in between parenthesis. Its absence is indicative of a sequence of values from 1 to the given horizon. As an example, let us represent the values of $\hat{y}_1$ from step 1 to i and also from step 1 to Ny_1:

$$\hat{\boldsymbol{y}}_1(t+i\,|\,t) = \begin{bmatrix} \hat{y}_1(t+1\,|\,t) \\ \hat{y}_1(t+2\,|\,t) \\ \vdots \\ \hat{y}_1(t+i\,|\,t) \end{bmatrix} \qquad \hat{\boldsymbol{y}}_1 = \hat{\boldsymbol{y}}_1(t+Ny_1|t) = \begin{bmatrix} \hat{y}_1(t+1\,|\,t) \\ \hat{y}_1(t+2\,|\,t) \\ \vdots \\ \hat{y}_1(t+Ny_1|t) \end{bmatrix}$$

Notice the difference between the scalar $\hat{y}_1(t+i\,|\,t)$ and the vector $\hat{\boldsymbol{y}}_1(t+i\,|\,t)$. The former indicates the value of $\hat{y}_1$ at time step $t + i$ while the latter denotes the vector of values of $\hat{y}_1$ from time step 1 to i. Subscripts are used in matrices for the same purpose.

The objective function can be written in the more elegant vector-matrix format. This will simplify the notation and help visualization of the solutions.

Claim 3.1. Equation (3.1) can be written in compact vector-matrix notation as

$$J = (\boldsymbol{r} - \hat{\boldsymbol{y}})^T \boldsymbol{\Phi} (\boldsymbol{r} - \hat{\boldsymbol{y}}) + \Delta \boldsymbol{u}^T \boldsymbol{\Lambda} \Delta \boldsymbol{u} \tag{3.2}$$

where the augmented vectors $\hat{\boldsymbol{y}}$ and $\boldsymbol{r} \in \Re^{Ny}$ and $\boldsymbol{u} \in \Re^{Nu}$ are given by

$$\hat{\boldsymbol{y}} = \begin{bmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_m \end{bmatrix}, \; \boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \\ \vdots \\ \boldsymbol{r}_m \end{bmatrix}, \; \boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_p \end{bmatrix} \tag{3.3}$$

and where matrices $\boldsymbol{\Phi} \in \Re^{Ny \times Ny}$ and $\boldsymbol{\Lambda} \in \Re^{Nu \times Nu}$ are given by

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{\Phi}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{\Phi}_m \end{bmatrix} \qquad \boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_1 & 0 & \dots & 0 \\ 0 & \boldsymbol{\Lambda}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \boldsymbol{\Lambda}_p \end{bmatrix} \tag{3.4}$$

where in turn

$$\hat{\boldsymbol{y}}_j = \begin{bmatrix} \hat{y}_j(t+1|t) \\ \hat{y}_j(t+2|t) \\ \vdots \\ \hat{y}_j(t+Ny_j|t) \end{bmatrix}, \ \boldsymbol{r}_j = \begin{bmatrix} r_j(t+1) \\ r_j(t+2) \\ \vdots \\ r_j(t+Ny_j) \end{bmatrix}, \ \boldsymbol{u}_\ell = \begin{bmatrix} u_\ell(t) \\ u_\ell(t+1) \\ \vdots \\ u_\ell(t+Nu_\ell-1) \end{bmatrix}, \tag{3.5}$$

$$\boldsymbol{\Phi}_j = \begin{bmatrix} \phi_j(1) & 0 & \cdots & 0 \\ 0 & \phi_j(2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \phi_j(Ny_j) \end{bmatrix}, \qquad \boldsymbol{\Lambda}_\ell = \begin{bmatrix} \lambda_\ell(1) & 0 & \cdots & 0 \\ 0 & \lambda_\ell(2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_\ell(Nu_\ell) \end{bmatrix} \tag{3.6}$$

and $\boldsymbol{y}_j$ and $\boldsymbol{r}_j \in \Re^{Ny_j}$, $\boldsymbol{u}_\ell \in \Re^{Nu_\ell}$, $\boldsymbol{\Phi}_j \in \Re^{Ny_j \times Ny_j}$ and $\boldsymbol{\Lambda}_\ell \in \Re^{Nu_\ell \times Nu_\ell}$.

Proof. From definitions (3.5) to (3.6), the summations in (3.1) can be written as

$$\sum_{i=1}^{Ny_1} \phi_1(i)(r_1(t+i) - \hat{y}_1(t+i|t))^2 = (\boldsymbol{r}_1 - \hat{\boldsymbol{y}}_1)^T \boldsymbol{\Phi}_1 (\boldsymbol{r}_1 - \hat{\boldsymbol{y}}_1)$$

$$\sum_{i=1}^{Ny_2} \phi_2(i)(r_2(t+i) - \hat{y}_2(t+i|t))^2 = (\boldsymbol{r}_2 - \hat{\boldsymbol{y}}_2)^T \boldsymbol{\Phi}_2 (\boldsymbol{r}_2 - \hat{\boldsymbol{y}}_2)$$

$$\vdots$$

$$\sum_{i=1}^{Ny_m} \phi_m(i)(r_m(t+i) - y_m(t+i|t))^2 = (\boldsymbol{r}_m - \hat{\boldsymbol{y}}_m)^T \boldsymbol{\Phi}_m (\boldsymbol{r}_m - \hat{\boldsymbol{y}}_m)$$

and

$$\sum_{i=0}^{Nu_1} \lambda_1(i)\Delta u_1(t+i)^2 = \Delta \boldsymbol{u}_1^T \boldsymbol{\Lambda}_1 \Delta \boldsymbol{u}_1$$

$$\sum_{i=0}^{Nu_2} \lambda_2(i)\Delta u_2(t+i)^2 = \Delta \boldsymbol{u}_2^T \boldsymbol{\Lambda}_2 \Delta \boldsymbol{u}_2$$

$$\vdots$$

$$\sum_{i=0}^{Nu_p} \lambda_p(i)\Delta u_p(t+i)^2 = \Delta \boldsymbol{u}_p^T \boldsymbol{\Lambda}_p \Delta \boldsymbol{u}_p$$

substituting the above equations in (3.1) and using the definitions of vectors and matrices in (3.3) to (3.4) it follows that the objective function equation can be written as $J = (\boldsymbol{r} - \hat{\boldsymbol{y}})^T \boldsymbol{\Phi} (\boldsymbol{r} - \hat{\boldsymbol{y}}) + \Delta \boldsymbol{u}^T \boldsymbol{\Lambda} \Delta \boldsymbol{u}$. *Q.E.D.*

3.2 Derivation of the Predictive Control Law

The problem of minimizing J with relationship to $\Delta \boldsymbol{u}$ requires an expression for the predicted outputs as a function of future inputs.

Claim 3.2. Future values of the output, $\boldsymbol{y}$, are the sum of the constant forcing response, $\boldsymbol{y}^0$, and the response to future inputs $\boldsymbol{G}\Delta \boldsymbol{u}$:

$$\hat{\boldsymbol{y}} = \boldsymbol{y}^0 + \boldsymbol{G}\Delta \boldsymbol{u} \tag{3.7}$$

where

$$\boldsymbol{y}^0 = \begin{bmatrix} \boldsymbol{y}_1^0 \\ \boldsymbol{y}_2^0 \\ \vdots \\ \boldsymbol{y}_m^0 \end{bmatrix} \qquad \boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_{11} & \boldsymbol{G}_{12} & \cdots & \boldsymbol{G}_{1p} \\ \boldsymbol{G}_{21} & \boldsymbol{G}_{22} & \cdots & \boldsymbol{G}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{G}_{m1} & \boldsymbol{G}_{m2} & \cdots & \boldsymbol{G}_{mp} \end{bmatrix}$$

where in turn

$$\boldsymbol{y}_j^0 = \begin{bmatrix} \boldsymbol{y}_j^0(t+1\,|\,t) \\ \boldsymbol{y}_j^0(t+2\,|\,t) \\ \vdots \\ \boldsymbol{y}_j^0(t+Ny_1\,|\,t) \end{bmatrix} \qquad \boldsymbol{G}_{j\ell} = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{Ny_j-1} & g_{Ny_j-2} & \cdots & g_{Ny_j-Nu_\ell} \end{bmatrix}_{j\ell}$$

and $\boldsymbol{G} \in \Re^{Ny \times Nu}$, $\boldsymbol{y}^0 \in \Re^{Ny}$, $\boldsymbol{y}_j^0 \in \Re^{Ny}$. Each *dynamic matrix* is truncated to the corresponding control and prediction horizons, i.e. $\boldsymbol{G}_{j\ell} \in \Re^{Ny_j \times Nu_\ell}$.

Proof Consider the future values of the first output of the system

$$\hat{\boldsymbol{y}}_1 = \boldsymbol{y}_1^0 + \boldsymbol{G}_{11}\Delta \boldsymbol{u}_1 + \boldsymbol{G}_{12}\Delta \boldsymbol{u}_2 + \ldots + \boldsymbol{G}_{1p}\Delta \boldsymbol{u}_p \tag{3.8}$$

where $\boldsymbol{y}_1^0$ is the constant forcing response and $\boldsymbol{G}_{1\ell}$ are the *dynamic matrices* relating input ℓ to output 1. Similar equations can be written for the remaining outputs of the MIMO system as follows

$$\hat{\boldsymbol{y}}_2 = \boldsymbol{y}_2^0 + \boldsymbol{G}_{21}\Delta\boldsymbol{u}_1 + \boldsymbol{G}_{22}\Delta\boldsymbol{u}_2 + \ldots + \boldsymbol{G}_{2p}\Delta\boldsymbol{u}_p$$

$$\vdots$$

$$\hat{\boldsymbol{y}}_m = \boldsymbol{y}_m^0 + \boldsymbol{G}_{m1}\Delta\boldsymbol{u}_1 + \boldsymbol{G}_{m2}\Delta\boldsymbol{u}_2 + \ldots + \boldsymbol{G}_{mp}\Delta\boldsymbol{u}_p$$

The above equations can be put together as

$$\begin{bmatrix} \hat{\boldsymbol{y}}_1 \\ \hat{\boldsymbol{y}}_2 \\ \vdots \\ \hat{\boldsymbol{y}}_m \end{bmatrix} = \begin{bmatrix} \boldsymbol{y}_1^0 \\ \boldsymbol{y}_2^0 \\ \vdots \\ \boldsymbol{y}_m^0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{G}_{11} & \boldsymbol{G}_{12} & \cdots & \boldsymbol{G}_{1p} \\ \boldsymbol{G}_{21} & \boldsymbol{G}_{22} & \cdots & \boldsymbol{G}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{G}_{m1} & \boldsymbol{G}_{m2} & \cdots & \boldsymbol{G}_{mp} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{u}_1 \\ \Delta\boldsymbol{u}_2 \\ \vdots \\ \Delta\boldsymbol{u}_p \end{bmatrix} \tag{3.9}$$

from where we conclude that

$$\hat{\boldsymbol{y}} = \boldsymbol{y}^0 + \boldsymbol{G}\Delta\boldsymbol{u} \qquad \textit{Q.E.D.}$$

This is the equation needed for an analytical solution of the objective function equation since it shows, explicitly, the influence of future inputs in the future outputs. The solution is a function of the future reference trajectory and the constant forcing response as seen in the next claim.

Claim 3.3. Given the objective function equation $J = (\boldsymbol{r} - \hat{\boldsymbol{y}})^T \boldsymbol{\Phi}(\boldsymbol{r} - \hat{\boldsymbol{y}}) + \Delta\boldsymbol{u}^T \boldsymbol{\Lambda}\Delta\boldsymbol{u}$ and the prediction equation for future outputs, $\hat{\boldsymbol{y}} = \boldsymbol{y}^0 + \boldsymbol{G}\Delta\boldsymbol{u}$, then the optimal control sequence is given by

$$\Delta\boldsymbol{u} = (\boldsymbol{G}^T \boldsymbol{\Phi}\boldsymbol{G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}(\boldsymbol{r} - \boldsymbol{y}^0) \tag{3.10}$$

Proof Substitute equation (3.7) in the objective function equation to get

$$J = (\boldsymbol{r} - \boldsymbol{y}^0 - \boldsymbol{G}\Delta\boldsymbol{u})^T \boldsymbol{\Phi}(\boldsymbol{r} - \boldsymbol{y}^0 - \boldsymbol{G}\Delta\boldsymbol{u}) + \Delta\boldsymbol{u}^T \boldsymbol{\Lambda}\Delta\boldsymbol{u}$$

Let $\boldsymbol{\Phi} = \boldsymbol{R}^T \boldsymbol{R}$

$$J = (\boldsymbol{r} - \boldsymbol{y}^0 - \boldsymbol{G}\Delta\boldsymbol{u})^T \boldsymbol{R}^T \boldsymbol{R}(\boldsymbol{r} - \boldsymbol{y}^0 - \boldsymbol{G}\Delta\boldsymbol{u}) + \Delta\boldsymbol{u}^T \boldsymbol{\Lambda}\Delta\boldsymbol{u}$$

then

$$J = (\boldsymbol{Rr} - \boldsymbol{Ry}^0 - \boldsymbol{RG}\Delta\boldsymbol{u})^T (\boldsymbol{Rr} - \boldsymbol{Ry}^0 - \boldsymbol{RG}\Delta\boldsymbol{u}) + \Delta\boldsymbol{u}^T \boldsymbol{\Lambda}\Delta\boldsymbol{u}$$

differentiating with respect to $\Delta\boldsymbol{u}$

$$\frac{\partial J}{\partial \Delta\boldsymbol{u}} = -2(\boldsymbol{Rr} - \boldsymbol{Ry}^0 - \boldsymbol{RG}\Delta\boldsymbol{u})^T \boldsymbol{RG} + 2\Delta\boldsymbol{u}^T \boldsymbol{\Lambda}$$

equating the derivative to zero in order to get the minimum of the function then

$$-2(\boldsymbol{Rr} - \boldsymbol{Ry}^0 - \boldsymbol{RG}\Delta\boldsymbol{u})^T \boldsymbol{RG} + 2\Delta\boldsymbol{u}^T \boldsymbol{\Lambda} = 0$$

and

$$\Delta\boldsymbol{u}^T \boldsymbol{\Lambda} = (\boldsymbol{r} - \boldsymbol{y}^0 - \boldsymbol{G}\Delta\boldsymbol{u})^T \boldsymbol{R}^T \boldsymbol{RG}$$

$$\Delta\boldsymbol{u}^T \boldsymbol{\Lambda} = (\boldsymbol{r} - \boldsymbol{y}^0)^T \boldsymbol{\Phi G} - \Delta\boldsymbol{u}^T \boldsymbol{G}^T \boldsymbol{\Phi G}$$

$$\Delta\boldsymbol{u}^T (\boldsymbol{\Lambda} + \boldsymbol{G}^T \boldsymbol{\Phi G}) = (\boldsymbol{r} - \boldsymbol{y}^0)^T \boldsymbol{\Phi G}$$

transposition yields

$$(\boldsymbol{\Lambda} + \boldsymbol{G}^T \boldsymbol{\Phi G})^T \Delta\boldsymbol{u} = \boldsymbol{G}^T \boldsymbol{\Phi}^T (\boldsymbol{r} - \boldsymbol{y}^0)$$

Note that $(\boldsymbol{\Lambda} + \boldsymbol{G}^T \boldsymbol{\Phi G})$ and $\boldsymbol{\Phi}$ are symmetric matrices (making the transposition irrelevant). Thus the optimal control sequence is given by

$$\Delta\boldsymbol{u} = (\boldsymbol{G}^T \boldsymbol{\Phi G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}(\boldsymbol{r} - \boldsymbol{y}^0) \qquad \textit{Q.E.D.}$$

From the receding horizon principle only the part corresponding to the first control move of matrix $(\boldsymbol{G}^T \boldsymbol{\Phi G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}$ is used. That way matrix $(\boldsymbol{G}^T \boldsymbol{\Phi G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}$ can be replaced by a smaller dimension matrix.

Claim 3.4. The control law for the multivariable predictive control is given by

$$\Delta\boldsymbol{u}(t) = \boldsymbol{K}(\boldsymbol{r} - \boldsymbol{y}^0) \tag{3.11}$$

where matrix $\boldsymbol{K} \in \Re^{pNy}$ is defined as

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{k}_{11}^T & \boldsymbol{k}_{12}^T & \cdots & \boldsymbol{k}_{1m}^T \\ \boldsymbol{k}_{21}^T & \boldsymbol{k}_{22}^T & \cdots & \boldsymbol{k}_{2m}^T \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{k}_{p1}^T & \boldsymbol{k}_{p2}^T & \cdots & \boldsymbol{k}_{pm}^T \end{bmatrix} \tag{3.12}$$

and its elements

$$\boldsymbol{k}_{\ell j}^T = [\, k_1 \quad k_2 \quad \cdots \quad k_{Ny_j} \,]_{\ell j} \tag{3.13}$$

are the rows of $(\boldsymbol{G}^T \boldsymbol{\Phi} \boldsymbol{G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}$ corresponding to the first control move.

See Appendix E for the proof.

3.3 Derivation of the Constant Forcing Function

In order to solve the control law equation, $\boldsymbol{u} = \boldsymbol{K}(\boldsymbol{r} - \boldsymbol{y}^0)$, an expression for future values of the constant forcing response, $\boldsymbol{y}^0$, is needed. This can be done via the solution of a series of Diophantine equations. For SISO cases the developments are widely documented in the literature (Clarke *et al.*, 1987; Clarke and Mohtade, 1989; Crisalle *et al.*, 1989). The derivation for the MIMO systems will be done here. The approach consists of first extending the SISO results to the MISO plants, and then applying the MISO approach to each of the outputs of the MIMO plant. The individual MISO predictors are then stacked into augmented vectors and matrices as needed to build the final MIMO operators sought, as was done in section 3.2 in the derivation of the predictive control law. The many different equations for the MIMO case make the derivation quite cumbersome and only the vectorial equations will be shown here while a detailed derivation is found in Appendix A. The definitions for the terms of the equations that follow will not be repeated here and the reader should refer to the appendix whenever necessary.

Consider the multivariable system defined by the DARMA model of equation (1.1), $\boldsymbol{A}(q^{-1})\boldsymbol{y}(t)=\boldsymbol{B}(q^{-1})\boldsymbol{u}(t-1)$. The predicted values of the output can be generated using *Matrix Diophantine* equation,

$$\boldsymbol{E}_i(q^{-1})\Delta\boldsymbol{A}(q^{-1})+\boldsymbol{q}^{-i}\boldsymbol{F}_i(q^{-1})=\boldsymbol{I}_i \tag{A.9}$$

where $\boldsymbol{E}_i(q^{-1})$ and $\boldsymbol{F}_i(q^{-1})$ are unique polynomial matrices defined in Appendix A. For that, multiply equation (1.1) by $\boldsymbol{E}_i(q^{-1})\Delta$ to get

$$\boldsymbol{E}_i(q^{-1})\Delta\boldsymbol{A}(q^{-1})\boldsymbol{y}(t)=\boldsymbol{E}_i(q^{-1})\boldsymbol{B}(q^{-1})\Delta\boldsymbol{u}(t-1)$$

From *Matrix Diophantine* equation we see that $\boldsymbol{E}_i(q^{-1})\Delta\boldsymbol{A}(q^{-1})=\boldsymbol{I}_i-\boldsymbol{q}^{-i}\boldsymbol{F}_i(q^{-1})$ and consequently

$$\left[\boldsymbol{I}_i-\boldsymbol{q}^{-i}\boldsymbol{F}_i(q^{-1})\right]\boldsymbol{y}(t)=\boldsymbol{E}_i(q^{-1})\boldsymbol{B}(q^{-1})\Delta\boldsymbol{u}(t-1) \tag{3.14}$$

Simple manipulation leads to $\boldsymbol{y}(t)=\boldsymbol{E}_i(q^{-1})\boldsymbol{B}(q^{-1})\Delta\boldsymbol{u}(t-1)+\boldsymbol{q}^{-i}\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t)$. Multiply by $\boldsymbol{q}^{i}$ to get an expression for predicted values of the output

$$\hat{\boldsymbol{y}}(t+i)=\boldsymbol{E}_i(q^{-1})\boldsymbol{B}(q^{-1})\Delta\boldsymbol{u}(t+i-1)+\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t) \tag{3.15}$$

Recognizing that

$$\boldsymbol{E}_i(q^{-1})\boldsymbol{B}(q^{-1})=\boldsymbol{G}_i(q^{-1})+\boldsymbol{q}^{-i}\boldsymbol{P}_i(q^{-1}) \tag{3.16}$$

then

$$\hat{\boldsymbol{y}}(t+i)=\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t)+\boldsymbol{G}_i(q^{-1})\Delta\boldsymbol{u}(t+i-1)+q^{-1}\boldsymbol{P}_i(q^{-1})\Delta\boldsymbol{u}(t) \tag{3.17}$$

Finally, verifying that the predicted values of the output, $\hat{\boldsymbol{y}}(t+i)$, are the unforced response of the system, $\boldsymbol{y}^0(t+i)$, when future values of the input are kept constant, i.e. $\Delta\boldsymbol{u}(t+i-1)=0$ for $i>0$, then

$$\boldsymbol{y}^0(t+i)=\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t)+q^{-1}\boldsymbol{P}_i(q^{-1})\Delta\boldsymbol{u}(t) \tag{3.18}$$

or if we consider the whole prediction and control horizons

$$\boldsymbol{y}^0=\boldsymbol{F}(q^{-1})\boldsymbol{y}(t)+q^{-1}\boldsymbol{P}(q^{-1})\Delta\boldsymbol{u}(t)$$

3.4 Final Control Law Equation

The control law equation (3.11) can be written, according to the notation adopted, in terms of a hypothetical sample time *t* + *i* as

$$\Delta\boldsymbol{u}(t)=\boldsymbol{K}_i\left[\boldsymbol{r}(t+i)-\boldsymbol{y}^0(t+i)\right]$$

and substituting the constant forcing function of equation (3.18) then

$$\Delta\boldsymbol{u}(t)=\boldsymbol{K}_i\left[\boldsymbol{r}(t+i)-\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t)-q^{-1}\boldsymbol{P}_i(q^{-1})\Delta\boldsymbol{u}(t)\right]$$

which can be rewritten as

$$\left[\boldsymbol{I}+q^{-1}\boldsymbol{K}_i\boldsymbol{P}_i(q^{-1})\right]\Delta\boldsymbol{u}(t)=\boldsymbol{K}_i\boldsymbol{q}^i\boldsymbol{r}(t)-\boldsymbol{K}_i\boldsymbol{F}_i(q^{-1})\boldsymbol{y}(t)$$

Define

$$\boldsymbol{R}_i(q^{-1})=\left[\boldsymbol{I}+q^{-1}\boldsymbol{K}_i\boldsymbol{P}_i(q^{-1})\right]\Delta \tag{3.19}$$

$$\boldsymbol{S}_i(q^{-1})=\boldsymbol{K}_i\boldsymbol{F}_i(q^{-1}) \tag{3.20}$$

$$\boldsymbol{T}_i(q)=\boldsymbol{K}_i(q)\boldsymbol{q}^i \tag{3.21}$$

to get

$$\boldsymbol{R}_i(q^{-1})\boldsymbol{u}(t)=\boldsymbol{T}_i(q)\boldsymbol{r}(t)-\boldsymbol{S}_i(q^{-1})\boldsymbol{y}(t) \tag{3.22}$$

Considering the prediction horizons it yields

$$\boldsymbol{R}(q^{-1})\boldsymbol{u}(t) = \boldsymbol{T}(q)\boldsymbol{r}(t) - \boldsymbol{S}(q^{-1})\boldsymbol{y}(t) \tag{3.23}$$

Note that $\boldsymbol{R}(q^{-1})$ and $\boldsymbol{S}(q^{-1})$ are polynomial matrices in backward shift-operator while matrix $\boldsymbol{T}(q)$ is in forward shift-operator.

A Matlab code was created to generate matrices $\boldsymbol{R}(q^{-1})$, $\boldsymbol{S}(q^{-1})$ and $\boldsymbol{T}(q)$. Once the set of parameters is specified, namely prediction horizons, control horizons, weights λ and ϕ, then matrices $\boldsymbol{R}(q^{-1})$, $\boldsymbol{S}(q^{-1})$ and $\boldsymbol{T}(q)$ are generated for a given linear model of a process plant.

3.5 Conclusion

In this chapter a concise mathematical development of the control law for the multivariable predictive control is shown. The appropriate definition of vectors and matrices allowed a mathematical formalism similar to the SISO case and a design procedure is derived from it. A Matlab code has been developed for the design of multivariable unconstrained predictive controllers.

CHAPTER 4
ANALYSIS OF CLOSED-LOOP BEHAVIOR

In this chapter the closed-loop system is analyzed for stability and performance. For that a practical method to determine the closed-loop poles of multivariable predictive control is searched. Proof of properties known to be valid for the SISO case are pursued for the multivariable case.

4.1 Stability Analysis of Closed-loop Multivariable Systems

The issue of ensuring stability of a closed-loop feedback controller is of utmost importance to control system design. Albeit the increasing popularity of predictive control over the last decades a general method for the analysis of stability of the resulting closed-loop is still not available for multivariable systems. Although unconstrained systems present analytical solutions to the problem of minimizing the objective function, obtaining the closed-loop transfer matrices is a major challenge. Numerical difficulties in inverting transfer matrices have limited the size of systems for which the poles can be calculated.

A stable system is a dynamic system with a bounded response to a bounded input. Many descriptions of stability exist, namely external stability, internal stability, marginal stability and exponential stability. In this dissertation external (also known as BIBO for bounded input bounded output) stability is analyzed.

During the design phase of predictive control no *a priori* guarantee of stability exists for the resulting closed-loop system. Stability of a system is determined by the position of its poles in the complex plane, which can be determined, among other ways, by analyzing the individual transfer functions of the closed-loop matrix. Matrix fraction descriptions can also be used to decompose rational matrices into 'numerator' and 'denominator' matrices allowing for the determination of poles and zeros of multivariable systems. Under certain circumstances the latter method proves to be better.

Since the aim of this chapter is to study stability of the closed-loop system, in the sequel, the complex variable z is substituted for the forward shift operator q. This substitution is allowed by the analogy between shift-operator theory and transfer function theory for linear systems with zero initial conditions. Hence matrix $\boldsymbol{R}(z^{-1})$ is substituted for matrix $\boldsymbol{R}(q^{-1})$ and so on. Furthermore, for simplicity of notation, the arguments z^{-1} and z are omitted whenever the interpretation of the transfer function in question is deemed unambiguous.

4.2 Closed-loop Transfer Function Matrices

Figure 4.1 shows the block diagram of the multivariable predictive control applied to a plant $\boldsymbol{H}(z)$.

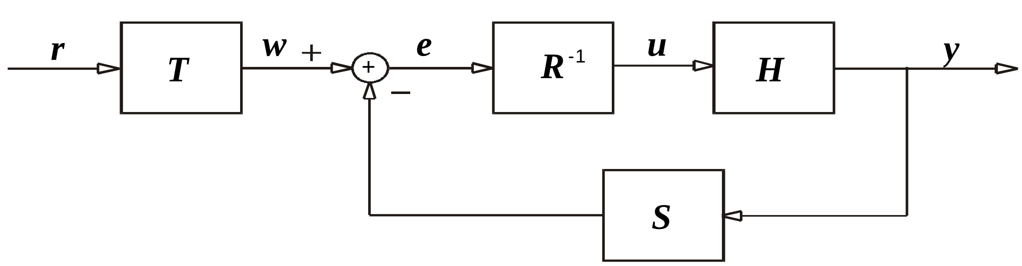

Figure 4.1 Closed-loop System

Let $\boldsymbol{H}(z)$ be the process plant. Using block-diagram algebra operations, the dynamics of the system shown in Figure 4.1 can be captured by the closed-loop transfer matrices

$$\boldsymbol{G}_{ur}(z) = \left[\boldsymbol{R}(z) + \boldsymbol{S}(z)\boldsymbol{H}(z)\right]^{-1}\boldsymbol{T}(z) \tag{4.1}$$

and

$$\boldsymbol{G}_{yr}(z) = \boldsymbol{H}(z)\left[\boldsymbol{R}(z) + \boldsymbol{S}(z)\boldsymbol{H}(z)\right]^{-1}\boldsymbol{T}(z) \tag{4.2}$$

Transfer function matrix $\boldsymbol{T}(z)$ is a polynomial matrix applied to the future reference trajectory. Hence $\boldsymbol{w}(z)$ is a bounded signal for a bounded reference signal $\boldsymbol{r}(z)$ and if all possible disturbances and noises are considered then it is seen (refer to Appendix D) that for the analysis of stability it suffices to analyze the transfer functions relating the control signal $\boldsymbol{u}(z)$ to $\boldsymbol{w}(z)$ and the output $\boldsymbol{y}(z)$ to $\boldsymbol{w}(z)$; $\boldsymbol{G}_{uw}(z)$ and $\boldsymbol{G}_{yw}(z)$ respectively.

$$\boldsymbol{G}_{uw}(z) = \left[\boldsymbol{R}(z) + \boldsymbol{S}(z)\boldsymbol{H}(z)\right]^{-1} \tag{4.3}$$

and

$$\boldsymbol{G}_{yw}(z) = \boldsymbol{H}(z)\left[\boldsymbol{R}(z) + \boldsymbol{S}(z)\boldsymbol{H}(z)\right]^{-1} \tag{4.4}$$

THEOREM 4.1 Transfer function matrix $\boldsymbol{R}(z)$ is biproper.

Proof Recall that a proper transfer matrix whose inverse is also proper is called biproper. From the definitions of $\boldsymbol{K}$ and $\boldsymbol{P}(z)$ its immediate to verify that the product $z^{-1}\boldsymbol{KP}(z)$ is a proper transfer matrix. The addition of the identity matrix to it, $\boldsymbol{I} + z^{-1}\boldsymbol{KP}(z)$, causes the determinant to be biproper and as a consequence $\boldsymbol{R}(z) = \left[\boldsymbol{I} + z^{-1}\boldsymbol{KP}(z)\right]\Delta$ is a biproper transfer matrix. *Q.E.D.*

This property of $\boldsymbol{R}(z)$ makes $\boldsymbol{R}(z)^{-1}$ biproper and allows the implementation of the controller.

THEOREM 4.2 Let $\boldsymbol{H}(z)$ be a proper transfer matrix, then $\boldsymbol{G}_{uw}(z)$ is biproper and $\boldsymbol{G}_{yw}(z)$ is proper.

Proof Matrix $\boldsymbol{S}(z)$ is the product of $\boldsymbol{K}$ and $\boldsymbol{F}(z)$. From the definitions of $\boldsymbol{K}$ and $\boldsymbol{F}(z)$ it is concluded that $\boldsymbol{S}(z)$ is a proper transfer matrix. Once $\boldsymbol{H}(z)$ is proper then the product $\boldsymbol{S}(z)\boldsymbol{H}(z)$ is also proper. Consequently the sum $\boldsymbol{R}(z)+\boldsymbol{S}(z)\boldsymbol{H}(z)$ is biproper since $\boldsymbol{R}(z)$ is, according to Theorem 4.1, biproper. Thus $\boldsymbol{G}_{uw}(z)$ is biproper. Bearing in mind that $\boldsymbol{G}_{yw}(z)=\boldsymbol{H}(z)\boldsymbol{G}_{uw}(z)$ then it is concluded that $\boldsymbol{G}_{yw}(z)$ is proper. *Q.E.D.*

4.3 MFD of Transfer Function Matrices

A transfer matrix may be represented as the "ratio" of two polynomial matrices as

$$\boldsymbol{G}(z)=\boldsymbol{N}_R(z)\boldsymbol{D}_R^{-1}(z)=\boldsymbol{D}_L^{-1}(z)\boldsymbol{N}_L(z)$$

where both matrices have the same number of columns. For an $m\times p$ matrix $\boldsymbol{G}(z)$ then $\boldsymbol{D}(z)$ is $m\times m$ and $\boldsymbol{N}(z)$ is $p\times m$. The first representation is a right matrix fraction description, MFD, while the second is called a left MFD of $\boldsymbol{G}(z)$. Since duality exists between these descriptions, for the most part, only right MFDs will be mentioned hereon. The subscript R shall be omitted and restored only when necessary.

Example 4.1 Consider

$$\boldsymbol{G}(z)=\begin{bmatrix}\dfrac{1}{z} & \dfrac{0.5}{z}\\ \dfrac{0.04}{z} & \dfrac{0.05}{z}\end{bmatrix}$$

which can be decomposed in a right MFD as:

$$\boldsymbol{N}(z)=\begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.05 \end{bmatrix} \quad \boldsymbol{D}(z)=\begin{bmatrix} z(z+2.5) & 0 \\ -2z & z \end{bmatrix} \qquad \blacksquare$$

An MFD is not unique and an infinity of others can be obtained by choosing any non-singular polynomial $\boldsymbol{W}(z)$ such that

$$\bar{\boldsymbol{N}}(z)=\boldsymbol{N}(z)\boldsymbol{W}^{-1}(z) \quad \bar{\boldsymbol{D}}(z)=\boldsymbol{D}(z)\boldsymbol{W}^{-1}(z)$$

are polynomial matrices. Then

$$\boldsymbol{G}(z)=\boldsymbol{N}(z)\boldsymbol{D}^{-1}(z)=\bar{\boldsymbol{N}}(z)\bar{\boldsymbol{D}}^{-1}(z)$$

since

$$\boldsymbol{N}(z)=\bar{\boldsymbol{N}}(z)\boldsymbol{W}(z) \quad \boldsymbol{D}(z)=\bar{\boldsymbol{D}}(z)\boldsymbol{W}(z)$$

Matrix $\boldsymbol{W}(z)$ is right divisor of $\boldsymbol{N}(z)$ and $\boldsymbol{D}(z)$.

Some polynomial matrices have inverses that are also polynomial. These are called unimodular and possess a determinant that is independent of z (a nonzero constant). When all right divisors of two matrices are unimodular these matrices are called right coprime and the MFD is called irreducible. Analogous to the scalar case, coprimeness of two polynomial matrices means that no cancellation of common terms is possible.

DEFINITION 4.1 A right MFD $\boldsymbol{G}(z)=\boldsymbol{N}(z)\boldsymbol{D}^{-1}(z)$ is called irreducible if $\boldsymbol{N}(z)$ and $\boldsymbol{D}(z)$ are right coprime. ■

No direct method for the generation of irreducible MFDs is known. The MFD should be tested for coprimeness and, if necessary, both matrices should be "divided" by their greatest common divisor. Bezout Identity can be used to determine if two matrices are coprime as $\boldsymbol{N}(z)$ and $\boldsymbol{D}(z)$ will be right coprime if and only if there exist polynomial matrices $\boldsymbol{X}(z)$ and $\boldsymbol{Y}(z)$ such that

$$\boldsymbol{X}(z)\boldsymbol{N}(z)+\boldsymbol{Y}(z)\boldsymbol{D}(z)=\boldsymbol{I}$$

The construction of a greatest common divisor is done by finding a unimodular matrix $\boldsymbol{U}(z)$ such that at least p of the bottom rows on the right hand side are identically zero:

$$\begin{bmatrix} \boldsymbol{U}_{11}(z) & \boldsymbol{U}_{12}(z) \\ \boldsymbol{U}_{21}(z) & \boldsymbol{U}_{22}(z) \end{bmatrix} \begin{bmatrix} \boldsymbol{D}(z) \\ \boldsymbol{N}(z) \end{bmatrix} = \begin{bmatrix} \boldsymbol{C}(z) \\ 0 \end{bmatrix}$$

Proof of this is done by Kaylath.

Example 4.2 Consider the MFD of Example 4.1. Matrices $\boldsymbol{D}(z)$ and $\boldsymbol{N}(z)$ are not coprime. Their greatest common divisor is

$$\boldsymbol{C}(z)=\begin{bmatrix} 1 & -0.5 \\ 0 & z+2.5 \end{bmatrix}$$

Make $\bar{\boldsymbol{N}} = \boldsymbol{N}\boldsymbol{C}^{-1}$ and $\bar{\boldsymbol{D}} = \boldsymbol{D}\boldsymbol{C}^{-1}$ then

$$\bar{\boldsymbol{N}}(z) = \boldsymbol{N}(z)\boldsymbol{C}^{-1}(z) = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.02 \end{bmatrix}$$

$$\bar{\boldsymbol{D}}(z) = \boldsymbol{D}(z)\boldsymbol{C}^{-1}(z) = \begin{bmatrix} z(z+2.5) & 0.5z \\ -2z & 0 \end{bmatrix}$$

which are now coprime. Note that this new MFD still represents $\boldsymbol{G}(z)$ correctly.

$$\boldsymbol{G}(z) = \bar{\boldsymbol{N}}(z)\bar{\boldsymbol{D}}^{-1}(z) = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.02 \end{bmatrix} \begin{bmatrix} z(z+2.5) & 0.5z \\ -2z & 0 \end{bmatrix}^{-1} = \begin{bmatrix} \dfrac{1}{z} & \dfrac{0.5}{z} \\ \dfrac{0.04}{z} & \dfrac{0.05}{z} \end{bmatrix} \qquad \blacksquare$$

4.4 Poles and Zeros of a Transfer Matrix

Poles and zeros of transfer matrices can be defined and determined in several ways one of which consists in decomposing the transfer matrix into a canonical form known as the Smith-McMillan form. Every rational transfer matrix $\boldsymbol{G}(z)$ can be reduced to its Smith-McMillan form $\boldsymbol{M}(z)$ as

$$\boldsymbol{M}(z) = diag\left\{\frac{\varepsilon_1(z)}{\psi_1(z)}, \frac{\varepsilon_2(z)}{\psi_2(z)}, \ldots, \frac{\varepsilon_r}{\psi_r(z)}, 0, \ldots, 0\right\}$$

where $\varepsilon_i(z)$ are called the invariant factors of $\boldsymbol{M}(z)$. Each pair $\{\varepsilon_i(z), \psi_i(z)\}$ is coprime. Note that multivariable systems can have poles and zeros at the same location once ε_i and ψ_j, for $i \neq j$, do not have to be coprime. Multiplication by unimodular matrices does not affect the basic properties of transfer matrices. Therefore matrices that differ only by unimodular factor, called equivalent matrices, have the same Smith-McMillan form.

Example 4.3 Consider a process $\boldsymbol{H}(z)$

$$\boldsymbol{H}(z) = \begin{bmatrix} \dfrac{z+1.5}{z(z-1)} & \dfrac{0.5}{(z-1)} \\ \dfrac{0.04}{(z-1)} & \dfrac{0.05}{(z-1)} \end{bmatrix}$$

The Smith-McMillan form of $\boldsymbol{H}(z)$ is given by

$$\boldsymbol{M}(z) = \begin{bmatrix} \dfrac{1}{z(z-1)} & 0 \\ 0 & \dfrac{z+2.5}{z-1} \end{bmatrix}$$

from where we conclude that the poles and zeros are $p = \{0,1,1\}$ and $z = \{-2.5\}$. The relative position of poles and zeros is seen along the diagonal of $\boldsymbol{M}(z)$. Thus it is best to rewrite the Smith-McMillan form as

$$\boldsymbol{M}(z) = diag\left\{\frac{1}{z(z-1)}, \frac{z+2.5}{z-1}\right\} \qquad \blacksquare$$

The Smith-McMillan form tells us more than we need to know. For the analysis of stability the relative position of poles and zeros, unless cancellations occur, is irrelevant. Ultimately it is the presence of unstable poles in any of the closed-loop transfer matrices that will determine instability. The Smith-McMillan form is also very elaborate computationally and not practical for real systems.

Another way of determining the poles of a system is by direct verification of the poles of the individual transfer functions in the transfer matrix. This, however, requires the calculation of the inverse of $\boldsymbol{R}+\boldsymbol{SH}$ which is, in general, a computationally difficult task. The round off errors associated to the many different convolutions, required during the process of inversion of polynomial matrices, makes the cancellation of common terms unfeasible. Even for systems of modest sizes, e.g. systems with two inputs and two outputs, the necessary cancellations, in general, do not occur. The use of Matlab commands, such as *minreal*, which reduces a realization to a minimal realization, solved the problem partially but cannot be considered a reliable procedure. Thus, from a practical point of view, a solution to the design problem that would avoid or reduce the number of inversion of matrices is preferred.

Matrix fraction descriptions may be used, instead, to determine poles of multivariable systems and are more suitable for computational purposes.

PROPERTY 4.1 Let $\mathbb{C}$ denote the complex plane. Let $\boldsymbol{G}(z)$ be a proper transfer function matrix and $\boldsymbol{N}(z)$, $\boldsymbol{D}(z)$ be an irreducible right MFD of $\boldsymbol{G}(z)$ such that

$$\boldsymbol{G}(z)=\boldsymbol{N}(z)\boldsymbol{D}^{-1}(z)$$

where $\boldsymbol{D}(z)$ is nonsingular. Then $p \in \mathbb{C}$ is a pole of $\boldsymbol{G}(z)$ if and only if p is a root of the determinant of $\boldsymbol{D}(z)$. ■

For square transfer matrices the zeros can also be determined from the MFD of $\boldsymbol{G}(z)$ as they are the roots of the determinant of $\boldsymbol{N}(z)$. For non-square systems, however, the concept of zeros is unclear.

The closed-loop transfer matrices can be represented as MFDs. Let $\boldsymbol{H} = \boldsymbol{B}\boldsymbol{A}^{-1}$ be a right MFD of $\boldsymbol{H}$. Then $\boldsymbol{G}_{uw} = (\boldsymbol{R} + \boldsymbol{S}\boldsymbol{B}\boldsymbol{A}^{-1})^{-1}$ and $\boldsymbol{G}_{uw} = \boldsymbol{A}(\boldsymbol{R}\boldsymbol{A} + \boldsymbol{S}\boldsymbol{B})^{-1}$. Let

$$\boldsymbol{\Omega} = \boldsymbol{R}\boldsymbol{A} + \boldsymbol{S}\boldsymbol{B} \tag{4.5}$$

then

$$\boldsymbol{G}_{uw} = \boldsymbol{A}\boldsymbol{\Omega}^{-1} \tag{4.6}$$

the same substitutions in $\boldsymbol{G}_{yw}$ give $\boldsymbol{G}_{yw} = \boldsymbol{B}\boldsymbol{A}^{-1}(\boldsymbol{R} + \boldsymbol{S}\boldsymbol{B}\boldsymbol{A}^{-1})^{-1}$ and $\boldsymbol{G}_{yw} = \boldsymbol{B}(\boldsymbol{R}\boldsymbol{A} + \boldsymbol{S}\boldsymbol{B})^{-1}$. Finally

$$\boldsymbol{G}_{yw} = \boldsymbol{B}\boldsymbol{\Omega}^{-1} \tag{4.7}$$

Thus in order to be able to determine the poles of the closed-loop system the right MFDs of $\boldsymbol{G}_{uw}$ and $\boldsymbol{G}_{yw}$ need to be reduced to lowest terms. For that, let $\boldsymbol{C}_{uw}$ be the greatest right common divisor of $\boldsymbol{A}$ and $\boldsymbol{\Omega}$, i.e. $\boldsymbol{C}_{uw} = gcrd(\boldsymbol{A}, \boldsymbol{\Omega})$, then $\boldsymbol{A} = \bar{\boldsymbol{A}}_{uw}\boldsymbol{C}_{uw}$ and $\boldsymbol{\Omega} = \bar{\boldsymbol{\Omega}}_{uw}\boldsymbol{C}_{uw}$. Substituting in equation (4.6) we get

$$\boldsymbol{G}_{uw} = \bar{\boldsymbol{A}}_{uw}\bar{\boldsymbol{\Omega}}_{uw}^{-1} \tag{4.8}$$

Similarly let $\boldsymbol{C}_{yw} = gcrd(\boldsymbol{B}, \boldsymbol{\Omega})$, then $\boldsymbol{B} = \bar{\boldsymbol{B}}_{yw}\boldsymbol{C}_{yw}$, $\boldsymbol{\Omega} = \bar{\boldsymbol{\Omega}}_{yw}\boldsymbol{C}_{yw}$ and

$$\boldsymbol{G}_{yw} = \bar{\boldsymbol{B}}_{yw}\bar{\boldsymbol{\Omega}}_{yw}^{-1} \tag{4.9}$$

We now have the necessary conditions to determine the poles of the closed-loop transfer matrices.

THEOREM 4.3 Consider the closed-loop system of Figure 4.1 with $\boldsymbol{H}(z)$ proper. Then p_y is a pole of $\boldsymbol{G}_{yw}(z)$ if and only if p_y is a root of the determinant of $\bar{\boldsymbol{\Omega}}_{yw}(z)$ and p_u is a pole of $\boldsymbol{G}_{uw}(z)$ if and only if p_u is a root of the determinant of $\bar{\boldsymbol{\Omega}}_{uw}(z)$. If $\boldsymbol{H}(z)$ is square and nonsingular then z_y is a zero of $\boldsymbol{G}_{yw}(z)$ if and only if z_y is a root of the determinant of $\bar{\boldsymbol{B}}_{yw}(z)$ and z_u is a zero of $\boldsymbol{G}_{uw}(z)$ if and only if z_u is a root of the determinant of $\bar{\boldsymbol{A}}_{uw}(z)$.

Proof Proof follows from Property 4.1. *Q.E.D.*

Note that the determination of the poles $\boldsymbol{G}_{uw}(z)$ or $\boldsymbol{G}_{yw}(z)$ does not require the inversion of $\bar{\boldsymbol{\Omega}}_{uw}(z)$ or $\bar{\boldsymbol{\Omega}}_{yw}(z)$. Instead we only need to calculate their determinants.

Example 4.4 Consider a process $\boldsymbol{H}(z)$

$$\boldsymbol{H}(z) = \begin{bmatrix} \dfrac{z+1.5}{z(z-1)} & \dfrac{0.5}{(z-1)} \\ \dfrac{0.04}{(z-1)} & \dfrac{0.05}{(z-1)} \end{bmatrix}$$

with irreducible right MFD $\boldsymbol{H}(z) = \boldsymbol{B}(z)\boldsymbol{A}^{-1}(z)$

$$\boldsymbol{B}(z) = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.05 \end{bmatrix} \quad \boldsymbol{A}(z) = \begin{bmatrix} z(z-1) & 0 \\ 0 & (z-1) \end{bmatrix}$$

let the predictive controller designed for this plant be such that $\boldsymbol{\Omega}(z)$ is

$$\boldsymbol{\Omega}(z) = \begin{bmatrix} z(z+2.5) & 0 \\ -2z & z \end{bmatrix}$$

The roots of the determinants of $\boldsymbol{B}(z)$, $\boldsymbol{A}(z)$ and $\boldsymbol{\Omega}(z)$ are {-2.5}, {0, 1, 1} and {-2.5, 0, 0,} respectively. Thus $\boldsymbol{G}_{uw}(z)$ is unstable once its zeros, {0, 1, 1}, cannot cancel the unstable pole of $\mathbf{\Omega}(z)$ located at –2.5. Refer to examples 1 and 2 to see that matrices $\boldsymbol{B}(z)$ and $\mathbf{\Omega}(z)$ are not coprime.

Matrix $\mathbf{\Omega}(z^{-1})$ is always polynomial when expressed in backward shift-operators. In forward shift-operators, however, it will normally be a rational matrix where the denominators are powers of *z*. Because the poles of a system are defined in forward shift-operators, this conversion must be executed before the poles are calculated. Multiplication by a diagonal matrix with appropriate powers of *z* will do this transformation but the numerator matrix must be considered also so that no poles at the origin are lost. Although the backward and forward shift-operator versions of a transfer function are mathematically equivalent changing the arguments may be misleading. Therefore the symbol $\widehat{\boldsymbol{G}}(z)$ will be used to refer to the forward shift-operator version of $\boldsymbol{G}(z^{-1})$.

DEFINITION 4.2 The conjugate of a polynomial $M(z^{-1}) = m_0 + m_1 z^{-1} + \ldots + m_{nM} z^{-nM}$ in backward shift-operator is defined as $\tilde{M}(z) = M(z^{-1}) z^{nM} = m_0 z^{nM} + m_1 z^{nM-1} + \ldots + m_{nM}$. The degree of the polynomial in backward shift-operator is the degree of its conjugate polynomial. ■

DEFINITION 4.3 The conjugate of a polynomial matrix $\boldsymbol{M}(z^{-1})$ in backward shift-operator is defined as $\tilde{\boldsymbol{M}}(z) = \boldsymbol{M}(z^{-1}) \boldsymbol{z}^{\boldsymbol{nM}}$ where $\boldsymbol{z}^{\boldsymbol{nM}} = diag\{z^{\boldsymbol{nM}_1}, z^{\boldsymbol{nM}_2}, \ldots, z^{\boldsymbol{nM}_m}\}$ is a diagonal matrix and $z^{\boldsymbol{nM}_j}$ is defined as the maximum degree of all polynomials along the *j*th column of $\boldsymbol{M}(z^{-1})$ ■

Note that for matrices with polynomials of different degree along its columns this procedure places excess *z*s in the lower degree polynomials. This is not the case of

$\boldsymbol{\Omega}(z^{-1})$ where degree of all polynomials along its columns are the same. This can be seen from the definitions of ***R***, ***A***, ***S*** and ***B***. Thus $\boldsymbol{\Omega}(z^{-1})$ possesses a conjugate matrix $\tilde{\boldsymbol{\Omega}}(z) = \boldsymbol{\Omega}(z^{-1})\boldsymbol{z}^{n\Omega}$ where no excess *z*s are present.

When transforming backward shift-operators MFDs into forward shift-operators MFDs the respective $\boldsymbol{z}^{n\boldsymbol{M}}$ of numerator and denominator matrices must be reduced to lowest terms to guarantee coprimeness.

DEFINITION 4.4 The degree of a transfer function

$$\frac{B(z^{-1})}{A(z^{-1})} = \frac{b_0 + b_1 z^{-1} + \ldots + b_{nB} z^{-nB}}{a_0 + a_1 z^{-1} + \ldots + a_{nA} z^{-nA}}$$

is the $\max\{nB, nA\}$. ■

DEFINITION 4.5 The conjugate of a right MFD $\boldsymbol{H}(z^{-1}) = \boldsymbol{B}(z^{-1})\boldsymbol{A}^{-1}(z^{-1})$ is $\tilde{\boldsymbol{H}}(z) = \tilde{\boldsymbol{B}}(z)\tilde{\boldsymbol{A}}^{-1}(z)$ where $\tilde{\boldsymbol{A}}(z) = \boldsymbol{A}(z^{-1})\boldsymbol{z}^{n\boldsymbol{H}}$ and $\tilde{\boldsymbol{B}}(z) = \boldsymbol{B}(z^{-1})\boldsymbol{z}^{n\boldsymbol{H}}$. Matrix $\boldsymbol{z}^{n\boldsymbol{H}}$ is a diagonal matrix defined as $\boldsymbol{z}^{n\boldsymbol{H}} = diag\{z^{n\boldsymbol{H}_1}, z^{n\boldsymbol{H}_2}, \ldots, z^{n\boldsymbol{H}_m}\}$ where $n\boldsymbol{H}_j$ is the maximum degree of all transfer functions along the *j*th column of $\boldsymbol{H}(z^{-1})$. ■

From Definition 4.5 it is trivial to see that $\boldsymbol{B}(z^{-1})\boldsymbol{A}^{-1}(z^{-1}) = \tilde{\boldsymbol{B}}(z)\tilde{\boldsymbol{A}}(z)^{-1}$.

In the case of the closed-loop transfer matrices, where numerator and denominator matrices do not have the same degree, a coprime factorization of the individual $\boldsymbol{z}^{n}$ matrices is necessary. Thus for $\boldsymbol{G}_{uw}(z^{-1})$ we have that

$$\boldsymbol{G}_{uw}(z^{-1}) = \boldsymbol{A}(z^{-1})\boldsymbol{\Omega}(z^{-1}) = \tilde{\boldsymbol{A}}(z)\boldsymbol{z}^{-n\boldsymbol{H}}(\tilde{\boldsymbol{\Omega}}(z)\boldsymbol{z}^{-n\Omega})^{-1} = \tilde{\boldsymbol{A}}(z)\boldsymbol{z}^{-n\boldsymbol{H}}\boldsymbol{z}^{n\Omega}\tilde{\boldsymbol{\Omega}}^{-1}(z)$$

Multiply matrices $\boldsymbol{z}^{n\Omega}\boldsymbol{z}^{-n\boldsymbol{H}}$ to get $\boldsymbol{z}_{\min}^{n\Omega}\boldsymbol{z}_{\min}^{-n\boldsymbol{H}}$ where the common terms have been cancelled. Then

$$\boldsymbol{G}_{uw}(z^{-1}) = \boldsymbol{A}(z^{-1})\boldsymbol{\Omega}^{-1}(z^{-1}) = \tilde{\boldsymbol{A}}(z)\boldsymbol{z}_{\min}^{n\Omega}(\tilde{\boldsymbol{\Omega}}(z)\boldsymbol{z}_{\min}^{\boldsymbol{nH}})^{-1}$$

$$\boldsymbol{G}_{uw}(z^{-1}) = \boldsymbol{A}(z^{-1})\boldsymbol{z}^{\boldsymbol{nH}}\boldsymbol{z}_{\min}^{n\Omega}(\boldsymbol{\Omega}(z^{-1})\boldsymbol{z}^{n\Omega}\boldsymbol{z}_{\min}^{\boldsymbol{nH}})^{-1}$$

define $\widehat{\boldsymbol{A}}(z) = \boldsymbol{A}(z^{-1})\boldsymbol{z}^{\boldsymbol{nH}}\boldsymbol{z}_{\min}^{n\Omega}$ and $\widehat{\boldsymbol{\Omega}}(z) = \boldsymbol{\Omega}(z^{-1})\boldsymbol{z}^{n\Omega}\boldsymbol{z}_{\min}^{\boldsymbol{nH}}$

$$\widehat{\boldsymbol{G}}_{uw}(z) = \widehat{\boldsymbol{A}}(z)\widehat{\boldsymbol{\Omega}}(z)^{-1}$$

and similarly

$$\boldsymbol{G}_{yw}(z^{-1}) = \boldsymbol{B}(z^{-1})\boldsymbol{\Omega}^{-1}(z^{-1}) = \tilde{\boldsymbol{B}}(z)\boldsymbol{z}_{\min}^{n\Omega}(\tilde{\boldsymbol{\Omega}}(z)\boldsymbol{z}_{\min}^{\boldsymbol{nH}})^{-1}$$

$$\boldsymbol{G}_{yw}(z^{-1}) = \boldsymbol{B}(z^{-1})\boldsymbol{z}^{\boldsymbol{nH}}\boldsymbol{z}_{\min}^{n\Omega}(\boldsymbol{\Omega}(z^{-1})\boldsymbol{z}^{n\Omega}\boldsymbol{z}_{\min}^{\boldsymbol{nH}})^{-1}$$

define $\widehat{\boldsymbol{B}}(z) = \boldsymbol{B}(z^{-1})\boldsymbol{z}^{\boldsymbol{nH}}\boldsymbol{z}_{\min}^{n\Omega}$

$$\widehat{\boldsymbol{G}}_{yw}(z) = \widehat{\boldsymbol{B}}(z)\widehat{\boldsymbol{\Omega}}^{-1}(z)$$

and the determinants are given by:

$$\det(\widehat{\boldsymbol{\Omega}}(z)) = \det(\boldsymbol{\Omega}(z^{-1}))\det(\boldsymbol{z}^{n\Omega})\det(\boldsymbol{z}_{\min}^{\boldsymbol{nH}})$$

$$\det(\widehat{\boldsymbol{B}}(z)) = \det(\boldsymbol{B}(z^{-1}))\det(\boldsymbol{z}^{\boldsymbol{nH}})\det(\boldsymbol{z}_{\min}^{\Omega})$$

$$\det(\widehat{\boldsymbol{A}}(z)) = \det(\boldsymbol{A}(z^{-1}))\det(\boldsymbol{z}^{\boldsymbol{nH}})\det(\boldsymbol{z}_{\min}^{\Omega})$$

Note that for non-square systems the determinant of $\det(\widehat{\boldsymbol{\Omega}}(z))$ does not exist. For the analysis of stability and performance the above development shows that the backward shift-operator polynomial matrices contain all the relevant information.

DEFINITION 4.6 Let $\boldsymbol{N}(z^{-1})\boldsymbol{D}^{-1}(z^{-1})$ be a coprime MFD of $\boldsymbol{G}(z^{-1})$. The finite roots of $\det(\boldsymbol{D}^{-1}(z^{-1}))$ are the relevant poles of $\boldsymbol{G}(z^{-1})$. ■

Example 4.5 Consider the process of Example 4.4

$$\boldsymbol{H}(z^{-1}) = \begin{bmatrix} \dfrac{z^{-1}+z^{-2}1.5}{1-z^{-1}} & \dfrac{0.5z^{-1}}{1-z^{-1}} \\ \dfrac{0.04z^{-1}}{1-z^{-1}} & \dfrac{0.05z^{-1}}{1-z^{-1}} \end{bmatrix}$$

$$\boldsymbol{A}(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 \\ 0 & 1-z^{-1} \end{bmatrix} \qquad \boldsymbol{z}^{n\boldsymbol{H}} = \begin{bmatrix} z^2 & 0 \\ 0 & z \end{bmatrix}$$

$$\boldsymbol{B}(z^{-1}) = \begin{bmatrix} z^{-1}+z^{-2}1.5 & 0.5z^{-1} \\ 0.04z^{-1} & 0.05z^{-1} \end{bmatrix}$$

$$\boldsymbol{\Omega}(z^{-1}) = \begin{bmatrix} 1+2.5z^{-1} & 0 \\ -2z^{-1} & 1 \end{bmatrix} \qquad \boldsymbol{z}^{n\Omega} = \begin{bmatrix} z & 0 \\ 0 & 1 \end{bmatrix}$$

then

$$\tilde{\boldsymbol{A}}(z) = \boldsymbol{A}(z^{-1})\boldsymbol{z}^{n\boldsymbol{H}} = \begin{bmatrix} 1-z^{-1} & 0 \\ 0 & 1-z^{-1} \end{bmatrix}\begin{bmatrix} z^2 & 0 \\ 0 & z \end{bmatrix} = \begin{bmatrix} z(z-1) & 0 \\ 0 & (z-1) \end{bmatrix}$$

$$\tilde{\boldsymbol{B}}(z) = \boldsymbol{B}(z^{-1})\boldsymbol{z}^{n\boldsymbol{H}} = \begin{bmatrix} z^{-1}+z^{-2}1.5 & 0.5z^{-1} \\ 0.04z^{-1} & 0.05z^{-1} \end{bmatrix}\begin{bmatrix} z^2 & 0 \\ 0 & z \end{bmatrix} = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.05 \end{bmatrix}$$

$$\tilde{\boldsymbol{\Omega}}(z^{-1}) = \boldsymbol{\Omega}(z^{-1})\boldsymbol{z}^{n\Omega} = \begin{bmatrix} 1+2.5z^{-1} & 0 \\ -2z^{-1} & 1 \end{bmatrix}\begin{bmatrix} z & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} z+2.5 & 0 \\ -2 & 1 \end{bmatrix}$$

$$\boldsymbol{z}^{n\Omega}\boldsymbol{z}^{-n\boldsymbol{H}} = \begin{bmatrix} \frac{1}{z} & 0 \\ 0 & \frac{1}{z} \end{bmatrix}$$

$$z_{\min}^{n\Omega} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad z_{\min}^{n\boldsymbol{H}} = \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix}$$

then

$$\widehat{\boldsymbol{\Omega}}(z) = \tilde{\boldsymbol{\Omega}}(z)z_{\min}^{n\boldsymbol{H}} = \begin{bmatrix} z+2.5 & 0 \\ -2 & 1 \end{bmatrix}\begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} = \begin{bmatrix} z(z+2.5) & 0 \\ -2z & z \end{bmatrix}$$

$$\widehat{\boldsymbol{A}}(z) = \tilde{\boldsymbol{A}}(z)z_{\min}^{\Omega} = \begin{bmatrix} z(z-1) & 0 \\ 0 & (z-1) \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} z(z-1) & 0 \\ 0 & (z-1) \end{bmatrix}$$

$$\widehat{\boldsymbol{B}}(z) = \tilde{\boldsymbol{B}}(z)\boldsymbol{z}_{\min}^{\Omega} = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.05 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} z+1.5 & 0.5 \\ 0.04z & 0.05 \end{bmatrix}$$

and

$$\det(\mathbf{\Omega}(z^{-1}))\det(\boldsymbol{z}^{n\Omega})\det(\boldsymbol{z}_{\min}^{n\boldsymbol{H}}) = (1+2.5z^{-1})(z)(z^2) = (z+2.5)z^2$$

$$\det(\boldsymbol{B}(z^{-1}))\det(\boldsymbol{z}^{n\boldsymbol{H}})\det(\boldsymbol{z}_{\min}^{\Omega}) = (0.03z^{-2}+0.075z^{-3})(z^3)(1) = 0.03z+0.075$$

$$\det(\boldsymbol{A}(z^{-1}))\det(\boldsymbol{z}^{n\boldsymbol{H}})\det(\boldsymbol{z}_{\min}^{\Omega}) = (1-z^{-1})^2(z^3)(1) = (z-1)^2 z$$

if we use the backward shift-operator to obtain the determinants

$$\det(\mathbf{\Omega}(z^{-1})) = 1+2.5z^{-1} = \frac{z+2.5}{z}$$

$$\det(\boldsymbol{B}(z^{-1})) = 0.03z^{-2}+0.075z^{-3} = \frac{0.03z+0.075}{z^3}$$

$$\det(\boldsymbol{A}(z^{-1})) = (1-z^{-1})^2 = \frac{(z-1)^2}{z^2} \qquad \blacksquare$$

The particular format of the closed-loop transfer matrices and the possibility of determination of the poles of the system lead to the following theorems:

THEOREM 4.4 Let $\boldsymbol{H}$ be a proper transfer function. The closed-loop system will be stable if the roots of $\det(\bar{\mathbf{\Omega}}_{uw})$ and $\det(\bar{\mathbf{\Omega}}_{yw})$ are within the unit circle.

Proof From Theorem 4.3 the poles of transfer matrices $\boldsymbol{G}_{uw}$ and $\boldsymbol{G}_{yw}$ are given by the roots of $\det(\bar{\mathbf{\Omega}}_{uw})$ and $\det(\bar{\mathbf{\Omega}}_{yw})$. Thus if these roots are within the unit circle then the system is stable. *Q.E.D.*

However if $\boldsymbol{A}$ and $\boldsymbol{B}$ are coprime then these matrices cannot cancel the same unstable poles of $\boldsymbol{G}_{uw}$ and $\boldsymbol{G}_{yw}$.

THEOREM 4.5 Let $\boldsymbol{H}$ be a proper transfer function. Let $\boldsymbol{H} = \boldsymbol{B}\boldsymbol{A}^{-1}$ be a coprime matrix fraction description of $\boldsymbol{H}$. The closed-loop system will be stable if and only if the roots of $\det(\mathbf{\Omega})$ are within the unit circle.

Proof The closed-loop transfer matrices $\boldsymbol{G}_{uw}$ and $\boldsymbol{G}_{yw}$ have, before any cancellation of common terms, the same denominator matrix $\boldsymbol{\Omega}$. Therefore, if the roots of the det($\boldsymbol{\Omega}$) are within the unit circle, then $\bar{\boldsymbol{\Omega}}_{uw}$ and $\bar{\boldsymbol{\Omega}}_{yw}$ have their roots within the unit circle and according to Theorem 4.3 the system is stable. Conversely if any one of the roots of the det($\boldsymbol{\Omega}$) are outside the unit circle they cannot all be cancelled by $\boldsymbol{A}$ and by $\boldsymbol{B}$ once these are coprime matrices and either $\bar{\boldsymbol{\Omega}}_{uw}$ or $\bar{\boldsymbol{\Omega}}_{yw}$, or both, will have roots outside the unit circle and according to Theorem 4.4 the system is unstable *Q.E.D.*

Although there is no direct method for obtaining a coprime matrix fraction representation of the process transfer matrix it is, in general, a simple procedure if the transfer matrix of the plant is minimal. Stability can be determined by evaluating the position of the roots of det($\boldsymbol{\Omega}$) which can be done with Jury's criteria.

Theorem 4.5 is very important for the implementation of a design procedure for multivariable predictive control once it allows a fast initial test of stability of the system.

4.5 Performance Analysis

Crisalle *et al.* (1989) have proven that for the SISO case if *Nu*=*Ny* and λ=0 then the predictive controller generates a control that results in the inversion of the plant and the output matches the reference. A similar result is obtained for the MIMO case as can be seen next.

THEOREM 4.6 Consider a system with equal number of inputs and outputs, i.e. $m = p$. Let $\boldsymbol{\Lambda} = 0$ and $Nu_j = Ny_j$ for all *j*, then the resulting closed-loop transfer matrix is the identity, $\boldsymbol{G}_{yr} \triangleq \boldsymbol{H}(\boldsymbol{R}+\boldsymbol{SH})^{-1}\boldsymbol{T} \equiv \boldsymbol{I}$.

Proof Recall equation (3.10), $\Delta\boldsymbol{u} = (\boldsymbol{G}^T\boldsymbol{\Phi}\boldsymbol{G} + \boldsymbol{\Lambda})^{-1}\boldsymbol{G}^T\boldsymbol{\Phi}(\boldsymbol{r} - \boldsymbol{y}^0)$, in the derivation of the control law. The condition that $\boldsymbol{\Lambda}$=0 and Nu_j=Ny_j for all j implies that $(\boldsymbol{G}^T\boldsymbol{\Phi}\boldsymbol{G} + \boldsymbol{\Lambda})^{-1}\boldsymbol{G}^T\boldsymbol{\Phi} = \boldsymbol{G}^{-1}$. Matrix $\boldsymbol{G}$ is a block lower triangular matrix and its inverse has a similar structure. Therefore, recalling that $\boldsymbol{K}$ is the matrix of rows corresponding to the first control move of $(\boldsymbol{G}^T\boldsymbol{\Phi}\boldsymbol{G} + \boldsymbol{\Lambda})^{-1}\boldsymbol{G}^T\boldsymbol{\Phi}$ (which is $\boldsymbol{G}^{-1}$ in this case), its elements are vectors with a nonzero first element followed by zeros, in this case. This particular format of $\boldsymbol{K}$ has a similar effect of restricting the prediction horizons to $Ny_j = 1$ in the formulation of the control problem, once the elements of $\boldsymbol{K}$ (defined as $\boldsymbol{k}_{\ell j}^T = [\,k_1 \quad k_2 \quad \cdots \quad k_{Ny_j}\,]_{\ell j}$) are $\boldsymbol{k}_{\ell j}^T = [\,k_1 \quad 0 \quad \cdots \quad 0\,]_{\ell j}$. Thus $\boldsymbol{KP} \equiv \boldsymbol{K}_1\boldsymbol{P}_1$ and $\boldsymbol{KF} \equiv \boldsymbol{K}_1\boldsymbol{F}_1$ (see Appendix B for details of both proofs). From the definitions of $\boldsymbol{R}$ and $\boldsymbol{S}$ then

$$\boldsymbol{R} = \Delta(\boldsymbol{I} + z^{-1}\boldsymbol{K}_1\boldsymbol{P}_1)$$

$$\boldsymbol{S} = \boldsymbol{K}_1\boldsymbol{F}_1$$

Therefore $\boldsymbol{\Omega}$ becomes

$$\boldsymbol{\Omega} = \Delta(\boldsymbol{I} + z^{-1}\boldsymbol{K}_1\boldsymbol{P}_1)\boldsymbol{A}_R + z^{-1}\boldsymbol{K}_1\boldsymbol{F}_1\boldsymbol{B}_R \tag{4.10}$$

From *Matrix Diophantine* equation, for $i = 1$ and bearing in mind that $\boldsymbol{E}_1 = \boldsymbol{I}$, then $\boldsymbol{F}_1 = z(\boldsymbol{I} - \Delta\boldsymbol{A}_L)$ and from equation (3.16) we get $\boldsymbol{P}_1 = z(\boldsymbol{B}_L - \boldsymbol{G}_1)$. Substituting in the expression for $\boldsymbol{\Omega}$ then

$$\boldsymbol{\Omega} = \Delta\left[\boldsymbol{I} + z^{-1}\boldsymbol{K}_1 z(\boldsymbol{B}_L - \boldsymbol{G}_1)\right]\boldsymbol{A}_R + z^{-1}\boldsymbol{K}_1 z(\boldsymbol{I} - \Delta\boldsymbol{A}_L)\boldsymbol{B}_R \tag{4.11}$$

but $\boldsymbol{K}_1 \equiv \boldsymbol{G}_1^{-1}$ and

$$\boldsymbol{\Omega} = \Delta\boldsymbol{A}_R + \Delta\boldsymbol{G}_1^{-1}\boldsymbol{B}_L\boldsymbol{A}_R - \Delta\boldsymbol{A}_R + \boldsymbol{G}_1^{-1}\boldsymbol{B}_R - \boldsymbol{G}_1^{-1}\Delta\boldsymbol{A}_L\boldsymbol{B}_R\,.$$

The coprime factorizations of $\boldsymbol{H}$ imply that $\boldsymbol{B}_L \boldsymbol{A}_R = \boldsymbol{A}_L \boldsymbol{B}_R$. Cancellation of common terms leads to

$$\boldsymbol{\Omega} = \boldsymbol{G}_1^{-1} \boldsymbol{B}_R \tag{4.12}$$

This way

$$\boldsymbol{G}_{yr} = \boldsymbol{B}_R (\boldsymbol{G}_1^{-1} \boldsymbol{B}_R)^{-1} \boldsymbol{G}_1 = \boldsymbol{I} \qquad \textit{Q.E.D.}$$

Example 4.6 Consider the process of Example 4.4.

$$\boldsymbol{H} = \begin{bmatrix} \dfrac{z+1.5}{z(z-1)} & \dfrac{0.5}{(z-1)} \\ \dfrac{0.04}{(z-1)} & \dfrac{0.05}{(z-1)} \end{bmatrix}$$

A multivariable predictive controller was designed for this process and the following parameters were used: $Ny = 1,\ Nu = 1,\ \phi_1 = 1,\ \ \phi_2 = 1,\ \lambda_1 = 0,\ \lambda_2 = 0$. Matrices $\boldsymbol{R}$, $\boldsymbol{S}$ and $\boldsymbol{T}$ are:

$$\boldsymbol{R} = \begin{bmatrix} \dfrac{z^2+1.5z-2.5}{z^2} & 0 \\ \dfrac{-2(z-1)}{z^2} & \dfrac{z-1}{z} \end{bmatrix} \quad \boldsymbol{S} = \begin{bmatrix} \dfrac{5(2z-1)}{3z} & -\dfrac{50(2z-1)}{3z} \\ -\dfrac{4(2z-1)}{3z} & \dfrac{100(2z-1)}{3z} \end{bmatrix} \quad \boldsymbol{T} = \begin{bmatrix} \dfrac{5}{3}z & -\dfrac{50}{3}z \\ -\dfrac{4}{3}z & \dfrac{100}{3}z \end{bmatrix}$$

and

$$(\boldsymbol{RA} + \boldsymbol{SB}) = \begin{bmatrix} z(z+2.5) & 0 \\ -2z & z \end{bmatrix}$$

$$\boldsymbol{G}_{yw} = \boldsymbol{B}(\boldsymbol{RA} + \boldsymbol{SB})^{-1} = \begin{bmatrix} \dfrac{1}{z} & \dfrac{0.5}{z} \\ \dfrac{0.04}{z} & \dfrac{0.05}{z} \end{bmatrix}$$

from where we see that

$$\boldsymbol{G}_{yr} = \boldsymbol{B}(\boldsymbol{RA} + \boldsymbol{SB})^{-1} \boldsymbol{T} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad \blacksquare$$

It is of interest to establish conditions under which the multivariable predictive controller produces offset-free responses to step changes in the set point of any given output as was proven by Crisalle *et al*. (1989) for SISO systems.

THEOREM 4.7 Consider a plant $\boldsymbol{H}$ with equal number of inputs and outputs where (i) $\boldsymbol{H}(1)$ is nonsingular, (ii) the weights ϕ_j in the objective function equation are > 0 for all *j* and (iii) the resulting closed-loop is stable. Then it achieves zero offset response

$$\lim_{t\to\infty}\left[\boldsymbol{r}(t)-\boldsymbol{y}(t)\right]=0$$

to step changes in the reference $\boldsymbol{r}(t)$.

Proof In the z domain $\boldsymbol{r}(z)-\boldsymbol{y}(z)=\left[\boldsymbol{I}-\boldsymbol{G}_{yr}(z)\right]\boldsymbol{r}(z)$. From the Final Value Theorem

$$\lim_{t\to\infty}[\boldsymbol{r}(t)-\boldsymbol{y}(t)]=\lim_{z\to 1}\Delta\left[\boldsymbol{I}-\boldsymbol{G}_{yr}(z)\right]\frac{\boldsymbol{\alpha}}{\Delta} \tag{4.13}$$

where $\frac{\boldsymbol{\alpha}}{\Delta}$ is a vector of step inputs of amplitude $\boldsymbol{\alpha}$. The transfer matrix $\boldsymbol{G}_{yr}(z)$ is

$$\boldsymbol{G}_{yr}=\boldsymbol{H}(\boldsymbol{R}+\boldsymbol{S}\boldsymbol{H})^{-1}\boldsymbol{T}$$

From the definitions of $\boldsymbol{R}$ and $\boldsymbol{S}$ (3.19) to (3.21) $\boldsymbol{G}_{yr}$ becomes

$$\boldsymbol{G}_{yr}=\boldsymbol{H}(\Delta(\boldsymbol{I}+z^{-1}\boldsymbol{K}_i\boldsymbol{P}_i)+\boldsymbol{K}_i\boldsymbol{F}_i\boldsymbol{H})^{-1}\boldsymbol{T}_i$$

From *Matrix Diophantine* equation $\boldsymbol{F}_i=z^i(\boldsymbol{I}-\boldsymbol{E}_i\Delta\boldsymbol{A})$ and

$$\boldsymbol{G}_{yr}=\boldsymbol{H}\left[\Delta(\boldsymbol{I}+z^{-1}\boldsymbol{K}_i\boldsymbol{P}_i)+\boldsymbol{K}_i z^i(\boldsymbol{I}-\Delta\boldsymbol{A})\boldsymbol{H}\right]^{-1}\boldsymbol{T}_i$$

Recall that $\boldsymbol{K}_i z^i=\boldsymbol{T}_i$. At $z=1$ the delta operator becomes $\Delta=0$ and therefore

$$\boldsymbol{G}_{yr}(1)=\boldsymbol{H}(1)(\boldsymbol{T}(1)\boldsymbol{H}(1))^{-1}\boldsymbol{T}(1)=\boldsymbol{I}$$

and substituting in (4.13)

$$\lim_{t\to\infty}\left[r(t)-y(t)\right]=0 \qquad \textit{Q.E.D.}$$

Condition (i) requires the plant to have a nonsingular gain, i.e., there exits the inverse matrix $\boldsymbol{H}^{-1}(1)$. This condition is a necessary requirement for offset-free behavior, otherwise, there is no solution $\bar{\boldsymbol{u}}$ to the steady-state equation $\bar{\boldsymbol{r}} = \boldsymbol{H}(1)\bar{\boldsymbol{u}}$ where $\bar{\boldsymbol{r}}$ is the final value of the step-function in the set point and $\bar{\boldsymbol{u}}$ is the final value of the input required to deliver an output that exactly matches the final value of the reference. Systems with singular gains are functionally uncontrollable (Skogestad, 1996) and hence perfect tracking for any arbitrary step change is impossible to attain. Finally, condition (iii) is necessary because $\phi_j = 0$ for some value of *j*, then deviations from the set point for output y_j are not penalized in the objective function equation, and consequently, the controller cannot ensure attainment of perfect tracking.

4.6 Implementation

The mathematical development of this chapter allowed the generation of a Matlab code for the analysis of stability of unconstrained multivariable predictive control.

Based on Theorem 4.5 a design procedure was created where the critical step is the generation of a right coprime description of a process model, $\boldsymbol{H}$, for which no direct method is known. Maple was used to generate the greatest common divisors for systems where reduction of MFDs to lowest terms was necessary. In order to avoid the round-off errors of floating point operations a rational representation of the coefficients of the polynomials was done allowing the division algorithm for polynomials to give exact results. This way the elementary row and column operations, required in the process of obtaining the greatest common divisors, are executed correctly. The drawback of this approach resides in the fact that as the convolutions of the polynomials are executed the rational number representations of the coefficients tend to have increasing numerators and denominators. This imposes a limitation on the size of MFDs that can be reduced to lowest terms.

For square systems the ‘numerator’ matrix possesses (in general) a determinant and a simple (initial) test of coprimeness is possible once coprime matrices will not have common roots in their determinant. The presence of common roots, however, does not imply reducibility of the MFD once poles and zeros can be located in different positions in the transfer matrix.

Thus given matrices $\boldsymbol{R}$ and $\boldsymbol{S}$ (refer to Chapter 3) and a right coprime MFD of the process matrix ($\boldsymbol{H} = \boldsymbol{B}\boldsymbol{A}^{-1}$) then $\boldsymbol{\Omega}$ is generated, Jury’s criteria is applied to the determinant of $\boldsymbol{\Omega}$ to check for stability. In the presence of unstable roots then these parameters are discarded for another set until only stable roots are present.

Performance is verified by simulating the system in Simulink. The block diagram structure of Simulink allows the simulation of the closed-loop system without the need to invert $\boldsymbol{R} + \boldsymbol{S}\boldsymbol{H}$ as would be the case if a numerical integrator were to be used. Although the inversion of $\boldsymbol{R}$ is still necessary the following algebraic manipulation manages to use the block diagram properties of Simulink to execute the inversion:

$$\boldsymbol{R} = (\boldsymbol{I} + z^{-1}\boldsymbol{K}\boldsymbol{P})\Delta = \boldsymbol{I} - z^{-1}(\boldsymbol{I} - \Delta\boldsymbol{K}\boldsymbol{P})$$

Let $\boldsymbol{R}' = z^{-1}(\boldsymbol{I} - \Delta\boldsymbol{K}\boldsymbol{P})$ then

$$\boldsymbol{R} = \boldsymbol{I} - \boldsymbol{R}'$$

and $\boldsymbol{R}^{-1} = (\boldsymbol{I} - \boldsymbol{R}')^{-1}$. This way the inversion of matrix $\boldsymbol{R}$ can be represented in block diagram as seen in Figure 4.2.

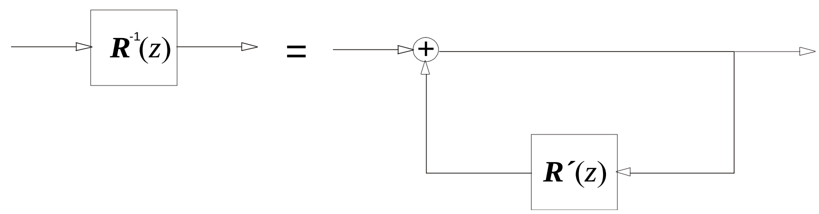

Figure 4.2 Inversion of ***R*** in Simulink

The Matlab code was modified to generate matrix $\boldsymbol{R}'$. Simulations done showed very good results.

4.7 Conclusion

In this chapter the closed-loop behavior of a plant with a multivariable generalized predictive controller is analyzed and a method, based on matrix fraction description (MFD), is developed to check for stability and determine the closed-loop poles for any choice of tuning parameters. Performance is analyzed by simulating the closed-loop system in Simulink.

Properties such as “zero offset response to step changes in the reference” and “inversion of the plant” are proven for plants with equal number of inputs and outputs.

CHAPTER 5
THREE-PHASE SEPARATOR

Offshore production systems are responsible for the treatment of the petroleum produced in the sea fields. These systems are subject to intense fluctuations in its feed due to the multiphase nature of the fluids coming from the reservoir. The feed consists of a mixture of oil, water and gas with different degrees of dispersion between phases. Separators are responsible for absorbing fluctuations of the feed as well as for promoting the separation. Its performance is of vital importance for the quality of both, the oil exported to refineries and the water discharged to the sea. Economic and environmental restrictions have led the industry to research means of improving the efficiency of the equipment. Little has been done, though, in the use of advanced control techniques.

Separators possess three independent SISO PI (proportional and integral) control loops, one for each phase. Frequently, bad tuning of the controllers is responsible for a low yield in the separation process. The apparent periodicity of the multiphase flow of the feed raises hope that it can be predicted at some degree. Although this phenomenon will not be studied here, future values of set points (reference trajectory) are assumed to be known.

Therefore this dissertation will, hopefully, show the benefits and lay the formal grounds for the future development of predictive controllers for three phase separators.

Most separators are horizontal as will be considered in this dissertation.

5.1 Description of the Separator

Separators use gravity as the driving force for separation. For that reason small droplets tend not to separate and a low efficiency is normally observed.

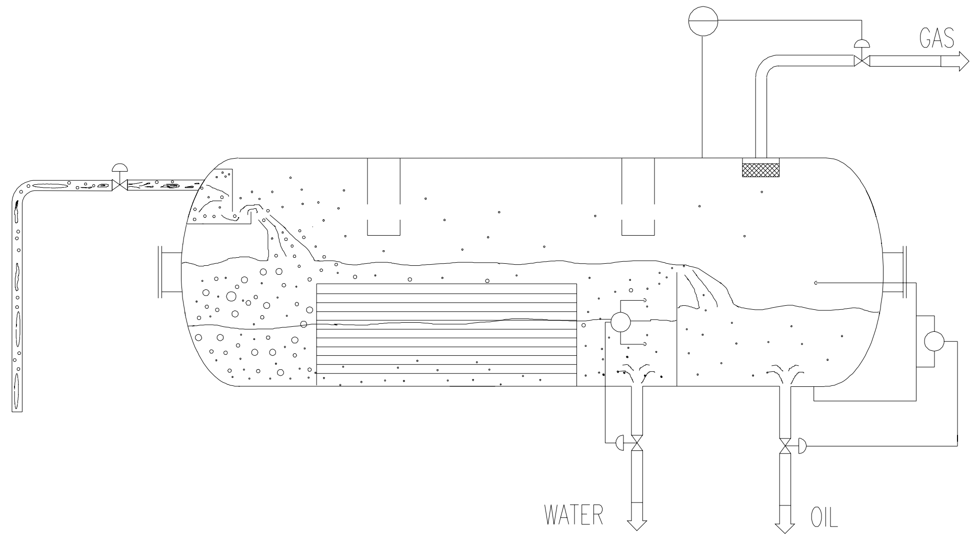

Figure 5.1 Oil-Water-Gas Separator

The vessel is divided in two sections separated by a weir; a separation chamber and an oil chamber. Gravitational force promotes the segregation of phases and a water rich phase with dispersed oil settles at the bottom of the separation chamber while an oil rich phase with dispersed water lays atop. An interface between the two phases is formed. The liquids flow in the direction of the weir and along this path a series of parallel plates help the liquid-liquid separation. The oil rich phase flows over the weir into the oil chamber. The water rich phase is discharged just before the weir to the water treatment unit.

An interface level controller manipulates the opening of the water (outlet) valve. A level controller controls the level of oil in the oil chamber. Pressure inside the vessel is regulated by a controller acting on the gas valve.

The outputs (measurable states) are pressure, height of water (height of interface) and height of oil in the oil chamber. The inputs are the opening of the valves.

Parallel plates are used to reduce the length of space required for the separation of dispersed droplets. These adhere to the plates and coalesce. The plates are inclined allowing for the fall (in the case of water droplets) or rise (in the case of oil) droplets.

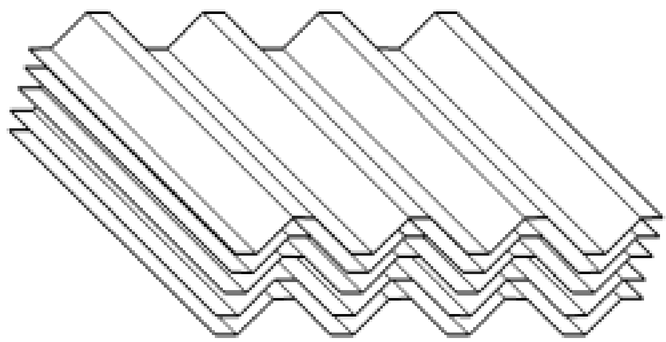

Figure 5.2 Parallel Plates

5.2 Rigorous Model

The model here developed is a simplification of a rigorous model (Nunes, 1994) which has the following aspects:

- The feed is composed of 15 chemical components and one pseudo-component.
- The model possesses a total of 81 variables.
- Thermodynamic equilibrium of phases. This is assumed because liquid phase residence time is approximately 5 minutes while thermodynamic equilibrium takes approximately 30 seconds to occur. Soave-Redlich-Kwong (SRK) equation of state is used.
- Droplet distribution of dispersions at the entrance of the separator is known.
- Stokes regime is considered for the calculation of terminal velocity of droplets of water and oil. This is a reasonable assumption since the liquid phases have low velocities and laminar flow is observed.
- Droplets are perfect spheres.

- Effects of wall and concentration are negligible in the calculation of terminal velocities

- No coalescence of droplets occurs.

- No emulsifying agents present. This implies that the system is a dispersion.

- No dispersion of liquid particles in the gas phase.

- A parabolic profile for the velocity of water is considered in the formulation.

- Droplets are uniformly distributed in space at the entrance of parallel plates.

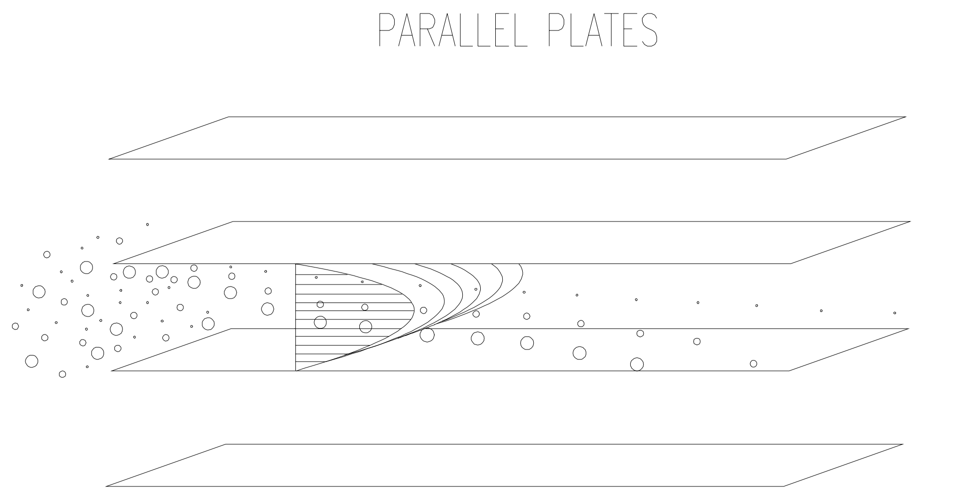

Figure 5.3 Velocity Profile

5.3 Simplified Model

Simulations done with the rigorous model show that thermodynamic representation of phases is important in studying the behavior of the gas but has little effect on the dynamics of liquid-liquid separation. Because the main interest here is to control and improve the oil-water separation, the following assumption is made:

- No mass transfer between thermodynamic phases occurs.

The major impact of this assumption is that variations in pressure, temperature or composition do not cause a mass change of the individual (thermodynamic) phases.

Because fluctuations observed (in practice) are negligible, the assumption is justified and allows for a simpler model with 7 state variables.

Thermodynamic calculations determine the total amount of each phase present in the vessel. The degree of dispersion between phases, however, is calculated using dynamic population balances. To avoid confusion between thermodynamic phases and dispersions, notation is carefully defined.

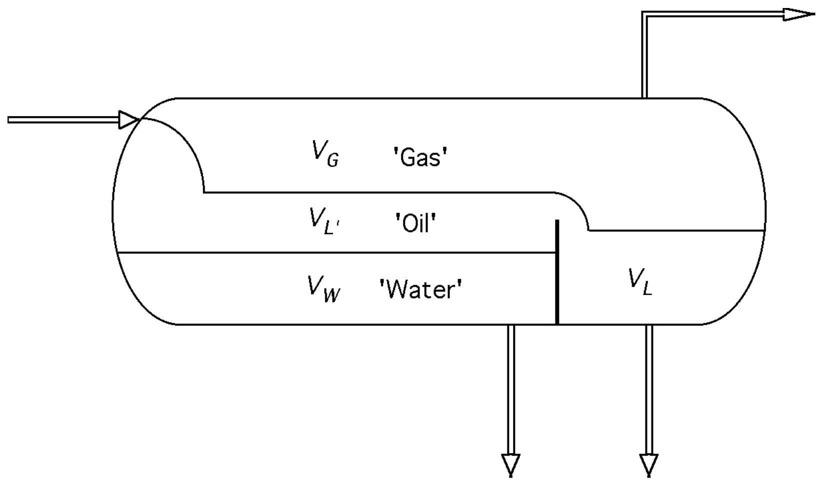

Figure 5.4 Volumes

Following common practice, 'water' will, hereon, refer to the water rich phase lying at the bottom of the separation chamber as seen in Figure 5.4. Note that it is a mixture of two thermodynamic phases, namely, the oil particles (dispersed phase) and the water (continuous phase). Properties associated to the 'water' will have a subscript W while lower case italic superscripts will refer to the thermodynamic phases; *w* standing for the water and ℓ for the oil. Thus V_{W}^{ℓ} is the volume of oil in the 'water' and V_{W}^{w} is the volume of water in the 'water'. The total volume of the 'water' is:

$$V_{\mathrm{W}} = V_{\mathrm{W}}^{\ell} + V_{\mathrm{W}}^{w}$$

Following the same rationale 'oil' will be used to name the mixture of oil (continuous phase) and water (dispersed phase) present in the separation chamber and the oil chamber. Distinction, however, must be made between the two since they have different properties such as volume, concentration of water, etc. Properties associated to the 'oil' in the oil chamber will have a subscript L while to represent the 'oil' in the separation chamber a subscript Lsc will be used. Thus V_{L}^{w} is the volume of water in the 'oil' in the oil chamber.

Subscript G will be used for the gas phase. Note that the assumption of no dispersion of droplets of oil or water in the gas phase makes it unnecessary to distinguish continuous and dispersed phases.

The above definitions allow the following algebraic relations to be established for volumes and concentrations:

Total volume restriction $$V = V_{\mathrm{G}} + V_{\mathrm{W}} + V_{\mathrm{Lsc}} + V_{\mathrm{L}}$$

Volume of 'water' in separation chamber: $$V_{\mathrm{W}} = V_{\mathrm{W}}^{\ell} + V_{\mathrm{W}}^{w}$$

Volume of 'oil' in separation chamber $$V_{\mathrm{Lsc}} = V_{\mathrm{Lsc}}^{\ell} + V_{\mathrm{Lsc}}^{w}$$

Volume of 'oil' in oil chamber $$V_{\mathrm{L}} = V_{\mathrm{L}}^{\ell} + V_{\mathrm{L}}^{w}$$

Concentration of dispersed water in 'oil' at separation chamber $$C_{\mathrm{Lsc}}^{w} = \frac{V_{\mathrm{Lsc}}^{w}}{V_{\mathrm{Lsc}}}$$

Concentration of dispersed oil in 'water' at separation chamber $$C_{\mathrm{W}}^{\ell} = \frac{V_{\mathrm{W}}^{\ell}}{V_{\mathrm{W}}}$$

Concentration of dispersed water in 'oil' at oil chamber $$C_{\mathrm{L}}^{w} = \frac{V_{\mathrm{L}}^{w}}{V_{\mathrm{L}}}$$

5.4 Flow Equations

Figure 5.5 shows the heights and flow rates of each phase. The equations are seen next:

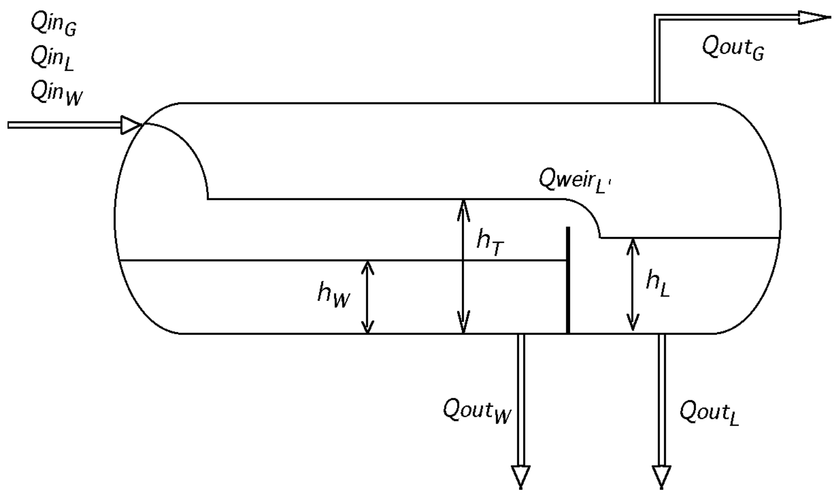

Figure 5.5 Heights and Flow rates

$$Qout_{\mathrm{W}} = \frac{Cv_{\mathrm{W}} s_{\mathrm{W}} \sqrt{d_{\mathrm{W}}(p - p_{\mathrm{W}}) + \gamma_{\mathrm{W}} h_{\mathrm{W}} + \gamma_{\mathrm{Lsc}}(h_{\mathrm{T}} - h_{\mathrm{W}})}}{\rho_{\mathrm{W}} 0.0693} \tag{5.1}$$

$$Qweir = 24.88\sqrt{2g}\,(length_{weir} - 0.2(h_T - h_{weir}))(h_T - h_{weir})^{1.5} \tag{5.2}$$

$$Qout_{\mathrm{L}} = \frac{Cv_{\mathrm{L}} s_{\mathrm{L}} \sqrt{d_{\mathrm{L}}(p - p_{\mathrm{L}}) + \gamma_{\mathrm{L}} h_{\mathrm{L}}}}{\rho_{\mathrm{L}} 0.0693} \tag{5.3}$$

$$Qout_{\mathrm{G}} = \frac{Cv_{\mathrm{G}} s_{\mathrm{G}} \sqrt{d_{\mathrm{G}}(p - p_{\mathrm{G}})(p + p_{\mathrm{G}})}}{\rho_{\mathrm{G}} 2.832} \tag{5.4}$$

5.5 Differential Equations

Total height at separation chamber (derived from the total material balance):

$$\frac{dh_{\mathrm{T}}}{dt} = \frac{Qin_{\mathrm{L}} + Qin_{\mathrm{W}} - Qweir - Qout_{\mathrm{W}}}{2C_{sc}\sqrt{(D - h_{\mathrm{T}})h_{\mathrm{T}}}} \tag{5.5}$$

Height of water (derived from the mass balance of the water phase):

$$\frac{dh_{\mathrm{W}}}{dt}=\frac{Qin_{\mathrm{W}}(1-TOG\cdot\varepsilon_{\mathrm{W}}^{\ell})-Qin_{\mathrm{L}}BSW\cdot\varepsilon_{\mathrm{L}}^{w}}{2C_{sc}\sqrt{(D-h_{\mathrm{W}})h_{\mathrm{W}}}} \tag{5.6}$$

Height of oil at the oil chamber (derived from the mass balance of the oil phase):

$$\frac{dh_{\mathrm{L}}}{dt}=\frac{Qweir-Qout_{\mathrm{L}}}{2C_{lc}\sqrt{(D-h_{\mathrm{L}})h_{\mathrm{L}}}} \tag{5.7}$$

Balance of volume of water dispersed in the ‘oil’ in the separation chamber:

$$\frac{dV_{\mathrm{Lsc}}^{w}}{dt}=Qin_{\mathrm{L}}BSW(1-\varepsilon_{\mathrm{Lsc}}^{w})-Qweir\frac{V_{\mathrm{Lsc}}^{w}}{V_{\mathrm{Lsc}}} \tag{5.8}$$

Balance of volume of oil dispersed in the ‘water’ in the separation chamber

$$\frac{dV_{\mathrm{W}}^{\ell}}{dt}=Qin_{\mathrm{W}}TOG(1-\varepsilon_{\mathrm{W}}^{\ell})\text{-}Qout_{\mathrm{W}}\frac{V_{\mathrm{W}}^{\ell}}{V_{\mathrm{W}}} \tag{5.9}$$

Balance of water dispersed in the ‘oil’ in the oil chamber

$$\frac{dV_{\mathrm{L}}^{w}}{dt}=Qweir_{\mathrm{L}}\frac{V_{\mathrm{Lsc}}^{w}}{V_{\mathrm{Lsc}}}-Qout_{\mathrm{L}}\frac{V_{\mathrm{L}}^{w}}{V_{\mathrm{L}}} \tag{5.10}$$

Pressure equation (derived from the material balance of gas)

$$\frac{dp}{dt}=\frac{1}{V-V_{\mathrm{W}}-V_{\mathrm{Lsc}}-V_{\mathrm{L}}}\left\{\frac{(RT)^{2}(Qin_{\mathrm{G}}-Qout_{\mathrm{G}})}{p\cdot MW_{\mathrm{G}}}-p\cdot(Qin_{\mathrm{L}}+Qin_{\mathrm{W}}-Qout_{\mathrm{L}}-Qout_{\mathrm{W}})\right\} \tag{5.11}$$

5.6 Conclusion

In this chapter a simplified dynamic model of the separator was created based on a more rigorous model previously developed. This simplification greatly reduced the number of states without loss of accuracy for the variables of interest in the development of a controller for the equipment. This model was programmed in Matlab and Simulations done in Simulink were used to check it against the rigorous model giving good results.

5.7 Nomenclature

h_{T}	total height ('water' and 'oil') at separation chamber	[m]
h_{Lsc}	height of 'oil' in separation chamber	[m]
h_{L}	height of 'oil' in the oil chamber	[m]
h_{W}	height of 'water' in separation chamber	[m]
D	diameter of separator	[m]
C_{sc}	length of separation chamber	[m]
C_{lc}	length of oil chamber	[m]
C_{Lsc}^{w}	concentration of water in the 'oil' in the sep. chamber	[m^3/m^3]
C_{W}^{ℓ}	concentration of oil in the 'water' in the sep. chamber	[m^3/m^3]
C_{L}^{w}	concentration of water in the 'oil' in the sep. chamber	[m^3/m^3]
d_{L}	density of 'oil' in oil chamber	[kgf/cm^3]
d_{W}	density of 'water' in oil chamber	[kgf/cm^3]
d_{G}	density of 'gas' in oil chamber	[kgf/cm^3]
$\varepsilon_{\mathrm{L}}^{w}$	efficiency of removal of water from 'oil' in oil chamber	[dimensionless]
$\varepsilon_{\mathrm{W}}^{\ell}$	efficiency of removal of oil from 'water' in oil chamber	[dimensionless]
$\varepsilon_{\mathrm{Lsc}}^{w}$	efficiency of removal of water from 'oil' in sep. chamber	[dimensionless]
Qin_{L}	flow rate of 'oil' into separator	[m^3/minutes]
Qin_{W}	flow rate of 'water' into separator	[m^3/minutes]
Qin_{G}	flow rate of 'gas' into separator	[m^3/minutes]
$Qweir$	flow rate of 'oil' over weir	[m^3/minutes]

$Qout_L$	flow rate of 'oil' out of separator	[m^3/minutes]
$Qout_W$	flow rate of 'water' out of separator	[m^3/minutes]
$Qout_G$	flow rate of gas out of separator	[m^3/minutes]
V	Total volume of separator	[m]
V_G	Volume of the gas	[m^3]
V_W	Volume of the 'water'	[m^3]
V_{Lsc}	Volume of the 'oil' in separation chamber	[m^3]
V_L	Volume of the 'oil' in oil chamber	[m^3]
V_{Lsc}^{ℓ}	Volume of oil in the 'oil' in separation chamber	[m^3]
V_{Lsc}^{w}	Volume of dispersed water in 'oil' in separation chamber	[m^3]
V_L^{ℓ}	Volume of oil in 'oil' in oil chamber	[m^3]
V_L^{w}	Volume of dispersed water in 'oil' in oil chamber	[m^3]
V_W^{ℓ}	Volume of oil in the 'water' in oil chamber	[m^3]
V_W^{w}	Volume of water in the 'water' in oil chamber	[m^3]
p	Pressure	[kgf/cm^3]
R	Constant of gases	[atm.liter/mol.K]
s_L	opening of the 'oil' valve	[dimensionless]
s_W	opening of the 'water' valve	[dimensionless]
s_G	opening of the 'gas' valve	[dimensionless]
T	Temperature	[K]
BSW	concentration of water in the 'oil' in the feed	[m^3/m^3]
TOG	concentration of oil in the 'water' in the feed	[m^3/m^3]

CHAPTER 6
DESIGN OF A PREDICTIVE CONTROLLER FOR THE SEPARATOR

In this chapter a multivariable predictive controller is designed for the separator with the help of the Matlab code previously developed. A linearized model of the separator is used and the closed-loop system is simulated for the analysis of performance.

Many different combination of tuning parameters are tested. For the set of parameters where stability is verified, performance is analyzed by simulating the closed-loop system. Initially two sets of parameters are chosen, a first one, which results in very intense oscillations in the height of water and a second set where the parameters are successfully changed with the purpose of reducing these oscillations. Further on another set up where the pressure is controlled separately and the two remaining controlled variables (heights of water and oil) are controlled by a multivariable predictive controller is proposed and studied.

6.1 Discrete Linear Model of the Separator

To design a predictive controller for the separator a discrete linear model of the equipment is necessary. The nonlinear model of the separator, developed in Chapter 5, was identified by evaluating the open loop response of the system to step perturbations in inputs (opening of valves). The following model was generated where the variables are now deviation variables.

$$\begin{bmatrix} h_{\mathrm{W}} \\ h_{\mathrm{L}} \\ p \end{bmatrix} = \begin{bmatrix} -\dfrac{17}{206s+1} & 0 & \dfrac{2.4}{367s+1} \\ -\dfrac{126}{322s+1} & -\dfrac{169}{330s+1} & \dfrac{43}{508s+1} \\ 0 & 0 & -\dfrac{2.4}{2s+1} \end{bmatrix} \begin{bmatrix} s_{\mathrm{W}} \\ s_{\mathrm{L}} \\ s_{\mathrm{G}} \end{bmatrix}$$

This model was discretized with a sampling time of 0.01 minutes or 0.6 seconds to give:

$$\begin{bmatrix} h_{\mathrm{W}} \\ h_{\mathrm{L}} \\ p \end{bmatrix} = \begin{bmatrix} -\dfrac{0.000825z^{-1}}{1-z^{-1}} & 0 & \dfrac{0.0000650z^{-1}}{1-z^{-1}} \\ -\dfrac{0.00391z^{-1}}{1-z^{-1}} & -\dfrac{0.00512z^{-1}}{1-z^{-1}} & \dfrac{0.000846z^{-1}}{1-z^{-1}} \\ 0 & 0 & -\dfrac{0.01197z^{-1}}{1-0.995z^{-1}} \end{bmatrix} \begin{bmatrix} s_{\mathrm{W}} \\ s_{\mathrm{L}} \\ s_{\mathrm{G}} \end{bmatrix}$$

The following list shows the relationship between the variables of the objective function equation and the variables of the model:

- y_1 output 1……………..…… h_{W} height of water
- y_2 output 2……………..…… h_{L} height of oil
- y_3 output 3……………..…… p pressure
- u_1 control action 1…………….. s_{W} opening of water valve
- u_2 control action 2……………. s_{L} opening of oil valve
- u_3 control action 3……………. s_P opening of gas valve

The linearized model implies the use of deviation variables (null initial conditions). Consequently the changes in the reference are deviations around the original steady state of the nonlinear model. The trajectory of the reference (or set point) is seen in Table 6.1:

Table 6.1 Variations in magnitude and time of change of set point

	Time (minutes)	Initial Value	Final Value
h_W	25	0	0.1
h_L	50	0	0.1
p	75	0	0.1

6.2 Design of Predictive Controller

Following the development of Chapter 4 a left MFD and a right irreducible MFD are created for the model.

$$\boldsymbol{B}_L(z^{-1}) = \begin{bmatrix} -0.000825z^{-1} & 0 & 0.0000650z^{-1} \\ -0.00391z^{-1} & -0.00512z^{-1} & 0.000846z^{-1} \\ 0 & 0 & -0.0120z^{-1} \end{bmatrix}$$

$$\boldsymbol{A}_L(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 & 0 \\ 0 & 1-z^{-1} & 0 \\ 0 & 0 & 1-0.995z^{-1} \end{bmatrix}$$

$$\boldsymbol{B}_R(z^{-1}) = \begin{bmatrix} -0.000825z^{-1} & 0 & -0.0000651z^{-1} \\ -0.00391z^{-1} & -0.00512z^{-1} & -0.000842z^{-1} \\ 0 & 0 & -0.0120z^{-1} \end{bmatrix}$$

$$\boldsymbol{A}_R(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 & -0.000396 \\ 0 & 1-z^{-1} & -0.000525 \\ 0 & 0 & -(1-0.995z^{-1}) \end{bmatrix}$$

The determinant of $\boldsymbol{B}_R$ has no relevant roots while the determinant of $\boldsymbol{A}_R$ has the following set of relevant roots: $\{1, 1, 0.995\}$. Recall that because the system has same number of inputs as outputs these roots correspond to the zeros of $\boldsymbol{G}_{yw}$ and $\boldsymbol{G}_{uw}$ respectively.

6.3 Case 1: First Tuning

To tune the controller many different combinations of tuning parameters were tested and most of them generated unstable systems. Table 6.2 lists a set of parameters for which stability was verified.

Table 6.2 Tuning parameters for Case 1

	h_W	h_L	p
Ny	20	20	20
Nu	8	8	8
λ	10	8	5
ϕ	5	3	7

Matrices ***R*** and ***S*** are:

$$\boldsymbol{R}(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 & 0 \\ 0 & 1-z^{-1} & 0 \\ 0 & 0 & 1-z^{-1} \end{bmatrix}$$

$$\boldsymbol{S}(z^{-1}) = \begin{bmatrix} -1.2 + 1.118z^{-1} & -3.004 + 2.797z^{-1} & -0.1537 + 0.1433\, z^{-1} \\ 0.1073 - 0.1004z^{-1} & -4.953 + 4.612\, z^{-1} & 0.2454 + 0.2288z^{-1} \\ 0.05104 - 0.04703z^{-1} & 0.388 - 0.3574\, z^{-1} & -13.98 + 12.86\, z^{-1} \end{bmatrix}$$

and the resulting matrix Ω is

$$\boldsymbol{\Omega}(z^{-1}) = \boldsymbol{R}(z^{-1})\boldsymbol{A}_R(z^{-1}) + \boldsymbol{S}(z^{-1})\boldsymbol{B}_R(z^{-1}) =$$
$$\begin{bmatrix} 1-1.987z^{-1}+0.9882z^{-2} & 0.01538z^{-1}-0.01432z^{-2} & -0.0003962+0.001164z^{-1}-0.0007128z^{-2} \\ 0.01923z^{-1}-0.01790z^{-2} & 1-1.975z^{-1}+0.9764z^{-2} & -0.0005247+0.001752z^{-1}-0.001138z^{-2} \\ -0.001555z^{-1}+0.001433z^{-2} & -0.001987z^{-1}+0.001830z^{-2} & -1+1.827z^{-1}-0.8408z^{-2} \end{bmatrix}$$

The determinant of $\boldsymbol{\Omega}(z^{-1})$ is

$$\det(\boldsymbol{\Omega}(z^{-1})) = \frac{-z^6+5.7893z^5-13.9692z^4+17.9827z^3-13.0261z^2+5.0343z-0.81101}{z^6}$$

whose relevant roots are

$$p_\Omega = \{0.9137 \pm 0.0772i, 0.9813 \pm 0.0471i, 0.9996 + 0.0069i\}.$$

All roots lie within the unitary circle and the system is stable. Matrix $\boldsymbol{T}$ is seen next:

$$\begin{aligned} T(1,1) = {} & -0.0004 - 0.0008z - 0.0012z^2 - 0.0016z^3 - 0.002z^4 - 0.0024z^5 - 0.0028z^6 - 0.0031z^7 - 0.0035z^8 \\ & - 0.0039z^9 - 0.0043z^{10} - 0.0047z^{11} - 0.0051z^{12} - 0.0055z^{13} - 0.0058z^{14} - 0.0062z^{15} \\ & - 0.0066z^{16} - 0.007z^{17} - 0.0074z^{18} - 0.0077z^{19} \end{aligned}$$

$$T(2,1) = 0.000007 + 0.00002z + 0.00004z^2 + 0.00007z^3 + 0.0001z^4 + 0.00014z^5 + 0.00018z^6 + 0.00022z^7 + 0.00027z^8$$
$$+ 0.0003z^9 + 0.00036z^{10} + 0.0004z^{11} + 0.00045z^{12} + 0.00049z^{13} + 0.00054z^{14} + 0.00058z^{15} + 0.0006z^{16}$$
$$+ 0.0007z^{17} + 0.0007z^{18} + 0.00076z^{19}$$

$$T(3,1) = 0.000054 + 0.0001z + 0.00014z^2 + 0.00016z^3 + 0.00018z^4 + 0.0002z^5 + 0.0002z^6 + 0.00021z^7 + 0.00021z^8$$
$$+ 0.00022z^9 + 0.00022z^{10} + 0.00022z^{11} + 0.0002z^{12} + 0.00023z^{13} + 0.000523z^{14} + 0.00023z^{15} + 0.00024z^{16}$$
$$+ 0.00024z^{17} + 0.00024z^{18} + 0.00025z^{19}$$

$$T(1,2) = -0.0011 - 0.0022z - 0.0033z^2 - 0.0043z^3 - 0.005z^4 - 0.0063z^5 - 0.007z^6 - 0.008z^7 - 0.009z^8$$
$$- 0.01z^9 - 0.01z^{10} - 0.012z^{11} - 0.013z^{12} - 0.014z^{13} - 0.015z^{14} - 0.016z^{15}$$
$$- 0.016z^{16} - 0.017z^{17} - 0.018z^{18} - 0.019z^{19}$$

$$T(2,2) = -0.0019 - 0.0036z - 0.0054z^2 - 0.007z^3 - 0.0087z^4 - 0.01z^5 - 0.012z^6 - 0.013z^7 - 0.015z^8$$
$$- 0.016z^9 - 0.018z^{10} - 0.019z^{11} - 0.021z^{12} - 0.013z^{13} - 0.013z^{14} - 0.014z^{15}$$
$$- 0.027z^{16} - 0.029z^{17} - 0.03z^{18} - 0.032z^{19}$$

$$T(3,2) = 0.0004 + 0.00077z + 0.001z^2 + 0.0013z^3 + 0.0014z^4 + 0.0015z^5 + 0.0016z^6 + 0.0016z^7 + 0.0016z^8$$
$$+ 0.0016z^9 + 0.0017z^{10} + 0.0017z^{11} + 0.0017z^{12} + 0.0017z^{13} + 0.0018z^{14} + 0.0018z^{15}$$
$$+ 0.0018z^{16} + 0.0018z^{17} + 0.0018z^{18} + 0.0019z^{19}$$

$$T(1,3) = -0.000012 - 0.000035z - 0.000067z^2 - 0.00011z^3 - 0.00016z^4 - 0.0002z^5 - 0.00028z^6 - 0.00034z^7 - 0.00041z^8$$
$$- 0.00047z^9 - 0.00054z^{10} - 0.0006z^{11} - 0.00067z^{12} - 0.00073z^{13} - 0.0008z^{14} - 0.0009z^{15} - 0.0009z^{16}$$
$$- 0.001z^{17} - 0.001z^{18} - 0.001z^{19}$$

$$T(2,3) = -0.00002 - 0.000056z - 0.0001z^2 - 0.00017z^3 - 0.00025z^4 - 0.00034z^5 - 0.00044z^6 - 0.00055z^7 - 0.00065z^8$$
$$- 0.00076z^9 - 0.00086z^{10} - 0.00096z^{11} - 0.001z^{12} - 0.0012z^{13} - 0.0013z^{14} - 0.0014z^{15}$$
$$- 0.0015z^{16} - 0.0016z^{17} - 0.0017z^{18} - 0.0018z^{19}$$

$$T(3,3) = -0.014 - 0.026z - 0.035z^2 - 0.043z^3 - 0.048z^4 - 0.053z^5 - 0.055z^6 - 0.057z^7 - 0.058z^8$$
$$- 0.059z^9 - 0.061z^{10} - 0.062z^{11} - 0.064z^{12} - 0.065z^{13} - 0.066z^{14} - 0.068z^{15}$$
$$- 0.069z^{16} - 0.07z^{17} - 0.072z^{18} - 0.073z^{19}$$

Figure 6.1 shows the results of the simulation of the linear model for the height of water. Excessive oscillation occurs and this can be explained by the fact that a very small sampling time was chosen in order to deal with the very fast dynamics of the pressure. Besides this, during the design of the controller, the greatest weight was placed on the first control action, Δu_1 (opening of the valve of water), as λ_1 is 10 and less importance was given to the error between the reference (set point) and the output (water level), r_1-y_1, as ϕ_1 is 5. This causes a restriction on the opening of the water valve while the level is free to reach higher values of error.

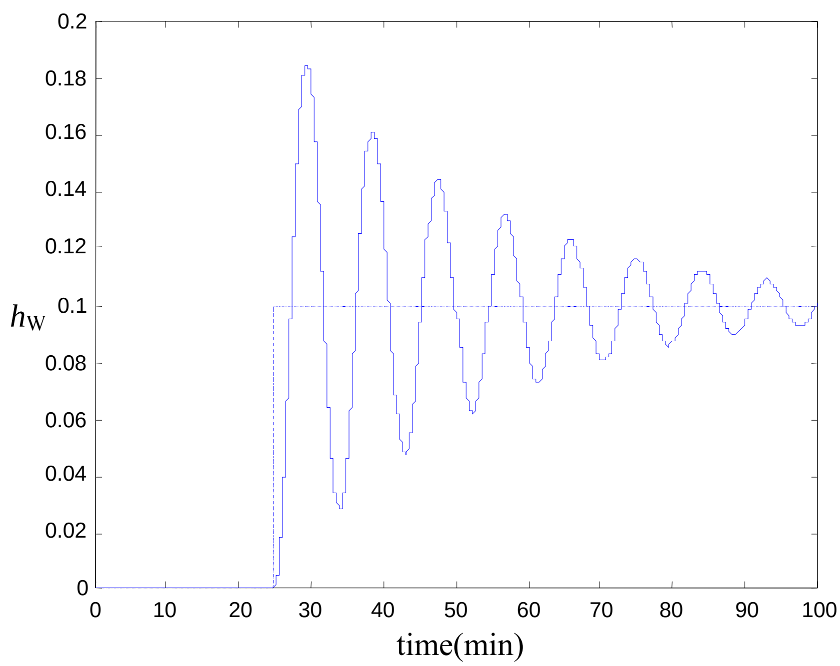

Figure 6.1 Height of Water

The multivariable aspect of the controller can be observed by the apparent absence of perturbations in the level of water when the height of oil changes (set point change at 50 minutes) and when the pressure changes (set point change at 75 minutes). Both these changes will promote an increase in the pressure of the separator, which in turn would increase the outflow of water. Consequently a decrease in the water level would be seen. Due to the predictive and multivariable nature of the predictive control the water valve (see Figure 6.4) closes while the oil valve is closing thus avoiding the decrease in the level. The end result is a smooth curve of height of water.

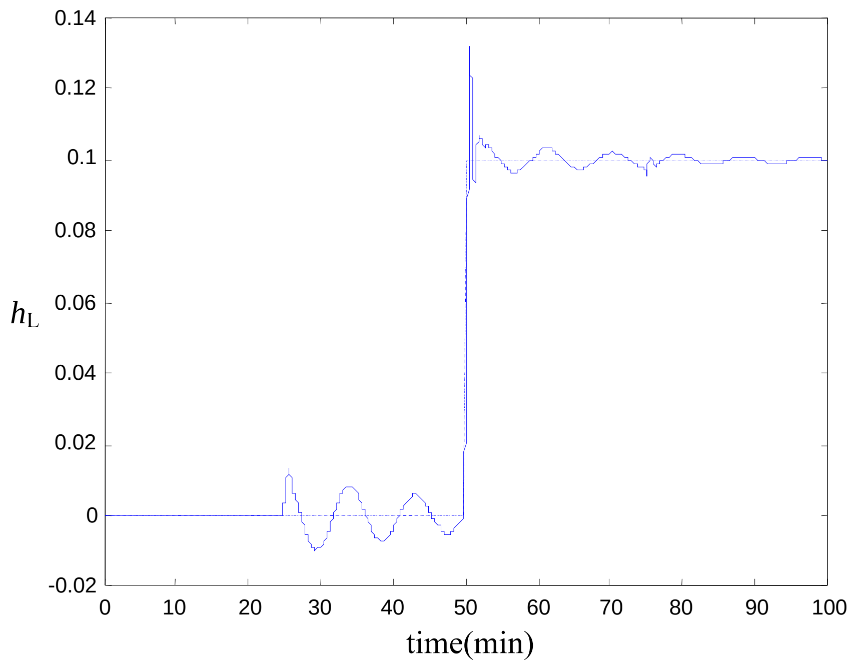

Figure 6.2 Height of Oil

The height of oil oscillates much less. If we recognize that fluctuations in the height of water directly affect the flow of oil over the weir (which is the feed to the oil chamber) then we see that most of the oscillations were caused by the behavior of the water level.

The peak seen at 50 minutes is caused by the change in the set point of level of oil. The level rapidly reduces and if it were not for the oscillations in the height of water a final value would soon have been reached. Thus we may conclude that the control of oil level had a good performance.

The results for the pressure show an excellent tracking of the set point as seen in Figure 6.3. Because the change in heights of oil and water have very little effect on the pressure this variable is almost independent of the others. However the opposite is not true, the pressure has a very strong effect on the heights since the flow rates leaving the vessel are a direct function of the internal pressure of the separator.

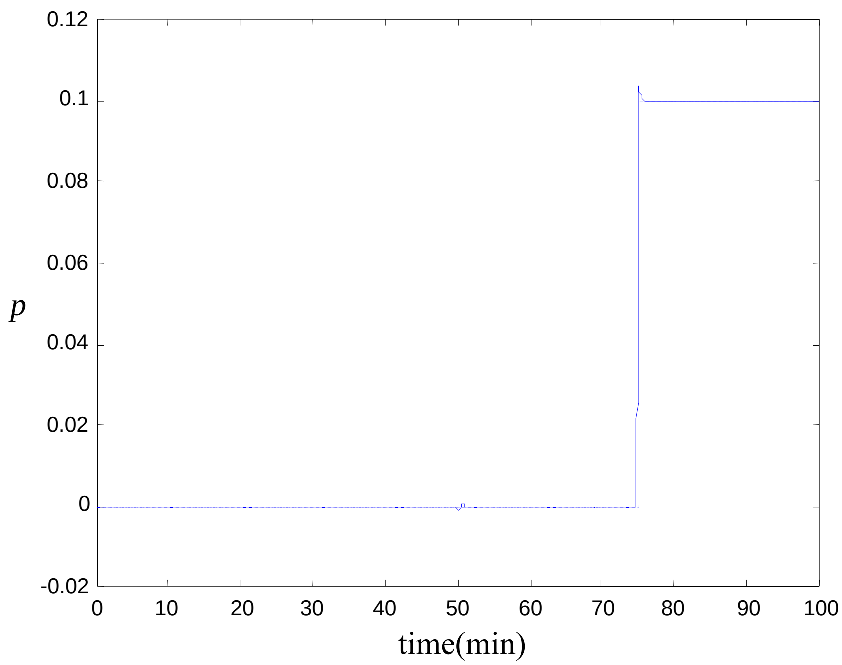

Figure 6.3 Pressure

The independence of the pressure added to the multivariable and predictive aspects of the controller create the expectation of a good performance for the control of pressure. The opening of the valves are seen in Figures 6.4, 6.5 and 6.6.

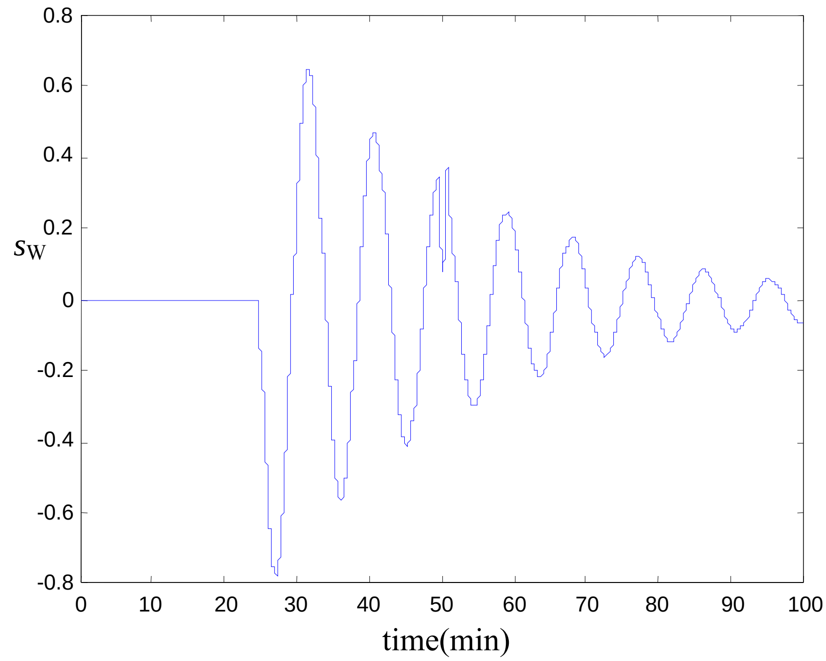

Figure 6.4 Opening of the Water Valve

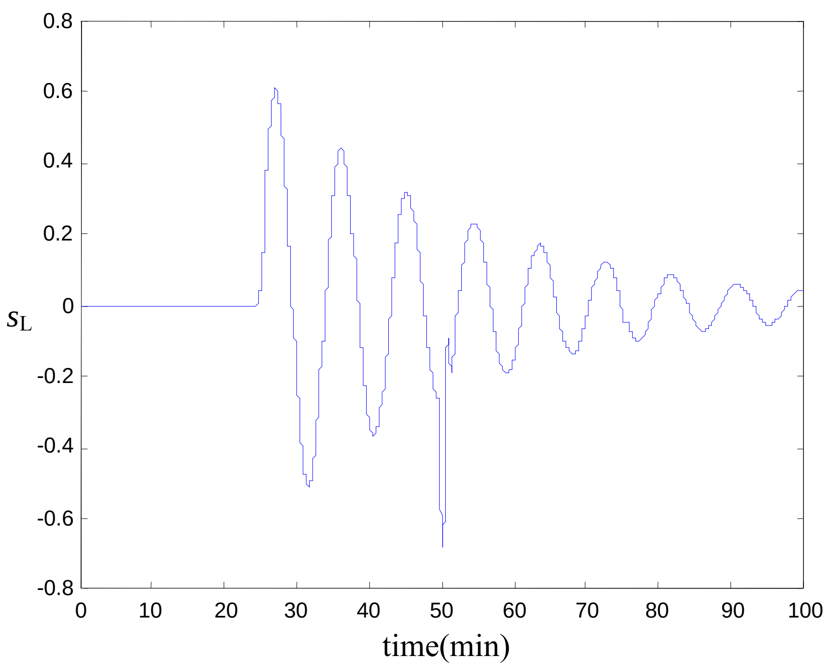

Figure 6.5 Opening of the Oil Valve

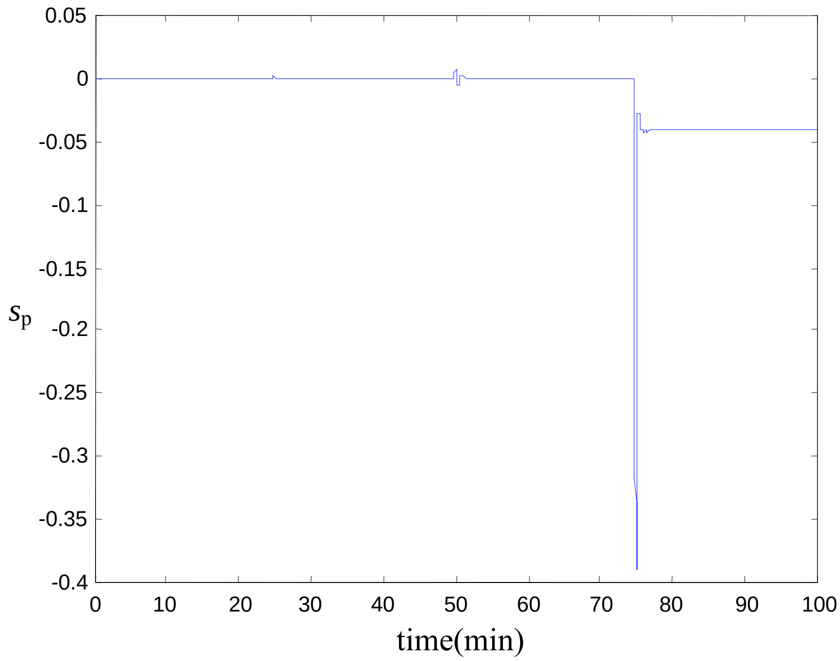

Figure 6.6 Opening of the Gas Valve

Thus the control of the oil level and pressure gave good results. The control of water level, however, did not show a good performance and another set of parameters is searched next. Looking for a better performance of the predictive controller the weights are changed. The aim of this new tuning is to reduce the oscillations of the level of water while maintaining the performance of the other two variables.

6.4 Case 1: Second Tuning

A second set of parameters was tested. Again, a series of different tuning parameters were tested and from the set of parameters that resulted in a stable closed-loop systems one was chosen. Table 6.3 shows the set of parameters used in this case.

Table 6.3 Tuning parameters for the second tuning of Case 1

	h_W	h_L	p
Ny	20	20	20
Nu	9	9	9
λ	3	5	3
ϕ	5	1	2

The weight on the control action of the water valve was reduced (from 10 to 3). The weights on the other control actions were also reduced. The weight on the error between the level of water and the set point, ϕ_1, was maintained. Thus the relative importance of this term in the objective function equation is higher and an improvement in the performance of the control of water level is expected while for the other variables, oil level and pressure, it should deteriorate. Control actions should be more intense.

Matrices $\boldsymbol{R}$ and $\boldsymbol{T}$ are:

$$\boldsymbol{R}(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 & 0 \\ 0 & 1-z^{-1} & 0 \\ 0 & 0 & 1-z^{-1} \end{bmatrix}$$

$$\boldsymbol{S}(z^{-1}) = \begin{bmatrix} -3.914 + 3.646z^{-1} & -3.444 + 3.207z^{-1} & -0.1445 + 0.1348z^{-1} \\ 0.2047 - 0.1913z^{-1} & -2.757 + 2.567z^{-1} & -0.1058 + 0.09866z^{-1} \\ 0.1245 - 0.1152z^{-1} & 0.3421 - 0.3168\, z^{-1} & -10.25 + 9.470\, z^{-1} \end{bmatrix}$$

and matrix Ω is giving by

$$\boldsymbol{\Omega}(z^{-1}) = \boldsymbol{R}(z^{-1})\boldsymbol{A}_R(z^{-1}) + \boldsymbol{S}(z^{-1})\boldsymbol{B}_R(z^{-1})$$

$$= \begin{bmatrix} 1-1.983z^{-1}+0.9845z^{-2} & 0.01764z^{-1}-0.01642z^{-2} & -0.003962+0.001821z^{-1}-0.001324z^{-2} \\ 0.01058z^{-1}-0.09854z^{-2} & 1-1.986z^{-1}+0.9869z^{-2} & -0.0005247+0.001567z^{-1}-0.0009685z^{-2} \\ -0.001437z^{-1}+0.001330z^{-2} & -0.001752z^{-1}+0.001622z^{-2} & -1+1.8720z^{-1}-0.8814z^{-2} \end{bmatrix}$$

$$\det(\boldsymbol{\Omega}(z^{-1})) = \frac{-z^6+5.841z^5-14.22z^4+18.47z^3-13.50z^2+5.266z-0.8561}{z^6}$$

whose relevant roots are

$$p_\Omega = \{0.9360 \ \pm\ 0.0725i, 0.9855 \ \pm\ 0.0423i, 0.9992+\ 0.0106i\}\,.$$

This lets us conclude that the system is stable. Matrix $\boldsymbol{T}$ is seen next:

$$\begin{aligned} T(1,1) = {} & -0.001354 \ -0.00269z-0.004z^2-0.0053z^3-0.00658z^4-0.00785z^5-0.0091z^6-0.0104z^7-0.0116z^8 \\ & -0.0128z^9-0.014z^{10}-0.015z^{11}-0.0165z^{12}-0.0178z^{13}-0.02z^{14}-0.02z^{15} \\ & -0.0215z^{16}-0.0227z^{17}-0.024z^{18}-0.025z^{19} \end{aligned}$$

$$\begin{aligned} T(2,1) = {} & 0.0000137 \ +0.00004z+0.000078z^2+0.00013z^3+0.00019z^4+0.00025z^5+0.00033z^6+0.0004z^7+0.0005z^8 \\ & +0.00058z^9+0.00067z^{10}+0.00076z^{11}+0.00085z^{12}+0.00094z^{13}+0.001z^{14}+0.0011z^{15}+0.0012z^{16} \\ & +0.0013z^{17}+0.0014z^{18}+0.0015z^{19} \end{aligned}$$

$$T(3,1) = 0.000095 + 0.00018z + 0.00025z^2 + 0.0003z^3 + 0.00035z^4 + 0.00039z^5 + 0.00042z^6 + 0.00045z^7 + 0.00047z^8$$
$$+ 0.00049z^9 + 0.0005z^{10} + 0.0005z^{11} + 0.0005z^{12} + 0.00056z^{13} + 0.00058z^{14} + 0.0006z^{15} + 0.00062z^{16}$$
$$+ 0.0006z^{17} + 0.00066z^{18} + 0.00067z^{19}$$

$$T(1,2) = -0.0013 - 0.0025z - 0.0037z^2 - 0.0049z^3 - 0.006z^4 - 0.0071z^5 - 0.0082z^6 - 0.009z^7 - 0.01z^8$$
$$- 0.011z^9 - 0.0125z^{10} - 0.014z^{11} - 0.015z^{12} - 0.0157z^{13} - 0.017z^{14} - 0.018z^{15}$$
$$- 0.019z^{16} - 0.02z^{17} - 0.021z^{18} - 0.022z^{19}$$

$$T(2,2) = -0.001 - 0.002z - 0.003z^2 - 0.004z^3 - 0.0048z^4 - 0.0057z^5 - 0.0065z^6 - 0.0074z^7 - 0.008z^8$$
$$- 0.009z^9 - 0.01z^{10} - 0.01z^{11} - 0.012z^{12} - 0.013z^{13} - 0.013z^{14} - 0.014z^{15}$$
$$- 0.015z^{16} - 0.016z^{17} - 0.017z^{18} - 0.018z^{19}$$

$$T(3,2) = 0.00025 + 0.00046z + 0.00064z^2 + 0.0008z^3 + 0.0009z^4 + 0.001z^5 + 0.001z^6 + 0.0012z^7 + 0.0013z^8$$
$$+ 0.0013z^9 + 0.0014z^{10} + 0.001z^{11} + 0.0015z^{12} + 0.0015z^{13} + 0.0016z^{14} + 0.0017z^{15}$$
$$+ 0.0017z^{16} + 0.0018z^{17} + 0.0018z^{18} + 0.0019z^{19}$$

$$T(1,3) = -0.00001 - 0.00003z - 0.00006z^2 - 0.0001z^3 - 0.00014z^4 - 0.00019z^5 - 0.00025z^6 - 0.00031z^7 - 0.00037z^8$$
$$- 0.00044z^9 - 0.0005z^{10} - 0.00056z^{11} - 0.00063z^{12} - 0.00069z^{13} - 0.00075z^{14} - 0.0008z^{15} - 0.00088z^{16}$$
$$- 0.00094z^{17} - 0.001z^{18} - 0.001z^{19}$$

$$T(2,3) = -0.000008 - 0.000023z - 0.000044z^2 - 0.00007z^3 - 0.0001z^4 - 0.00014z^5 - 0.00018z^6 - 0.0002z^7 - 0.00027z^8$$
$$- 0.00032z^9 - 0.00037z^{10} - 0.0004z^{11} - 0.00046z^{12} - 0.0005z^{13} - 0.00055z^{14} - 0.00059z^{15}$$
$$- 0.00064z^{16} - 0.00069z^{17} - 0.00073z^{18} - 0.0008z^{19}$$

$$T(3,3) = -0.0071 - 0.013z - 0.019z^2 - 0.023z^3 - 0.027z^4 - 0.031z^5 - 0.034z^6 - 0.036z^7 - 0.038z^8$$
$$- 0.04z^9 - 0.042z^{10} - 0.044z^{11} - 0.046z^{12} - 0.048z^{13} - 0.05z^{14} - 0.052z^{15}$$
$$- 0.054z^{16} - 0.056z^{17} - 0.058z^{18} - 0.06z^{19}$$

By comparison to Figure 6.1 it is clear how this tuning causes less oscillations in the height of water. The amplitudes are smaller and the attenuation is faster.

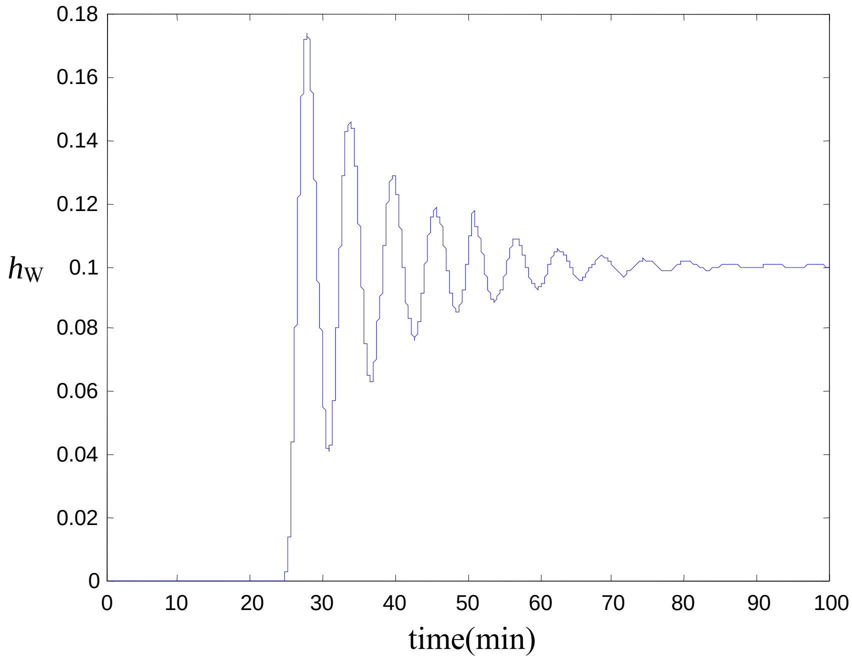

Figure 6.7 Height of Water

Figure 6.8 shows the results for the height of oil. If we compare it to Figure 6.2 we see that the oscillations have increased. Bearing in mind that with the improvement of the control of water level the oscillations in the flow rate of oil to the oil chamber (Q_{weir}) will decrease, then it is easy to conclude that the performance of the control of oil level has worsened. This is an expected downside to the change made in the parameters.

For the pressure this new tuning has had little effect. The results are very close to the last tuning (see Figure 6.9). Once again this is caused by the independence of the pressure with relationship to the other variables.

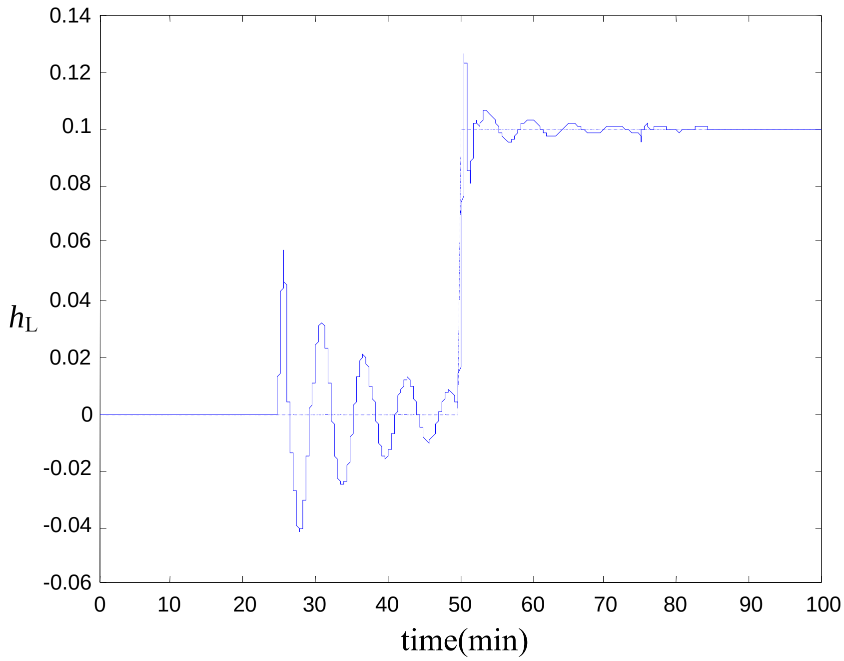

Figure 6.8 Height of Oil

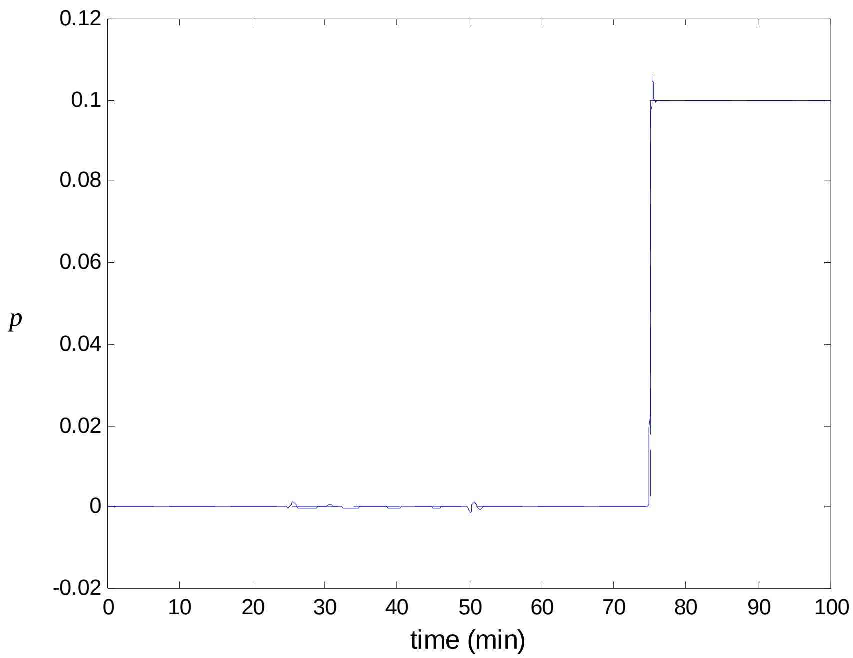

Figure 6.9 Pressure

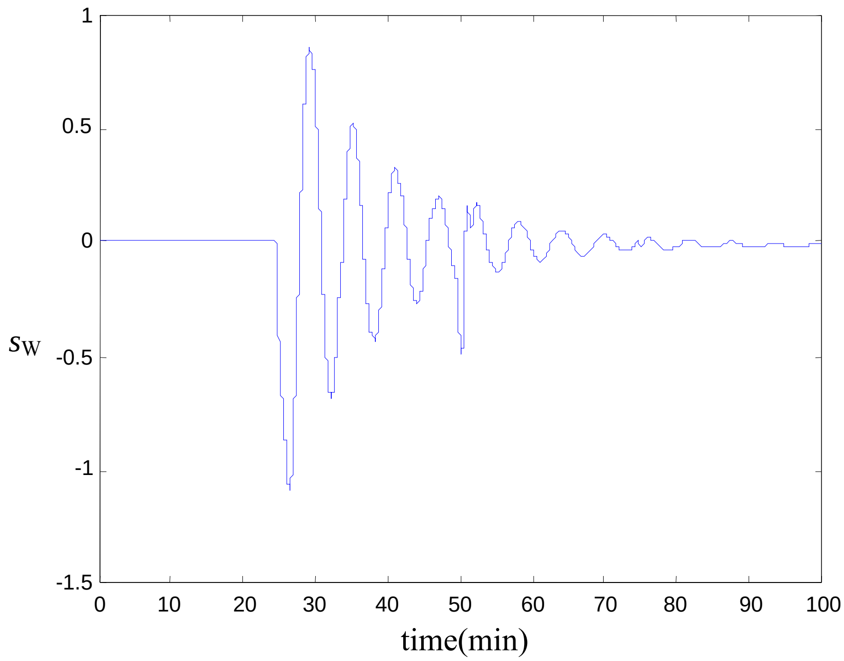

Figure 6.10 Opening of Water Valve

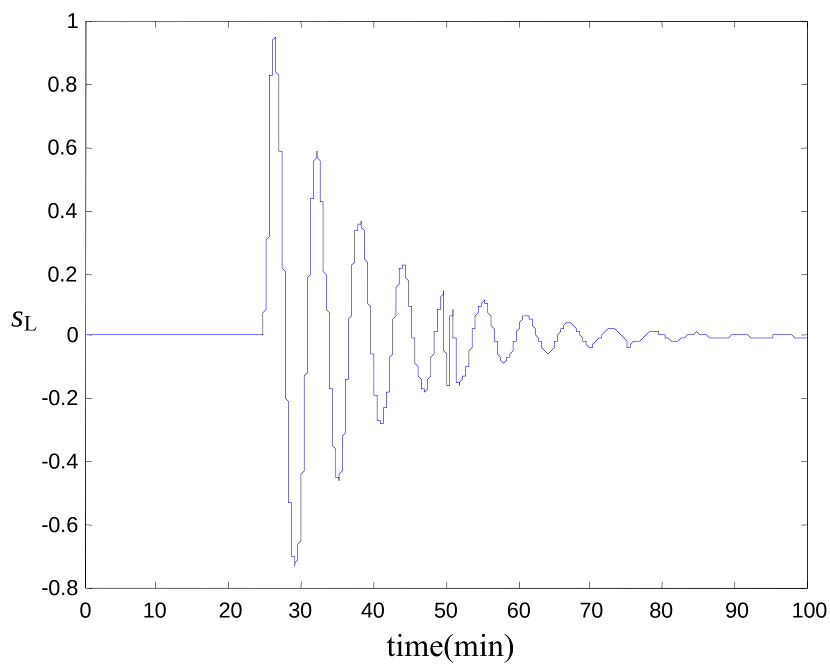

Figure 6.11 Opening of Oil Valve

The control action for the opening of the water valve (see Figure 6.10) is more aggressive while the opposite has happened to the oil valve and the gas valve. This behavior was also expected from the modification in parameters.

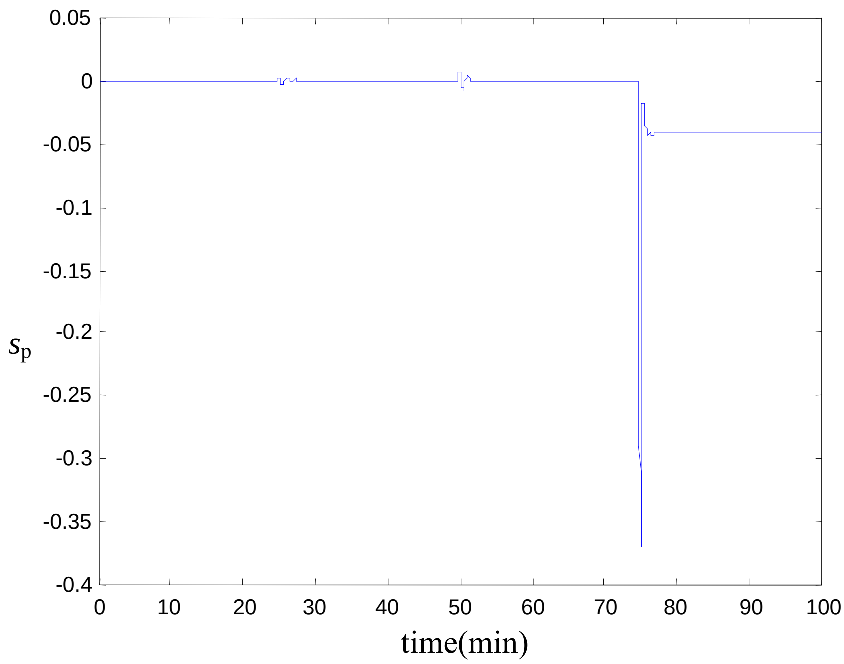

Figure 6.12 Opening of the valve of Gas

It was very difficult to tune the 3x3 predictive control for the separator. The above tuning does not show a good performance. The reason for this lies in the fact that the transfer function relating the opening of the gas valve to the pressure has a time constant very small (τ=2) if compared to the other transfer functions. That way a very small sampling time was required (Ts=0.01minutes), which, for the horizons used (Ny=20 and Nu=9), caused the loss of the predictiveness of the predictive control once 20 sampling times correspond to 1.2 seconds. The controller became very sensitive to the tuning parameters and only a few sets of these generated stable closed-loop systems.

6.5 Case 2

In this case the control of pressure was left out of the loop and a predictive controller was designed for a system with two inputs (openings of water and oil valve) and two outputs (height of water and oil).

$$\begin{bmatrix} h_W \\ h_L \end{bmatrix} = \begin{bmatrix} -\dfrac{17}{206s+1} & 0 \\ -\dfrac{126}{322s+1} & -\dfrac{169}{330s+1} \end{bmatrix} \begin{bmatrix} s_W \\ s_L \end{bmatrix}$$

The absence of the pressure in the transfer matrix allowed the sampling time to be much higher (0.1 minutes). With this new sampling time the model was discretized yielding:

$$\begin{bmatrix} h_W \\ h_L \end{bmatrix} = \begin{bmatrix} -\dfrac{0.00825z^{-1}}{1-z^{-1}} & 0 \\ -\dfrac{0.0391z^{-1}}{1-z^{-1}} & -\dfrac{0.0512z^{-1}}{1-z^{-1}} \end{bmatrix} \begin{bmatrix} s_W \\ s_L \end{bmatrix}$$

A sensitivity analysis was done to measure the influence of the prediction horizon on the location of the closed-loop poles. Table 6.4 shows the control horizon and weights adopted.

Table 6.4 Tuning parameters for Case 2

	h_W	h_L
Nu	9	9
λ	1	10
ϕ	10	5

The prediction horizon and the relevant roots of $\det(\mathbf{\Omega}(z^{-1}))$ are seen on Table 6.5. It is seen that as the prediction horizon increases the location of the roots of $\det(\mathbf{\Omega}(z^{-1}))$ move (slightly) towards the origin. Because the change in the position of the roots of $\det(\mathbf{\Omega}(z^{-1}))$ are small with prediction horizon, the value of *Ny*=20 was chosen for the design of the controller.

Table 6.5 Variations of relevant roots of $\det(\mathbf{\Omega}(z^{-1}))$ with prediction horizon

Ny	Roots of $\det(\mathbf{\Omega}(z^{-1}))$
20	$0.9986\pm0.0138i$ $0.9362\pm0.0726i$
30	$0.9956\pm0.0199i$ $0.9223\pm0.0541i$
40	$0.9906\pm0.0243i$ $0.9205\pm0.0451i$
50	$0.9839\pm0.0260i$ $0.9204\pm0.0432i$
60	$0.9765\pm0.0245i$ $0.9203\pm0.0446i$

Matrices $\boldsymbol{R}$, $\boldsymbol{S}$ and $\mathbf{\Omega}$ are

$$\boldsymbol{R}(z^{-1}) = \begin{bmatrix} 1-z^{-1} & 0 \\ 0 & 1-z^{-1} \end{bmatrix}$$

$$\boldsymbol{S}(z^{-1}) = \begin{bmatrix} -12.84+11.91z^{-1} & -25.82+23.92z^{-1} \\ 1.488-1.393z^{-1} & -3.693+3.425z^{-1} \end{bmatrix}$$

$$\mathbf{\Omega}(z^{-1}) = \begin{bmatrix} 1-1.888z^{-1}+0.8966z^{-2} & 0.1322z^{-1}-0.1225z^{-2} \\ 0.01322z^{-1}-0.01225z^{-2} & 1-1.981z^{-1}+0.9825z^{-2} \end{bmatrix}$$

whose determinant is

$$\det(\mathbf{\Omega}(z^{-1})) = \frac{z^4-3.8695z^3+5.6184z^2+3.6283z+0.8794}{z^4}$$

and the relevant roots of $\det(\mathbf{\Omega}(z^{-1}))$ are

$$p_{\Omega} = \{0.9362 \pm 0.0726i, 0.9986 \pm 0.0138i\}$$

Matrix $\boldsymbol{T}$ is seen next:

$$T(1,1) = -0.00741 - 0.01405z - 0.01998z^2 - 0.02526z^3 - 0.02997z^4 - 0.03416z^5 - 0.03789z^6 - 0.04122z^7 - 0.0442z^8$$
$$- 0.04688z^9 - 0.04956z^{10} - 0.05224z^{11} - 0.05492z^{12} - 0.0576z^{13} - 0.06028z^{14} - 0.06296z^{15} - 0.06564z^{16}$$
$$- 0.06832z^{17} - 0.071z^{18} - 0.07368z^{19}$$

$$T(2,1) = 0.0000995 + 0.00029z + 0.000566z^2 + 0.000916z^3 + 0.001336z^4 + 0.001817z^5 + 0.002352z^6 + 0.002934z^7 + 0.003558z^8$$
$$0.004217z^9 + 0.004876z^{10} + 0.005535z^{11} + 0.006195z^{12} + 0.006854z^{13} + 0.007513z^{14} + 0.008172z^{15} + 0.008831z^{16}$$
$$0.009491z^{17} + 0.01015z^{18} + 0.01081z^{19}$$

$$T(1,2) = -0.01725 - 0.03238z - 0.04558z^2 - 0.05702z^3 - 0.06687z^4 - 0.07531z^5 - 0.08249z^6 - 0.08857z^7 - 0.09369z^8$$
$$- 0.098z^9 - 0.1023z^{10} - 0.1066z^{12} - 0.1109z^{13} - 0.1152z^{14} - 0.1195z^{15} - 0.1238z^{16} - 0.1281z^{17} - 0.1325z^{18} - 0.1368z^{19}$$

$$T(2,2) = -0.00228 - 0.004302z - 0.006088z^2 - 0.00766z^3 - 0.009039z^4 - 0.01024z^5 - 0.0113z^6 - 0.0122z^7 - 0.01302z^8$$
$$- 0.01372z^9 - 0.01443z^{10} - 0.01513z^{11} - 0.01583z^{12} - 0.01653z^{13} - 0.01724z^{14} - 0.01794z^{15} - 0.01864z^{16}$$
$$- 0.01935z^{17} - 0.02005z^{18} - 0.02075z^{19}$$

For the PID control of pressure the tuning parameters were:

Table 6.6 Tuning Parameters of PID

Proportional	0.5
Integral	0.1
Derivative	0

The results for this case are very good. The height of water doesn't show the oscillatory behavior seen in Case 1. From Figure 6.13 the predictive nature of the controller is clearly seen. The height of water starts rising at 10 sampling times, or 1 minute, before the actual change in the set point occurs. A smooth transition to the final value follows and a very small overshoot is seen.

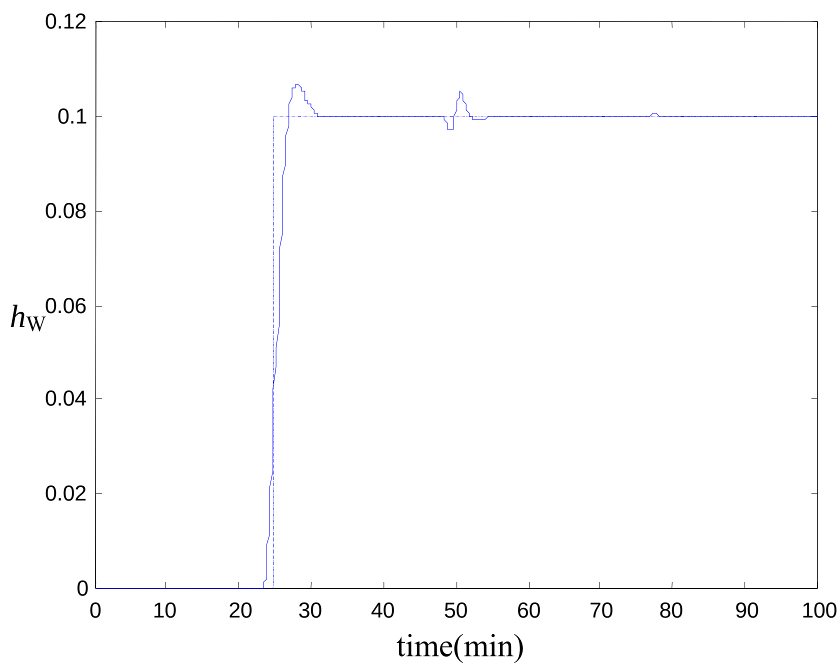

Figure 6.13 Height of Water

The peak seen at 50 minutes is due to the change in height of oil. The increase in oil level elevates the pressure of the vessel and the water valve (Figure 6.16) closes preventively to counteract this higher pressure.

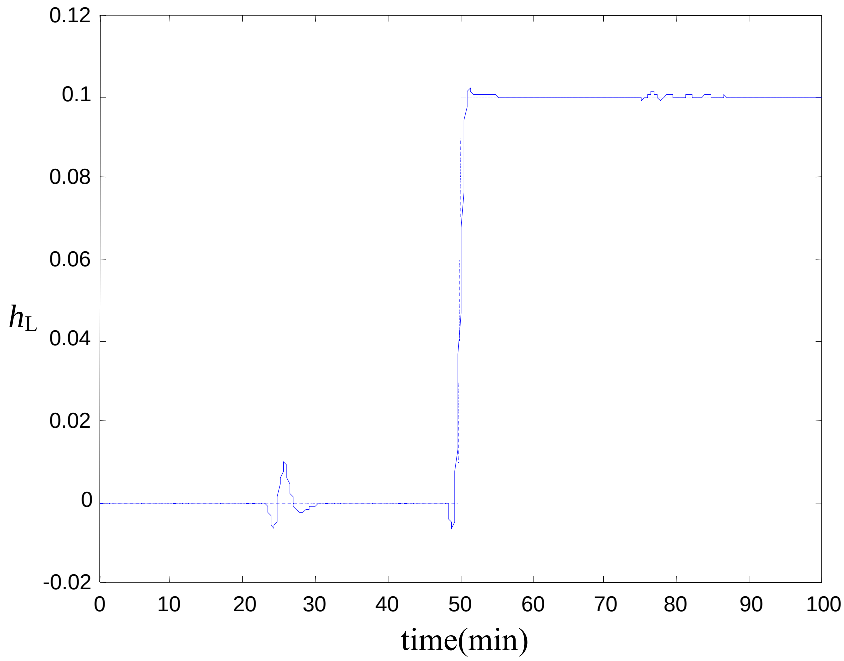

Figure 6.14 Height of Oil

A PID controls the pressure and the response is seen in Figure 6.15.

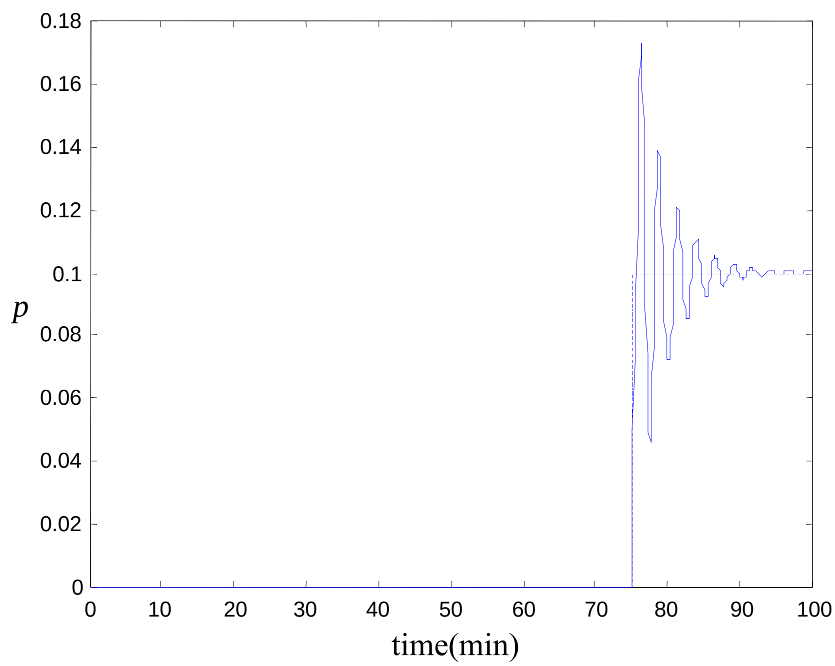

Figure 6.15 Pressure

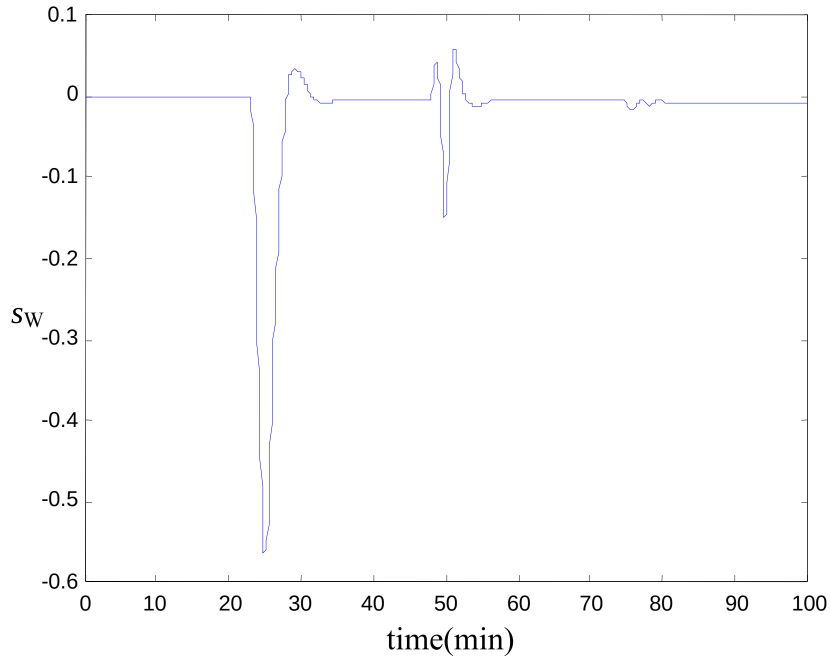

Figure 6.16 Opening of Valve of Water

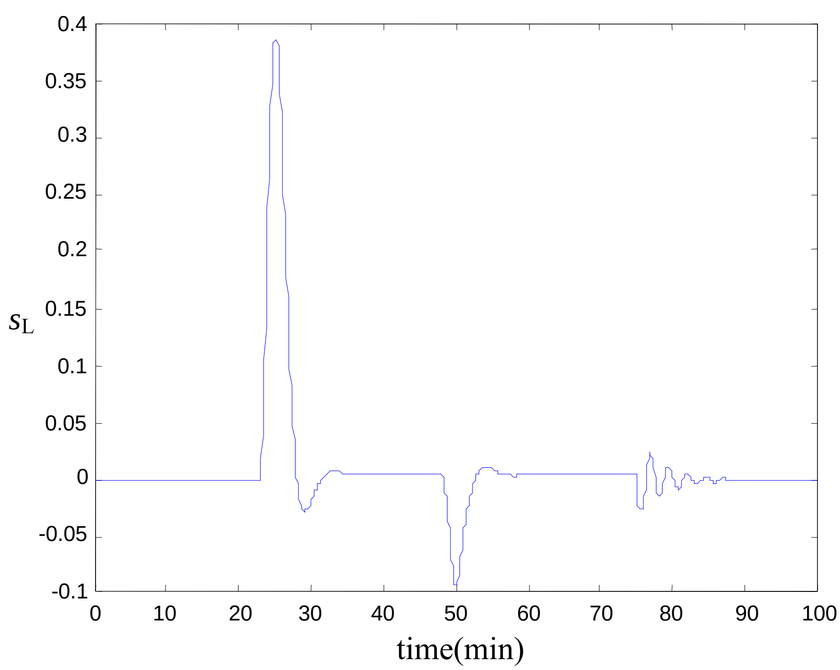

Figure 6.17 Opening of the Valve of Oil

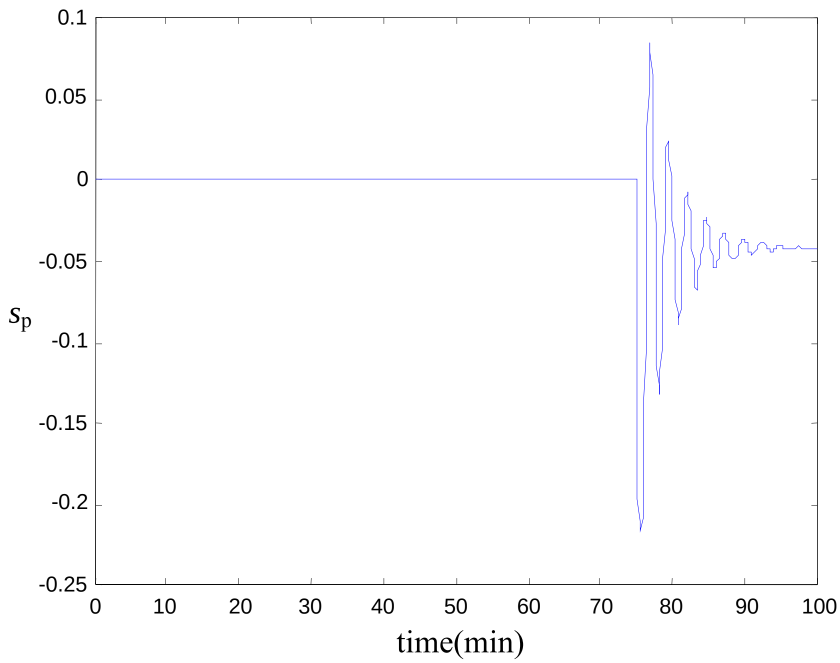

Figure 6.18 Opening of the Valve of Gas

6.6 Conclusion

The design of a predictive controller with three inputs and three outputs for the separator was difficult due to the difference in time constants between the pressure and the levels. Initially a small sampling time was required and the search for a stable system with good performance required many trials. A second case was studied where a predictive controller with two inputs and two outputs was designed for the control of levels while a PID controller was used for the pressure. A sensitivity study was done where the closed-loop poles are calculated for different prediction horizons. Tuning the controller was much easier in this case and a good performance was observed.

CHAPTER 7
SIMULATION AND RESULTS

In this chapter both predictive controllers designed in Chapter 6 are tested against the nonlinear model of the separator by simulating the closed-loop in Simulink of Matlab. The same series of step changes of set point made in Chapter 6 (see Table 1) is used.

7.1 Results of Case 1

Figure 7.1 shows the change in the height of water. The results resemble the simulations done for the linearized model in Chapter 6 (see Figure 6.1). The fluctuations decay very fast and a steady state is reached before the set point of oil height is changed.

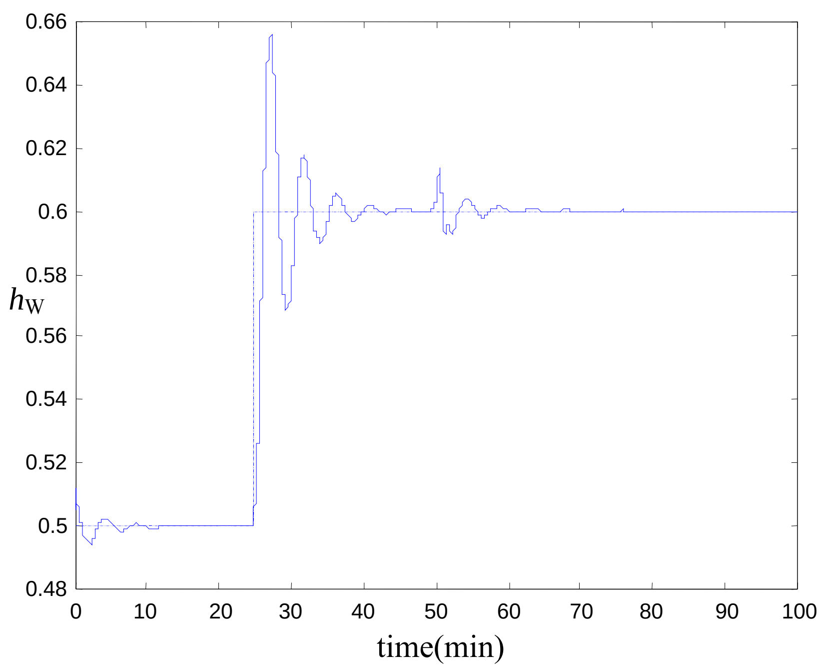

Figure 7.1 Height of Water

At 50 minutes a peak occurs due to the change in height of oil. If we observe the opening of the water valve in Figure 7.4 we see that the controller initially closes the valve to counteract the increase in pressure caused by the increase in oil level. This is done before the change in set point of oil level demonstrating the predictive and multivariable characteristics of the controller. While this peak is almost unnoticed for the linear model it is very evident in this case as it is a consequence of the nonlinearities of the model.

Figure 7.2 shows the response of the oil level. Compared to Figure 6.8 we see that, here, the decay is slightly faster. Once again the nonlinearities are responsible for this change in response. In general terms the performance is good.

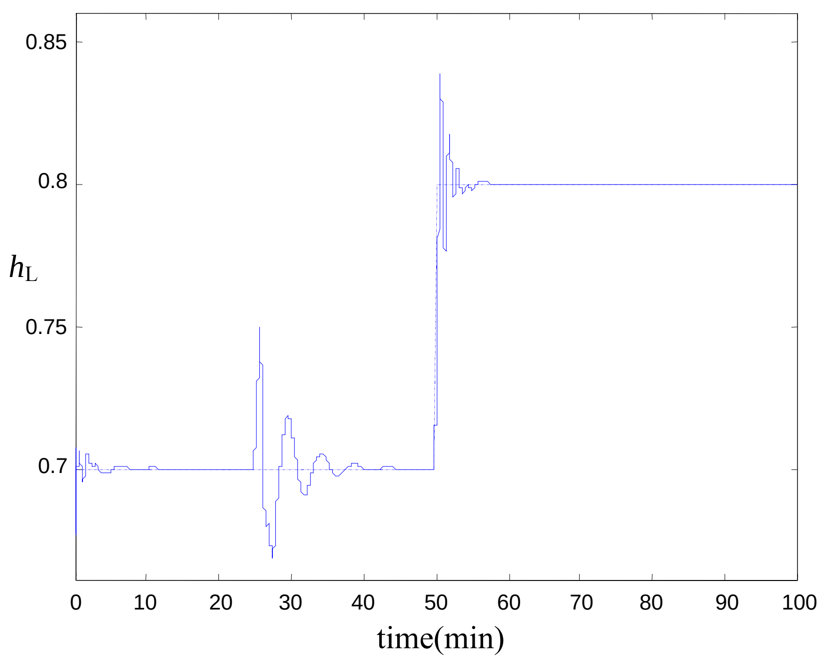

Figure 7.2 Height of Oil

The response of the pressure is very similar to that of the linear model.

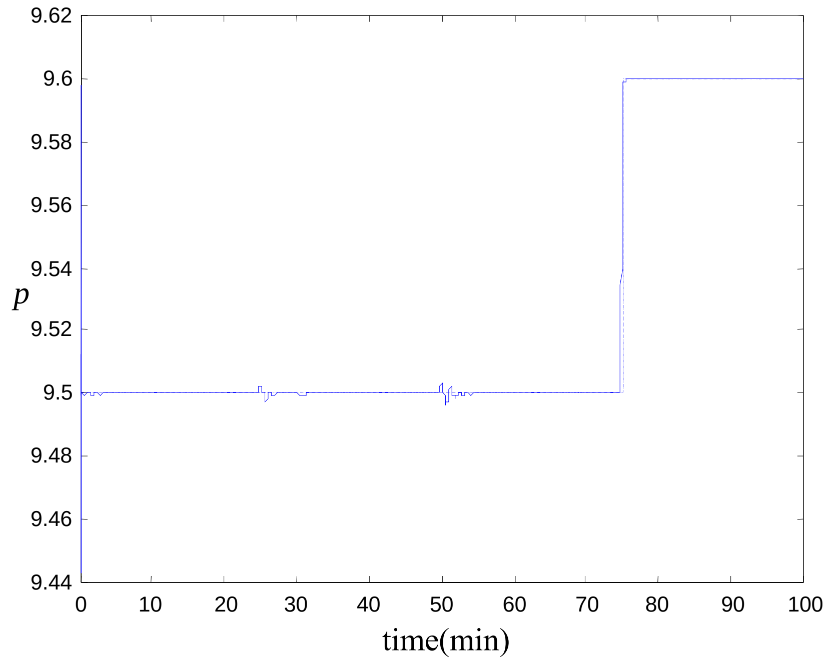

Figure 7.3 Pressure

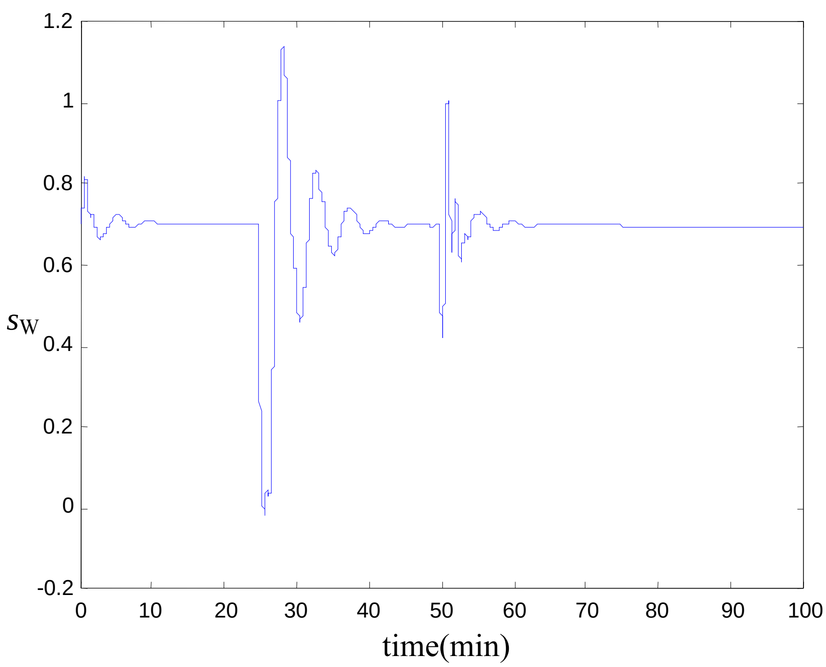

Figure 7.4 Opening of Water Valve

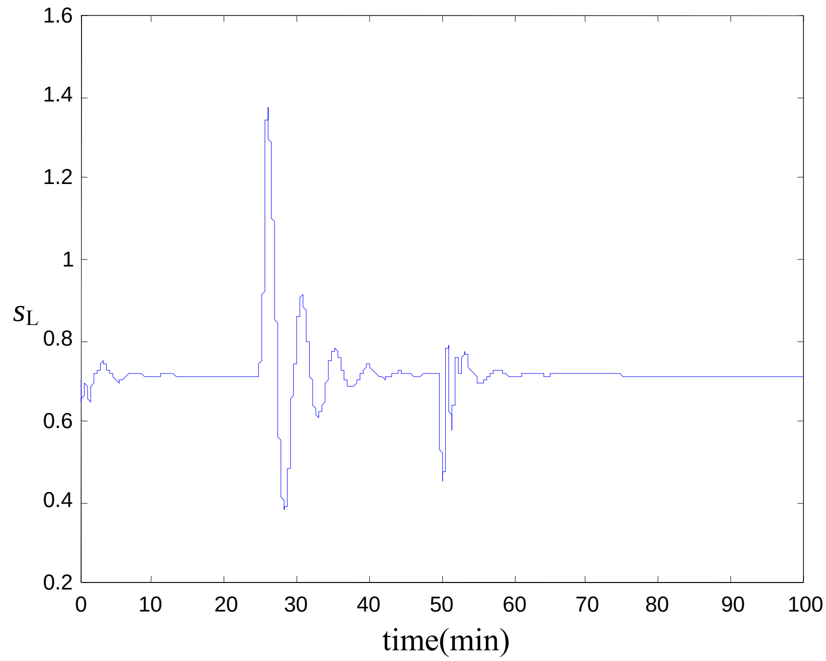

Figure 7.5 Opening of Oil Valve

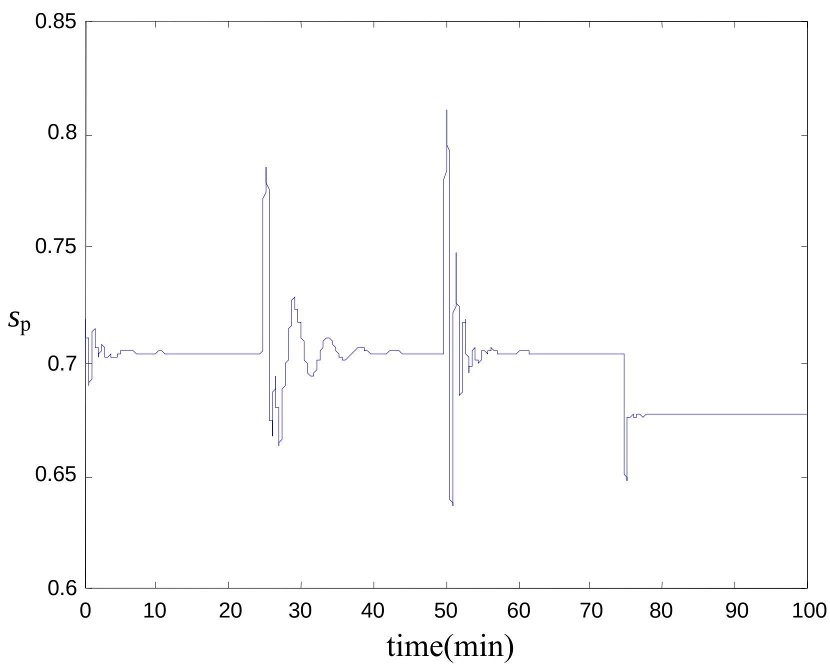

Figure 7.6 Opening of Gas Valve

7.2 Results of Case 2

Figure 7.2 shows the response of the water level. Compared to the linear model (Figure 6.13) the overshoot is smaller while the peak around 50 minutes is very similar.

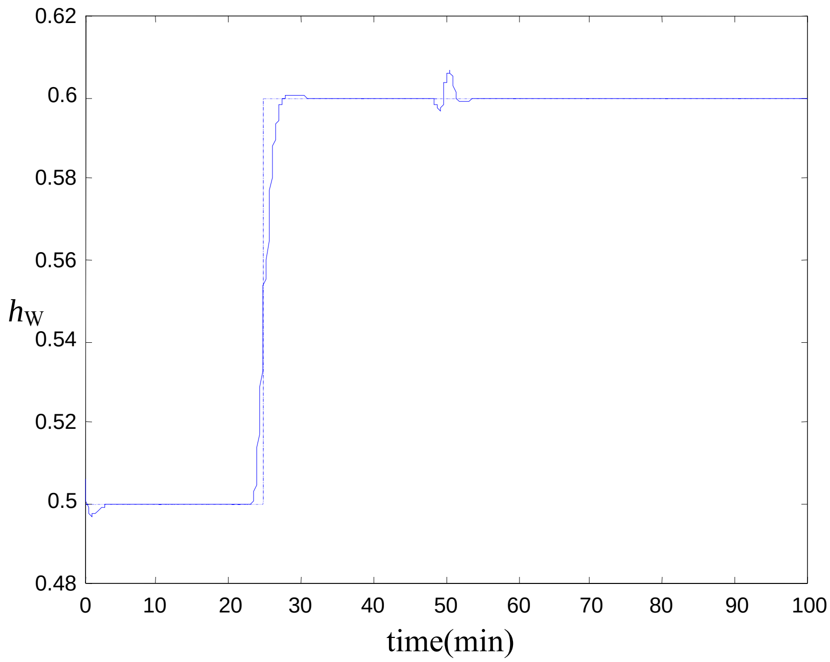

Figure 7.7 Height of Water

All other simulations show that the results for the nonlinear model are very close to that of the linear model. It is important to note that the height of the weir (0.9 meters) is the maximum allowable value for the height of water and that the change in set point was 0.1 (from 0.5 to 0.6 meters). Thus the new set point corresponds to a change in 20% in the height of water, which makes it a good test on how the linear model applies to the separator.

Figures 7.8 through 7.12 show the response of the nonlinear model to the remaining variables.

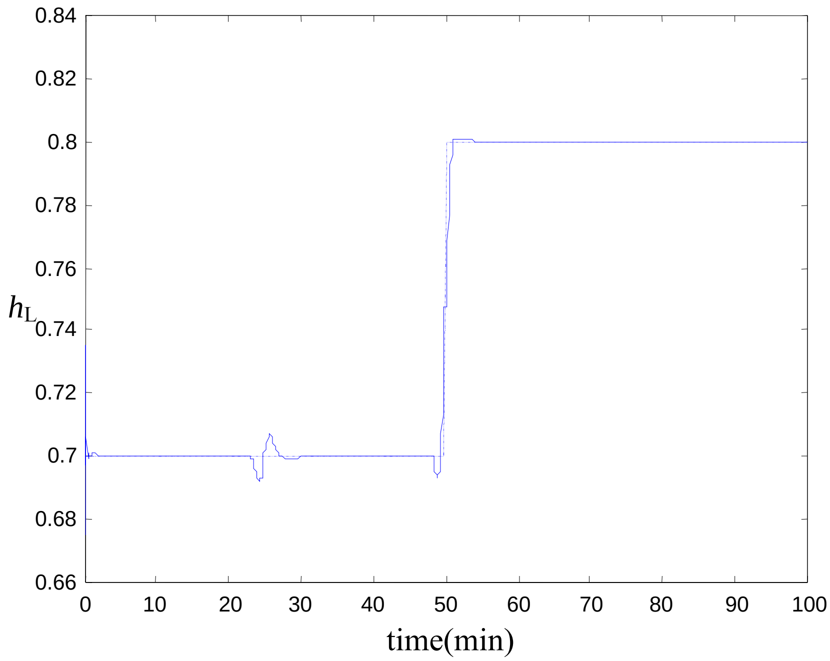

Figure 7.8 Height of Oil

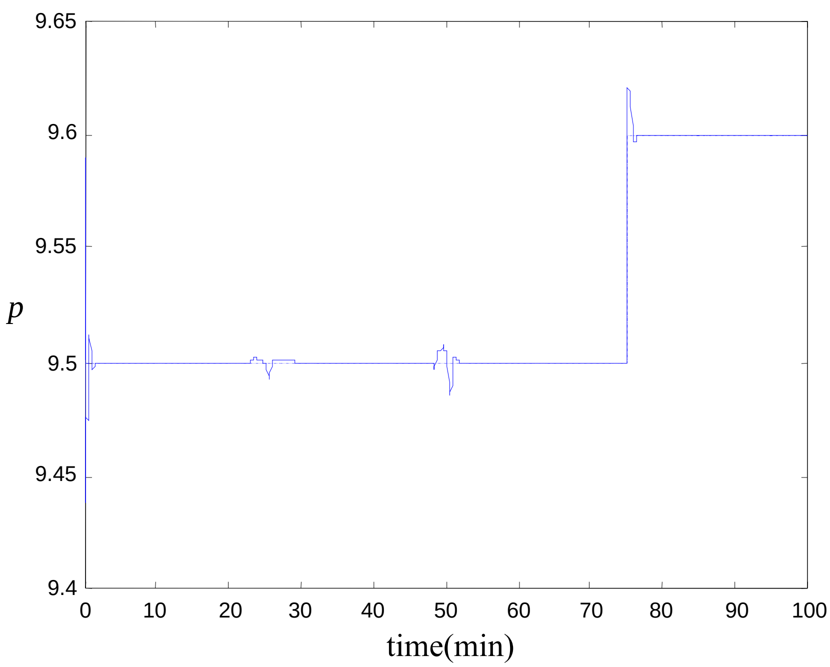

Figure 7.9 Pressure

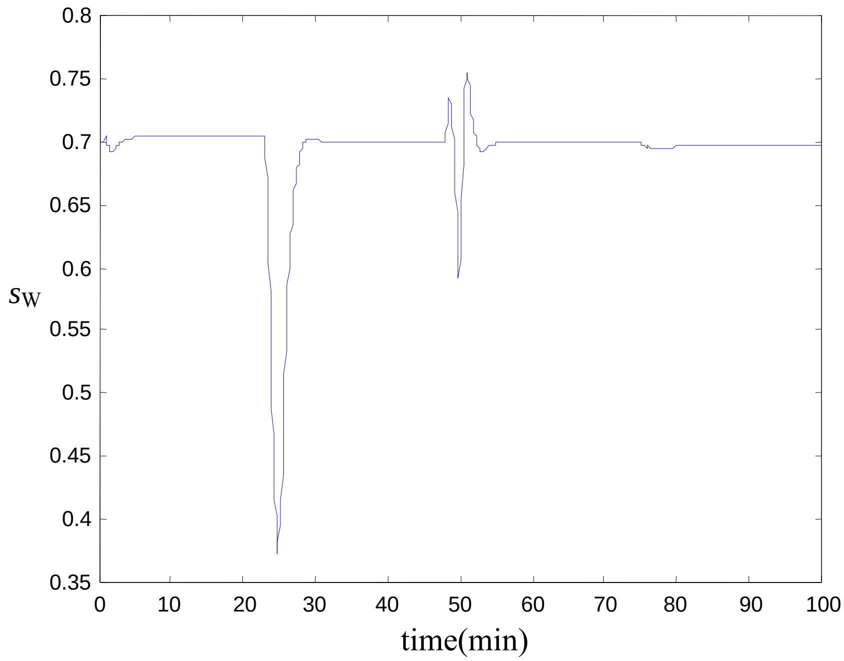

Figure 7.10 Opening of Water Valve

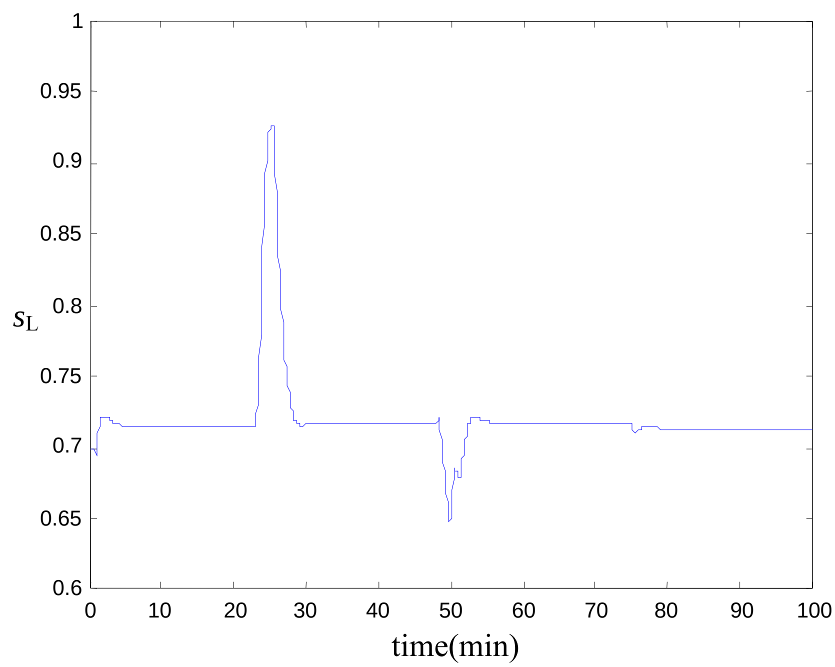

Figure 7.11 Opening of Oil Valve

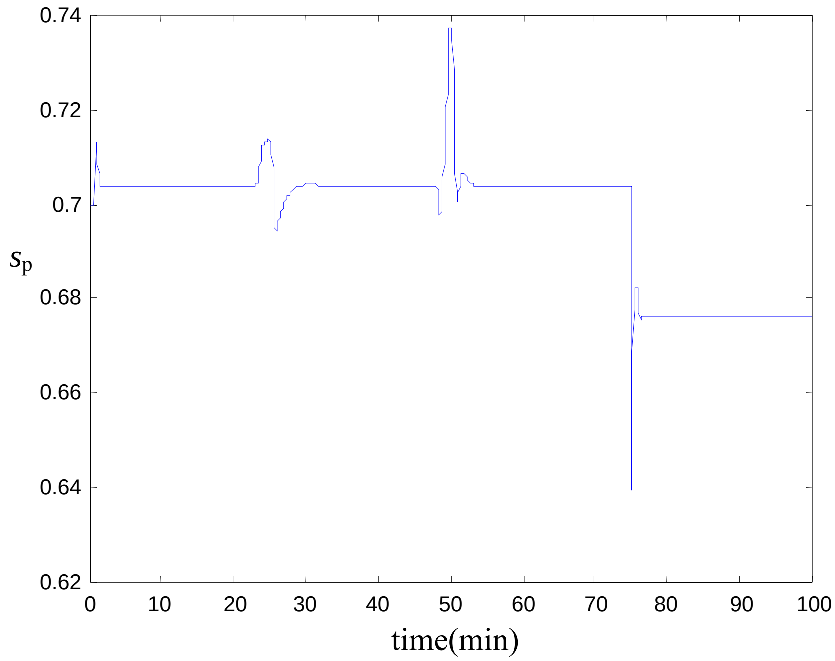

Figure 7.12 Opening of Gas Valve

7.3 Conclusion

The nonlinear simulations gave results close to the linear model of Chapter 6. Thus we conclude that the linearization was successful and the design of the predictive controller for the separator using the techniques of linear control is a good approach.

CHAPTER 8
CONCLUSIONS

This dissertation has successfully developed a method based on matrix fraction descriptions (MFD) for determining the closed-loop poles of unconstrained multivariable predictive control.

The method, proposed here, takes advantage of the polynomial nature of the matrices of the controller to represent the closed-loop transfer matrices as an MFD, a ratio of "numerator" and "denominator" polynomial matrices. It is shown that if a right coprime matrix description of the process transfer matrix is done then the closed-loop poles of the system are determined by evaluating the roots of the determinant of the denominator matrix giving necessary and sufficient conditions for stability. This method avoids the inversion of transfer matrices which is a numerically difficult task. Instead only the determinant of the denominator matrix is required and stability is verified with the help of Jury's criteria by determining if there are any poles that lie outside the unitary circle. Because in this formulation of predictive control a transfer function model was used to represent the process, as opposed to over-parametrized models such as step/impulse response models of DMC like methods, the degree of the determinant is considered minimal. This approach reduces enormously the numerical complexity of the problem allowing the tuning of multivariable systems with many inputs and outputs.

It is proven that the system has zero offset response to step changes in the reference, a property known to be valid for the single input single output case. For systems with equal number of inputs and outputs the “inversion of the plant” is also proven. In this case it is seen that if the weights on the input are zero then the solution to the optimization problem is a controller that inverts the plant and the output matches the reference.

This method becomes an important tool for the analysis of predictive controllers allowing practitioners and researchers to follow the effect of tuning parameters on the behavior of the system.

The use of a multivariable predictive controller for an oil-water-gas separator is studied. The nonlinear model of the plant is linearized around a steady state and a three-inputs three-outputs predictive controller is designed for it. It is seen that the controller responds positively to changes in the parameters and performance objectives can be pursued. Another setup is also studied where a controller with two inputs and two outputs is designed to control the levels of water and oil. The pressure is controlled separately. Results show agreement with the simulations done for the linear model and it is concluded that predictive control is a successful control method for oil-water-gas separators.

APPENDIX A
DETAILED DERIVATION OF CONTROL LAW EQUATION

In this appendix intermediate equations used in the derivation of the control law equation are developed, starting from the model of the plant. Multiple input single output (MISO) formulation is initially used to show the basic structure of the equations. MIMO system equations are derived as a combination of these.

A.1 Derivation of Matrix Diophantine Equation

Consider the Darma model

$$\begin{bmatrix} A_{11} & 0 & \cdots & 0 \\ 0 & A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{mm} \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1p} \\ B_{21} & B_{22} & \cdots & B_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mp} \end{bmatrix} \begin{bmatrix} u_1(t-1) \\ u_2(t-1) \\ \vdots \\ u_p(t-1) \end{bmatrix}$$

Analyze, initially, a MISO system of one output, $y_1(t)$, and p inputs, $u_1(t)$, $u_2(t),\ldots,u_p(t)$. For that the first row of the above matrices can be represented as follows:

$$A_{11}y_1(t) = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1p} \end{bmatrix} \begin{bmatrix} u_1(t-1) \\ u_2(t-1) \\ \vdots \\ u_p(t-1) \end{bmatrix} \tag{A.1}$$

where

$$A_{11}(q^{-1}) = 1 + a_{1_{11}} q^{-1} + a_{2_{11}} q^{-2} + \ldots + a_{nA_{1111}} q^{-nA_{11}}$$

$$B_{11}(q^{-1}) = b_{0_{11}} + b_{1_{11}} q^{-1} + b_{2_{11}} q^{-2} + \ldots + b_{nB_{11_{11}}} q^{-nB_{11}}$$
$$B_{12}(q^{-1}) = b_{0_{12}} + b_{1_{12}} q^{-1} + b_{2_{12}} q^{-2} + \ldots + b_{nB_{12_{12}}} q^{-nB_{12}}$$
$$\vdots$$
$$B_{1p}(q^{-1}) = b_{0_{1p}} + b_{1_{1p}} q^{-1} + b_{2_{1p}} q^{-2} + \ldots + b_{nB_{1p_{1p}}} q^{-nB_{1p}}$$

The first and second indices of polynomials B represent the output and input respectively. Polynomials A have equal indices once matrix $\boldsymbol{A}$, a $m \times m$ diagonal matrix, is related to the outputs only.

Diophantine equation

$$E_{i_1} \Delta A_{11} + q^{-i} F_{i_1} = 1 \tag{A.2}$$

can be used to derive an expression for future values of $y(t)$. Here E_{i_1} and F_{i_1} are polynomials in backward shift operator

$$E_{i_1} = e_{0_1} + e_{1_1} q^{-1} + \ldots + e_{(i-1)_1} q^{-(i-1)} \tag{A.3}$$

$$F_{i_1} = f_{i_{01}} + f_{i_{11}} q^{-1} + \ldots + f_{i_{nA_{11}1}} q^{-nA_{11}} \tag{A.4}$$

and index i shows the time dependence of these equations. Subscript 1 denotes the output (y_1 in this case). Expanding Diophantine equation for time steps from 1 to i then,

$$\begin{gathered} E_{1_1} \Delta A_{11} + q^{-1} F_{1_1} = 1 \\ E_{2_1} \Delta A_{11} + q^{-2} F_{2_1} = 1 \\ \vdots \\ E_{i_1} \Delta A_{11} + q^{-i} F_{i_1} = 1 \end{gathered} \tag{A.5}$$

which can be written in vector-matrix form as

$$\begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \end{bmatrix}_1 A_{11} + \begin{bmatrix} q^{-1} & 0 & \cdots & 0 \\ 0 & q^{-2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^{-i} \end{bmatrix}_1 \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix}_1 = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_1 \tag{A.6}$$

Defining

$$\boldsymbol{E}_{i_1} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \end{bmatrix}_1 \quad \boldsymbol{I}_{i_1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_1 \quad \boldsymbol{q}_1^{-i} = \begin{bmatrix} q^{-1} & 0 & \cdots & 0 \\ 0 & q^{-2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^{-i} \end{bmatrix}_1 \quad \boldsymbol{F}_{i_1} = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix}_1$$

we may write equation (A.6) as:

$$\boldsymbol{E}_{i_1} \Delta A_{11} + \boldsymbol{q}_1^{-i} \boldsymbol{F}_{i_1} = \boldsymbol{I}_{i_1} \tag{A.7}$$

For the other MISO systems we get similar equations:

$$\begin{gathered} \boldsymbol{E}_{i_2} \Delta A_{22} + \boldsymbol{q}_2^{-i} \boldsymbol{F}_{i_2} = \boldsymbol{I}_{i_2} \\ \boldsymbol{E}_{i_3} \Delta A_{33} + \boldsymbol{q}_3^{-i} \boldsymbol{F}_{i_3} = \boldsymbol{I}_{i_3} \\ \vdots \\ \boldsymbol{E}_{i_m} \Delta A_{mm} + \boldsymbol{q}_m^{-i} \boldsymbol{F}_{i_m} = \boldsymbol{I}_{i_m} \end{gathered}$$

and writing in vector-matrix form

$$\begin{bmatrix} \boldsymbol{E}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{E}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{E}_{i_m} \end{bmatrix} \begin{bmatrix} \Delta A_{11} & 0 & \cdots & 0 \\ 0 & \Delta A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \Delta A_{mm} \end{bmatrix} +$$

$$\begin{bmatrix} \boldsymbol{q}_1^{-i} & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}_2^{-i} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}_m^{-i} \end{bmatrix} \begin{bmatrix} \boldsymbol{F}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{I}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{I}_{i_m} \end{bmatrix}$$

Define

$$\begin{gathered} \boldsymbol{y}(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{bmatrix} \quad \boldsymbol{F}_i = \begin{bmatrix} \boldsymbol{F}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m} \end{bmatrix} \quad \boldsymbol{I}_i = \begin{bmatrix} \boldsymbol{I}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{I}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{I}_{i_m} \end{bmatrix} \\ \boldsymbol{q}^{-i} = \begin{bmatrix} \boldsymbol{q}_1^{-i} & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}_2^{-i} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}_m^{-i} \end{bmatrix} \quad \boldsymbol{E}_i = \begin{bmatrix} \boldsymbol{E}_{i_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{E}_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{E}_{i_m} \end{bmatrix} \end{gathered} \tag{A.8}$$

to get the *Matrix Diophantine* equation

$$\boldsymbol{E}_i\boldsymbol{A} + \boldsymbol{q}^{-i}\boldsymbol{F}_i = \boldsymbol{I}_i \tag{A.9}$$

A.2 Derivation of Predictor Equation

If we continue to investigate the first MISO system we see from Diophantine equation, that $E_{i_1}\Delta A_{11}y_1(t) = (1 - q^{-i}F_{i_1})y_1(t)$. Multiply equation (A.1) by $E_{i_1}\Delta$ to get

$$E_{i_1}\Delta A_{11}y_1(t) = E_{i_1}B_{11}\Delta u_1(t-1) + E_{i_1}B_{12}\Delta u_2(t-1) + \ldots + E_{i_1}B_{1p}\Delta u_p(t-1)$$

and

$$(1 - q^{-i}F_{i_1})y_1(t) = E_{i_1}B_{11}\Delta u_1(t-1) + E_{i_1}B_{12}\Delta u_2(t-1) + \ldots + E_{i_1}B_{1p}\Delta u_p(t-1)$$

or

$$y_1(t) = q^{-i}F_{i_1}y_1(t) + E_{i_1}B_{11}\Delta u_1(t-1) + E_{i_1}B_{12}\Delta u_2(t-1) + \ldots + E_{i_1}B_{1p}\Delta u_p(t-1)$$

multiply by q^i to get the expression for the predicted values of y_1

$$\hat{y}_1(t+i) = F_{i_1}y_1(t) + E_{i_1}B_{11}\Delta u_1(t+i-1) + E_{i_1}B_{12}\Delta u_2(t+i-1) + \ldots + E_{i_1}B_{1p}\Delta u_p(t+i-1)$$

and expanding it then

$$\hat{y}_1(t+1) = F_{1_1}y_1(t) + E_{1_1}B_{11}\Delta u_1(t) + E_{1_1}B_{12}\Delta u_2(t) + \ldots + E_{1_1}B_{1p}\Delta u_p(t)$$

$$\hat{y}_1(t+2) = F_{2_1}y_1(t) + E_{2_1}B_{11}\Delta u_1(t+1) + E_{2_1}B_{12}\Delta u_2(t+1) + \ldots + E_{2_1}B_{1p}\Delta u_p(t+1)$$

$$\vdots$$

$$\hat{y}_1(t+i) = F_{i_1}y_1(t) + E_{i_1}B_{11}\Delta u_1(t+i-1) + E_{i_1}B_{12}\Delta u_2(t+i-1) + \ldots + E_{i_1}B_{1p}\Delta u_p(t+i-1) \tag{A.10}$$

the above equations can be written as

$$\begin{bmatrix} \hat{y}_1(t+1) \\ \hat{y}_1(t+2) \\ \vdots \\ \hat{y}_1(t+i) \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix}_1 y_1(t) + \begin{bmatrix} q^1 & 0 & \cdots & 0 \\ 0 & q^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^i \end{bmatrix}_1 \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_i \end{bmatrix}_1 \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1p} \end{bmatrix} \begin{bmatrix} \Delta u_1(t-1) \\ \Delta u_2(t-1) \\ \vdots \\ \Delta u_p(t-1) \end{bmatrix}$$

Define

$$\hat{\boldsymbol{y}}_1(t+i)=\begin{bmatrix}\hat{y}_1(t+1)\\ \hat{y}_1(t+2)\\ \vdots\\ \hat{y}_1(t+i)\end{bmatrix}\quad \boldsymbol{q}_{_1}^i=\begin{bmatrix}q^1 & 0 & \cdots & 0\\ 0 & q^2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & q^i\end{bmatrix}_1\quad \boldsymbol{B}_1=\begin{bmatrix}B_{11} & B_{12} & \cdots & B_{1p}\end{bmatrix}$$

to get

$$\hat{\boldsymbol{y}}_1(t+i)=\boldsymbol{F}_{i_1}y_1(t)+\boldsymbol{q}_{_1}^i\boldsymbol{E}_{i_1}\boldsymbol{B}_1\Delta\boldsymbol{u}(t-1)\tag{A.11}$$

For the other MISO systems we get

$$\hat{\boldsymbol{y}}_2(t+i)=\boldsymbol{F}_{i_2}y_2(t)+\boldsymbol{q}_{_2}^i\boldsymbol{E}_{i_2}\boldsymbol{B}_2\Delta\boldsymbol{u}(t-1)$$

$$\hat{\boldsymbol{y}}_3(t+i)=\boldsymbol{F}_{i_3}y_3(t)+\boldsymbol{q}_{_3}^i\boldsymbol{E}_{i_3}\boldsymbol{B}_3\Delta\boldsymbol{u}(t-1)$$

$$\vdots$$

$$\hat{\boldsymbol{y}}_m(t+i)=\boldsymbol{F}_{i_m}y_m(t)+\boldsymbol{q}_{_m}^i\boldsymbol{E}_{i_m}\boldsymbol{B}_m\Delta\boldsymbol{u}(t-1)$$

which can be written as

$$\begin{bmatrix}\hat{\boldsymbol{y}}_1(t+i)\\ \hat{\boldsymbol{y}}_2(t+i)\\ \vdots\\ \hat{\boldsymbol{y}}_m(t+i)\end{bmatrix}=\begin{bmatrix}\boldsymbol{F}_{i_1} & 0 & \cdots & 0\\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m}\end{bmatrix}\begin{bmatrix}y_1(t)\\ y_2(t)\\ \vdots\\ y_m(t)\end{bmatrix}$$
$$+\begin{bmatrix}\boldsymbol{q}_{_1}^i & 0 & \cdots & 0\\ 0 & \boldsymbol{q}_{_2}^i & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{q}_{_m}^i\end{bmatrix}\begin{bmatrix}\boldsymbol{E}_{i_1} & 0 & \cdots & 0\\ 0 & \boldsymbol{E}_{i_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{E}_{i_m}\end{bmatrix}\begin{bmatrix}\boldsymbol{B}_1\\ \boldsymbol{B}_2\\ \vdots\\ \boldsymbol{B}_m\end{bmatrix}\Delta\boldsymbol{u}(t-1)$$

where once again the subscripts related to the outputs were doubled. If we define

$$\hat{\boldsymbol{y}}(t+i)=\begin{bmatrix}\hat{\boldsymbol{y}}_1(t+i)\\ \hat{\boldsymbol{y}}_2(t+i)\\ \vdots\\ \hat{\boldsymbol{y}}_m(t+i)\end{bmatrix}\quad \boldsymbol{q}^i=\begin{bmatrix}\boldsymbol{q}_{_1}^i & 0 & \cdots & 0\\ 0 & \boldsymbol{q}_{_2}^i & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{q}_{_m}^i\end{bmatrix}$$

we then get the matrix form of the predictor equation

$$\hat{\boldsymbol{y}}(t+i)=\boldsymbol{F}_i\boldsymbol{y}(t)+\boldsymbol{q}^i\boldsymbol{E}_i\boldsymbol{B}\Delta\boldsymbol{u}(t-1)\tag{A.12}$$

A.3 Simplification of Predictor Equation

Back to the scalar equations again, if we recognize that $EB = G + q^{-i}P$ then each element of $\boldsymbol{B}_1$ gives us

$$E_{i_1}B_{11} = G_{i_{11}} + q^{-i}P_{i_{11}},\ E_{i_1}B_{12} = G_{i_{12}} + q^{-i}P_{i_{12}}, \cdots, E_{i_1}B_{1p} = G_{i_{1p}} + q^{-i}P_{i_{1p}}$$

where G and P are polynomials defined as

$$G_{i_{1\ell}} = g_{0_{1\ell}} + g_{1_{1\ell}}q^{-1} + \ldots + g_{1-i_{1\ell}}q^{-(i-1)} \tag{A.13}$$

$$P_{i_{1\ell}} = p_{i_{0_{1\ell}}} + p_{i_{1_{1\ell}}}q^{-1} + \ldots + p_{i_{nB_{1\ell}-1}}q^{-(nB_{1\ell}-1)} \tag{A.14}$$

substituting in equation (A.10) then

$$\begin{aligned}\hat{y}_1(t+i) = F_{i_1}y_1(t) &+ (G_{i_{11}} + q^{-i}P_{i_{11}})\Delta u_1(t+i-1) + (G_{i_{12}} + q^{-i}P_{i_{12}})\Delta u_2(t+i-1) + \ldots + \\ &(G_{i_{1p}} + q^{-i}P_{i_{1p}})\Delta u_p(t+i-1)\end{aligned}$$

and expanding in time we get

$$\begin{aligned}\hat{y}_1(t+1) = F_{1_1}y_1(t) &+ (G_{1_{11}} + q^{-1}P_{1_{11}})\Delta u_1(t) + (G_{1_{12}} + q^{-1}P_{1_{12}})\Delta u_2(t) + \ldots + \\ &(G_{1_{1p}} + q^{-1}P_{1_{1p}})\Delta u_p(t)\end{aligned}$$

$$\begin{aligned}\hat{y}_1(t+2) = F_{2_1}y_1(t) &+ (G_{2_{11}} + q^{-2}P_{2_{11}})\Delta u_1(t+1) + (G_{2_{12}} + q^{-2}P_{2_{12}})\Delta u_2(t+1) + \ldots + \\ &(G_{2_{1p}} + q^{-2}P_{2_{1p}})\Delta u_p(t+1)\end{aligned}$$

$$\vdots$$

$$\begin{aligned}\hat{y}_1(t+i) = F_{i_1}y_1(t) &+ (G_{i_{11}} + q^{-i}P_{i_{11}})\Delta u_1(t+i-1) + (G_{i_{12}} + q^{-i}P_{i_{12}})\Delta u_2(t+i-1) + \ldots + \\ &(G_{i_{1p}} + q^{-i}P_{i_{1p}})\Delta u_p(t+i-1)\end{aligned}$$

and rearranging common terms we get

$$\begin{aligned}\hat{y}_1(t+1) = F_{1_1}y_1(t) &+ G_{1_{11}}\Delta u_1(t) + G_{1_{12}}\Delta u_2(t) + \ldots + G_{1_{1p}}\Delta u_p(t) \\ &+ P_{1_{11}}\Delta u_1(t-1) + P_{1_{12}}\Delta u_2(t-1) + \ldots + P_{1_{1p}}\Delta u_p(t-1)\end{aligned}$$

$$\begin{aligned}\hat{y}_1(t+2) = F_{2_1}y_1(t) &+ G_{2_{11}}\Delta u_1(t+1) + G_{2_{12}}\Delta u_2(t+1) \ldots + G_{2_{1p}}\Delta u_p(t+1) \\ &+ P_{2_{11}}\Delta u_1(t-1) + P_{2_{12}}\Delta u_2(t-1) + \ldots + P_{2_{1p}}\Delta u_p(t-1)\end{aligned}$$

$$\vdots$$

$$\hat{y}_1(t+i) = F_{i_1} y_1(t) + G_{i_{11}} \Delta u_1(t+i-1) + G_{i_{12}} \Delta u_2(t+i-1) + \ldots + G_{i_{1p}} \Delta u_p(t+i-1)$$
$$+ P_{i_{11}} \Delta u_1(t-1) + P_{i_{12}} \Delta u_2(t-1) + \ldots + P_{i_{1p}} \Delta u_p(t-1)$$

which can be written as

$$\begin{bmatrix} \hat{y}_1(t+1) \\ \hat{y}_1(t+2) \\ \vdots \\ \hat{y}_1(t+i) \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix}_1 y_1(t) + \begin{bmatrix} q^1 & 0 & \cdots & 0 \\ 0 & q^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^i \end{bmatrix}_1 \begin{bmatrix} G_{1_{11}} & G_{1_{12}} & \cdots & G_{1_{1p}} \\ G_{2_{11}} & G_{2_{12}} & \cdots & G_{2_{1p}} \\ \vdots & \vdots & \ddots & \vdots \\ G_{i_{11}} & G_{i_{12}} & \cdots & G_{i_{1p}} \end{bmatrix} \begin{bmatrix} \Delta u_1(t-1) \\ \Delta u_2(t-1) \\ \vdots \\ \Delta u_p(t-1) \end{bmatrix} +$$
$$\begin{bmatrix} P_{1_{11}} & P_{1_{12}} & \cdots & P_{1_{1p}} \\ P_{2_{11}} & P_{2_{12}} & \cdots & P_{2_{1p}} \\ \vdots & \vdots & \ddots & \vdots \\ P_{i_{11}} & P_{i_{12}} & \cdots & P_{i_{1p}} \end{bmatrix} \begin{bmatrix} \Delta u_1(t-1) \\ \Delta u_2(t-1) \\ \vdots \\ \Delta u_p(t-1) \end{bmatrix}$$

Defining:

$$\boldsymbol{G}_{i_{11}} = \begin{bmatrix} G_{1_{11}} \\ G_{2_{11}} \\ \vdots \\ G_{Ny_{1_{11}}} \end{bmatrix} \quad \boldsymbol{G}_{i_{12}} = \begin{bmatrix} G_{1_{12}} \\ G_{2_{12}} \\ \vdots \\ G_{Ny_{1_{12}}} \end{bmatrix} \quad \cdots \quad \boldsymbol{G}_{i_{1p}} = \begin{bmatrix} G_{1_{1p}} \\ G_{2_{1p}} \\ \vdots \\ G_{Ny_{1_{1p}}} \end{bmatrix}$$

$$\boldsymbol{P}_{i_{11}} = \begin{bmatrix} P_{1_{11}} \\ P_{2_{11}} \\ \vdots \\ P_{Ny_{1_{11}}} \end{bmatrix} \quad \boldsymbol{P}_{i_{12}} = \begin{bmatrix} P_{1_{12}} \\ P_{2_{12}} \\ \vdots \\ P_{Ny_{1_{12}}} \end{bmatrix} \quad \cdots \quad \boldsymbol{P}_{i_{1p}} = \begin{bmatrix} P_{1_{1p}} \\ P_{2_{1p}} \\ \vdots \\ P_{Ny_{1_{1p}}} \end{bmatrix}$$

then

$$\hat{\boldsymbol{y}}_1(t+i) = \boldsymbol{F}_{i_1} y_1(t) + \left[\boldsymbol{q}_1^i \left[\boldsymbol{G}_{i_{11}} \quad \boldsymbol{G}_{i_{12}} \quad \ldots \quad \boldsymbol{G}_{i_{1p}} \right] + \left[\boldsymbol{P}_{i_{11}} \quad \boldsymbol{P}_{i_{12}} \quad \ldots \quad \boldsymbol{P}_{i_{1p}} \right] \right] \begin{bmatrix} \Delta u_1(t-1) \\ \Delta u_2(t-1) \\ \vdots \\ \Delta u_p(t-1) \end{bmatrix}$$

For the other MISO systems we get

$$\hat{\boldsymbol{y}}_2(t+i)=\boldsymbol{F}_{i_2}y_2(t)+\left[\boldsymbol{q}_2^{\ i}\left[\boldsymbol{G}_{i_{21}}\quad \boldsymbol{G}_{i_{22}}\quad \ldots\quad \boldsymbol{G}_{i_{2p}}\right]+\left[\boldsymbol{P}_{i_{21}}\quad \boldsymbol{P}_{i_{22}}\quad \ldots\quad \boldsymbol{P}_{i_{2p}}\right]\right]\begin{bmatrix}\Delta u_1(t-1)\\ \Delta u_2(t-1)\\ \vdots \\ \Delta u_p(t-1)\end{bmatrix}$$

$$\hat{\boldsymbol{y}}_3(t+i)=\boldsymbol{F}_{i_3}y_3(t)+\left[\boldsymbol{q}_3^{\ i}\left[\boldsymbol{G}_{i_{31}}\quad \boldsymbol{G}_{i_{32}}\quad \ldots\quad \boldsymbol{G}_{i_{3p}}\right]+\left[\boldsymbol{P}_{i_{31}}\quad \boldsymbol{P}_{i_{32}}\quad \ldots\quad \boldsymbol{P}_{i_{3p}}\right]\right]\begin{bmatrix}\Delta u_1(t-1)\\ \Delta u_2(t-1)\\ \vdots \\ \Delta u_p(t-1)\end{bmatrix}$$

$$\vdots$$

$$\hat{\boldsymbol{y}}_m(t+i)=\boldsymbol{F}_{i_m}y_m(t)+\left[\boldsymbol{q}_m^{\ i}\left[\boldsymbol{G}_{i_{m1}}\quad \boldsymbol{G}_{i_{m2}}\quad \ldots\quad \boldsymbol{G}_{i_{mp}}\right]+\left[\boldsymbol{P}_{i_{m1}}\quad \boldsymbol{P}_{i_{m2}}\quad \ldots\quad \boldsymbol{P}_{i_{mp}}\right]\right]\begin{bmatrix}\Delta u_1(t-1)\\ \Delta u_2(t-1)\\ \vdots \\ \Delta u_p(t-1)\end{bmatrix}$$

which can be written as

$$\begin{bmatrix}\hat{\boldsymbol{y}}_1(t+i)\\ \hat{\boldsymbol{y}}_2(t+i)\\ \vdots \\ \hat{\boldsymbol{y}}_m(t+i)\end{bmatrix}=\begin{bmatrix}\boldsymbol{F}_{i_1} & 0 & \cdots & 0\\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m}\end{bmatrix}\begin{bmatrix}y_1(t)\\ y_2(t)\\ \vdots \\ y_m(t)\end{bmatrix}+$$

$$\begin{bmatrix}\boldsymbol{q}_1^{\ i} & 0 & \cdots & 0\\ 0 & \boldsymbol{q}_2^{\ i} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}_m^{\ i}\end{bmatrix}\begin{bmatrix}\boldsymbol{G}_{i_{11}} & \boldsymbol{G}_{i_{12}} & \cdots & \boldsymbol{G}_{i_{1p}}\\ \boldsymbol{G}_{i_{21}} & \boldsymbol{G}_{i_{22}} & \cdots & \boldsymbol{G}_{i_{2p}}\\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{G}_{i_{m1}} & \boldsymbol{G}_{i_{m2}} & \cdots & \boldsymbol{G}_{i_{mp}}\end{bmatrix}\begin{bmatrix}\Delta \boldsymbol{u}_1(t-1)\\ \Delta \boldsymbol{u}_2(t-1)\\ \vdots \\ \Delta \boldsymbol{u}_p(t-1)\end{bmatrix}+$$

$$\begin{bmatrix}\boldsymbol{P}_{i_{11}} & \boldsymbol{P}_{i_{12}} & \cdots & \boldsymbol{P}_{i_{1p}}\\ \boldsymbol{P}_{i_{21}} & \boldsymbol{P}_{i_{22}} & \cdots & \boldsymbol{P}_{i_{2p}}\\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{i_{m1}} & \boldsymbol{P}_{i_{m2}} & \cdots & \boldsymbol{P}_{i_{mp}}\end{bmatrix}\begin{bmatrix}\Delta u_1(t-1)\\ \Delta u_2(t-1)\\ \vdots \\ \Delta u_p(t-1)\end{bmatrix}$$

Finally

$$\hat{\boldsymbol{y}}(t+i)=\boldsymbol{F}_i\boldsymbol{y}(t)+\boldsymbol{q}^i\boldsymbol{G}_i\Delta\boldsymbol{u}(t-1)+\boldsymbol{P}_i\Delta\boldsymbol{u}(t-1) \qquad \text{(A.15)}$$

Note that the elements of each matrix $\boldsymbol{G}_{i_{j\ell}}$ and $\boldsymbol{P}_{i_{j\ell}}$ are defined as:

$$G_{i_{j\ell}}=g_{0_{j\ell}}+g_{1_{j\ell}}q^{-1}+\ldots+g_{(i-1)_{j\ell}}q^{-(i-1)}$$

$$P_{i_{j\ell}} = p_{i_{0_{j\ell}}} + p_{i_{1_{j\ell}}} q^{-1} + \ldots + p_{i_{nB_{j\ell}-1}} q^{-(nB_{j\ell}-1)}$$

A.4 Derivation of Constant Forcing Response Equation

Back to the scalar equations, if we recognize that $\hat{y}(t+i) = y^0(t+i)$ when the inputs are kept constant

$$\begin{aligned} \Delta u_1(t+i-1) &= 0 \\ \Delta u_2(t+i-1) &= 0 \\ &\vdots \\ \Delta u_p(t+i-1) &= 0 \end{aligned} \tag{A.16}$$

then equation (A.15) yields

$$y_1^0(t+i) = F_{i_1} y_1(t) + P_{i_{11}} \Delta u_1(t-1) + P_{i_{12}} \Delta u_2(t-1) + \ldots + P_{i_{1p}} \Delta u_p(t-1) \tag{A.17}$$

For the matrix equations we get the same result if $\boldsymbol{q}^i \boldsymbol{G}_i \Delta \boldsymbol{u}(t-1) = 0$. This way equation (A.15) yields

$$\boldsymbol{y}^0(t+i) = \boldsymbol{F}_i \boldsymbol{y}(t) + q^{-1} \boldsymbol{P}_i \Delta \boldsymbol{u}(t) \tag{A.18}$$

A.5 Substitution of Constant Forcing Response in the Control Law Equation

Back again to the MISO case, if we substitute the expression of $y_1^0(t+i)$, in equation (A.17), in the control law equation, (3.11)

$$\begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} = \begin{bmatrix} [k_1 \quad k_2 \quad \cdots \quad k_i]_{11} \\ [k_1 \quad k_2 \quad \cdots \quad k_i]_{21} \\ \vdots \\ [k_1 \quad k_2 \quad \cdots \quad k_i]_{p1} \end{bmatrix} \begin{bmatrix} r_1(t+1) - y_1^0(t+1) \\ r_1(t+2) - y_1^0(t+2) \\ \vdots \\ r_1(t+i) - y_1^0(t+i) \end{bmatrix}$$

then

$$
\begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{11} \\ \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{21} \\ \vdots \\ \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{p1} \end{bmatrix}
\begin{bmatrix} r_1(t+1) - F_{1_1} y_1(t) + q^{-1} P_{1_{11}} \Delta u_1(t) + q^{-1} P_{1_{12}} \Delta u_2(t) + \ldots + q^{-1} P_{1_{1p}} \Delta u_p(t) \\ r_1(t+2) - F_{2_1} y_1(t) + q^{-1} P_{2_{11}} \Delta u_1(t) + q^{-1} P_{2_{12}} \Delta u_2(t) + \ldots + q^{-1} P_{2_{1p}} \Delta u_p(t) \\ \vdots \\ r_1(t+i) - F_{i_1} y_1(t) + q^{-1} P_{i_{11}} \Delta u_1(t) + q^{-1} P_{i_{12}} \Delta u_2(t) + \ldots + q^{-1} P_{i_{1p}} \Delta u_p(t) \end{bmatrix}
$$

which can be written as

$$
\begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{11} \\ \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{21} \\ \vdots \\ \begin{bmatrix} k_1 & k_2 & \cdots & k_i \end{bmatrix}_{p1} \end{bmatrix}
\left[\begin{bmatrix} r_1(t+1) \\ r_1(t+2) \\ \vdots \\ r_1(t+i) \end{bmatrix} - \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_i \end{bmatrix}_1 y_1(t) + q^{-1} \begin{bmatrix} P_{1_{11}} & P_{1_{12}} & \cdots & P_{1_{1p}} \\ P_{2_{11}} & P_{2_{12}} & \cdots & P_{2_{1p}} \\ \vdots & \vdots & \ddots & \vdots \\ P_{i_{11}} & P_{i_{12}} & \cdots & P_{i_{1p}} \end{bmatrix} \begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} \right]
$$

or

$$
\begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{k}_{11}^T \\ \boldsymbol{k}_{21}^T \\ \vdots \\ \boldsymbol{k}_{p1}^T \end{bmatrix}
\left[\boldsymbol{r}_1(t+i) - \boldsymbol{F}_{i_1} y_1(t) - q^{-1} \begin{bmatrix} \boldsymbol{P}_{i_{11}} & \boldsymbol{P}_{i_{12}} & \ldots & \boldsymbol{P}_{i_{1p}} \end{bmatrix} \begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} \right]
$$

Thus for a MIMO system we get:

$$\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}=\begin{bmatrix}\begin{bmatrix}\boldsymbol{k}_{11}^T\\ \boldsymbol{k}_{21}^T\\ \vdots\\ \boldsymbol{k}_{p1}^T\end{bmatrix}\left[\boldsymbol{r}_1(t+i)-\boldsymbol{F}_{i_1}y_1(t)-q^{-1}\begin{bmatrix}\boldsymbol{P}_{i_{11}} & \boldsymbol{P}_{i_{12}} & \dots & \boldsymbol{P}_{i_{1p}}\end{bmatrix}\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}\right]\\ +\begin{bmatrix}\boldsymbol{k}_{12}^T\\ \boldsymbol{k}_{22}^T\\ \vdots\\ \boldsymbol{k}_{p2}^T\end{bmatrix}\left[\boldsymbol{r}_2(t+i)-\boldsymbol{F}_{i_2}y_2(t)-q^{-1}\begin{bmatrix}\boldsymbol{P}_{i_{21}} & \boldsymbol{P}_{i_{22}} & \dots & \boldsymbol{P}_{i_{2p}}\end{bmatrix}\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}\right]\\ \vdots\\ +\begin{bmatrix}\boldsymbol{k}_{1m}^T\\ \boldsymbol{k}_{2m}^T\\ \vdots\\ \boldsymbol{k}_{pm}^T\end{bmatrix}\left[\boldsymbol{r}_m(t+i)-\boldsymbol{F}_{i_m}y_m(t)-q^{-1}\begin{bmatrix}\boldsymbol{P}_{i_{m1}} & \boldsymbol{P}_{i_{m2}} & \dots & \boldsymbol{P}_{i_{mp}}\end{bmatrix}\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}\right]\end{bmatrix}$$

which can be expressed as

$$\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}=\begin{bmatrix}\boldsymbol{k}_{11}^T & \boldsymbol{k}_{12}^T & \cdots & \boldsymbol{k}_{1m}^T\\ \boldsymbol{k}_{21}^T & \boldsymbol{k}_{22}^T & \cdots & \boldsymbol{k}_{2m}^T\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{k}_{p1}^T & \boldsymbol{k}_{p2}^T & \cdots & \boldsymbol{k}_{pm}^T\end{bmatrix}$$

$$\left[\begin{bmatrix}\boldsymbol{r}_1(t+i)\\ \boldsymbol{r}_2(t+i)\\ \vdots\\ \boldsymbol{r}_m(t+i)\end{bmatrix}-\begin{bmatrix}\boldsymbol{F}_{i_1} & 0 & \cdots & 0\\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m}\end{bmatrix}\begin{bmatrix}y_1(t)\\ y_2(t)\\ \vdots\\ y_m(t)\end{bmatrix}+q^{-1}\begin{bmatrix}\boldsymbol{P}_{i_{11}} & \boldsymbol{P}_{i_{12}} & \cdots & \boldsymbol{P}_{i_{1p}}\\ \boldsymbol{P}_{i_{21}} & \boldsymbol{P}_{i_{22}} & \cdots & \boldsymbol{P}_{i_{2p}}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{P}_{i_{m1}} & \boldsymbol{P}_{i_{m2}} & \cdots & \boldsymbol{P}_{i_{mp}}\end{bmatrix}\begin{bmatrix}\Delta u_1(t)\\ \Delta u_2(t)\\ \vdots\\ \Delta u_p(t)\end{bmatrix}\right]$$

defining

$$\boldsymbol{F}_i=\begin{bmatrix}\boldsymbol{F}_{i_1} & 0 & \cdots & 0\\ 0 & \boldsymbol{F}_{i_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \boldsymbol{F}_{i_m}\end{bmatrix}\qquad \boldsymbol{P}_i=\begin{bmatrix}\boldsymbol{P}_{i_{11}} & \boldsymbol{P}_{i_{12}} & \cdots & \boldsymbol{P}_{i_{1p}}\\ \boldsymbol{P}_{i_{21}} & \boldsymbol{P}_{i_{22}} & \cdots & \boldsymbol{P}_{i_{2p}}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{P}_{i_{m1}} & \boldsymbol{P}_{i_{m2}} & \cdots & \boldsymbol{P}_{i_{mp}}\end{bmatrix}$$

we finally get

$$\Delta\boldsymbol{u}(t)=\boldsymbol{K}_i\left[\boldsymbol{r}(t+i)-\boldsymbol{F}_i\boldsymbol{y}(t)-q^{-1}\boldsymbol{P}_i\Delta\boldsymbol{u}(t)\right] \qquad \text{(A.19)}$$

or yet

$$(\boldsymbol{I}_i + q^{-1}\boldsymbol{K}_i\boldsymbol{P}_i)\Delta\boldsymbol{u}(t) = \boldsymbol{K}_i\boldsymbol{q}^i\boldsymbol{r}(t) - \boldsymbol{K}_i\boldsymbol{F}_i\boldsymbol{y}(t) \tag{A.20}$$

Define

$$\boldsymbol{R}_i = (\boldsymbol{I}_i + q^{-1}\boldsymbol{K}_i\boldsymbol{P}_i)\Delta \tag{A.21}$$

$$\boldsymbol{S}_i = \boldsymbol{K}_i\boldsymbol{F}_i \tag{A.22}$$

$$\boldsymbol{T}_i = \boldsymbol{K}_i\boldsymbol{q}^i \tag{A.23}$$

The final control law equation is

$$\boldsymbol{R}_i\boldsymbol{u}(t) = \boldsymbol{T}_i\boldsymbol{r}(t) - \boldsymbol{S}_i\boldsymbol{y}(t)$$

If the whole prediction horizon is considered then

$$\boldsymbol{R} = (\boldsymbol{I} + q^{-1}\boldsymbol{K}\boldsymbol{P})\Delta \tag{A.24}$$

$$\boldsymbol{S} = \boldsymbol{K}\boldsymbol{F} \tag{A.25}$$

$$\boldsymbol{T} = \boldsymbol{K}\boldsymbol{q}^{Ny} \tag{A.26}$$

where

$$\boldsymbol{q}^{Ny} = \begin{bmatrix} \boldsymbol{q}_1^{Ny_1} & 0 & \cdots & 0 \\ 0 & \boldsymbol{q}_2^{Ny_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{q}_m^{Ny_m} \end{bmatrix} \text{ and } \boldsymbol{q}_j^{Ny_j} = \begin{bmatrix} q^1 & 0 & \cdots & 0 \\ 0 & q^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q^{Ny_j} \end{bmatrix}_j$$

then the final control law is

$$\boldsymbol{R}\boldsymbol{u}(t) = \boldsymbol{T}\boldsymbol{r}(t) - \boldsymbol{S}\boldsymbol{y}(t)$$

APPENDIX B
DERIVATION OF PRODUCT *KP* AND *KF*

B.1 Product of *KP*

Consider the definitions of $\boldsymbol{K}$ and $\boldsymbol{P}$ as in Appendix A.

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{k}_{11}^T & \boldsymbol{k}_{12}^T & \cdots & \boldsymbol{k}_{1m}^T \\ \boldsymbol{k}_{21}^T & \boldsymbol{k}_{22}^T & \cdots & \boldsymbol{k}_{2m}^T \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{k}_{p1}^T & \boldsymbol{k}_{p2}^T & \cdots & \boldsymbol{k}_{pm}^T \end{bmatrix} \qquad \boldsymbol{P} = \begin{bmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} & \cdots & \boldsymbol{P}_{1p} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} & \cdots & \boldsymbol{P}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{P}_{m1} & \boldsymbol{P}_{m2} & \cdots & \boldsymbol{P}_{mp} \end{bmatrix}$$

where $\boldsymbol{k}_{\ell j}^T = [\, k_1 \quad k_2 \quad \cdots \quad k_{Ny_j} \,]_{\ell j}$ are the elements of matrix $\boldsymbol{K}$. The elements of matrix $\boldsymbol{P}$ are column vectors

$$\boldsymbol{P}_{j\ell} = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{Ny_j} \end{bmatrix}_{j\ell}$$

of polynomials in backward shift operator defined as $P_{i_{j\ell}} = p_{i_{0_{j\ell}}} + p_{i_{1_{j\ell}}} q^{-1} + \ldots + p_{i_{nB_{j\ell}-1}} q^{-(nB_{j\ell}-1)}$. The product $\boldsymbol{KP}$ is

$$\begin{bmatrix} \boldsymbol{k}_{11}^T\boldsymbol{P}_{11} + \boldsymbol{k}_{12}^T\boldsymbol{P}_{21} + \ldots + \boldsymbol{k}_{1m}^T\boldsymbol{P}_{m1} & \boldsymbol{k}_{11}^T\boldsymbol{P}_{12} + \boldsymbol{k}_{12}^T\boldsymbol{P}_{22} + \ldots + \boldsymbol{k}_{1m}^T\boldsymbol{P}_{m2} & \cdots & \boldsymbol{k}_{11}^T\boldsymbol{P}_{1p} + \boldsymbol{k}_{12}^T\boldsymbol{P}_{2p} + \ldots + \boldsymbol{k}_{1m}^T\boldsymbol{P}_{mp} \\ \boldsymbol{k}_{21}^T\boldsymbol{P}_{11} + \boldsymbol{k}_{22}^T\boldsymbol{P}_{21} + \ldots + \boldsymbol{k}_{2m}^T\boldsymbol{P}_{m1} & \boldsymbol{k}_{21}^T\boldsymbol{P}_{12} + \boldsymbol{k}_{22}^T\boldsymbol{P}_{22} + \ldots + \boldsymbol{k}_{2m}^T\boldsymbol{P}_{m2} & \cdots & \boldsymbol{k}_{21}^T\boldsymbol{P}_{1p} + \boldsymbol{k}_{22}^T\boldsymbol{P}_{2p} + \ldots + \boldsymbol{k}_{2m}^T\boldsymbol{P}_{mp} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{k}_{m1}^T\boldsymbol{P}_{11} + \boldsymbol{k}_{m2}^T\boldsymbol{P}_{21} + \ldots + \boldsymbol{k}_{pm}^T\boldsymbol{P}_{m1} & \boldsymbol{k}_{p1}^T\boldsymbol{P}_{12} + \boldsymbol{k}_{p2}^T\boldsymbol{P}_{22} + \ldots + \boldsymbol{k}_{pm}^T\boldsymbol{P}_{m2} & \cdots & \boldsymbol{k}_{p1}^T\boldsymbol{P}_{1p} + \boldsymbol{k}_{p2}^T\boldsymbol{P}_{2p} + \ldots + \boldsymbol{k}_{pm}^T\boldsymbol{P}_{mp} \end{bmatrix}$$

Note that each product $\boldsymbol{k}_{\ell j}^T \boldsymbol{P}_{j\ell}$ is

$$\boldsymbol{k}_{\ell j}^T \boldsymbol{P}_{j\ell} = [\,k_1 \quad k_2 \quad \cdots \quad k_{Ny_j}\,]_{\ell j} \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{Ny_j} \end{bmatrix}_{j\ell}$$

The structure of $\boldsymbol{K}$, in this case, is such that its elements are vectors with a first non-zero term followed by zeros, i.e. $\boldsymbol{k}_{\ell j}^T = \begin{bmatrix} k_1 & 0 & \cdots & 0 \end{bmatrix}_{\ell j}$, which, except for the sequence of zeros, is identical to $\begin{bmatrix} k_1 \end{bmatrix}_{\ell j}$, i.e. $\boldsymbol{k}_{1_{\ell j}}^T \equiv \begin{bmatrix} k_1 \end{bmatrix}_{\ell j}$. Then except for the first polynomial $P_{1_{j\ell}}$, where $i = 1$, the remaining polynomials, $P_{2_{j\ell}} \ldots P_{Ny_{j_{j\ell}}}$, are cancelled by the zeros of $\boldsymbol{k}_{\ell j}^T$ during the process of multiplication.

$$\boldsymbol{k}_{\ell j}^T \boldsymbol{P}_{j\ell} = [\,k_1 \quad 0 \quad \cdots \quad 0\,]_{\ell j} \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{Ny_j} \end{bmatrix}_{j\ell} = k_{1_{\ell j}} P_{1_{j\ell}} \equiv [k_1]_{\ell j} \begin{bmatrix} P_1 \end{bmatrix}_{j\ell} = \boldsymbol{k}_{1_{\ell j}}^T \boldsymbol{P}_{1_{j\ell}}$$

That way the product of $\boldsymbol{KP}$ is

$$\begin{bmatrix} k_{1_{11}} P_{1_{11}} + k_{1_{12}} P_{1_{21}} + \ldots + k_{1_{1m}} P_{1_{m1}} & k_{1_{11}} P_{1_{12}} + k_{1_{12}} P_{1_{22}} + \ldots + k_{1_{1m}} P_{1_{m2}} & \cdots & k_{1_{11}} P_{1_{1p}} + k_{1_{12}} P_{1_{2p}} + \ldots + k_{1_{1m}} P_{1_{mp}} \\ k_{1_{21}} P_{1_{11}} + k_{1_{22}} P_{1_{21}} + \ldots + k_{1_{2m}} P_{1_{m1}} & k_{1_{21}} P_{1_{12}} + k_{1_{22}} P_{1_{22}} + \ldots + k_{1_{2m}} P_{1_{m2}} & \cdots & k_{1_{21}} P_{1_{1p}} + k_{1_{22}} P_{1_{2p}} + \ldots + k_{1_{2m}} P_{1_{mp}} \\ \vdots & \vdots & \ddots & \vdots \\ k_{1_{m1}} P_{1_{11}} + k_{1_{m2}} P_{1_{21}} + \ldots + k_{1_{pm}} P_{1_{m1}} & k_{1_{p1}} P_{1_{12}} + k_{1_{p2}} P_{1_{22}} + \ldots + k_{1_{pm}} P_{1_{m2}} & \cdots & k_{1_{p1}} P_{1_{1p}} + k_{1_{p2}} P_{1_{2p}} + \ldots + k_{1_{pm}} P_{1_{mp}} \end{bmatrix}$$

which is identical to $\boldsymbol{K}_1\boldsymbol{P}_1$. Thus it is concluded that under the given conditions $\boldsymbol{KP} \equiv \boldsymbol{K}_1\boldsymbol{P}_1$.

B.2 Product of $\boldsymbol{KF}$

Consider the definition of $\boldsymbol{F}$ as in (A.8):

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1 & 0 & \cdots & 0 \\ 0 & \boldsymbol{F}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{F}_m \end{bmatrix}$$

where $\boldsymbol{F}_j = \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_{Ny_j} \end{bmatrix}_j$ and each $F_{i_j} = f_{i_0 j} + f_{i_1 j} q^{-1} + \ldots + f_{i_{nA_{jj}} j} q^{-nA_{jj}}$ is a polynomial in backward shift operator. Similarly to the previous case the product $\boldsymbol{KF}$ is

$$\boldsymbol{KF} = \begin{bmatrix} \boldsymbol{k}_{11}^T \boldsymbol{F}_1 & \boldsymbol{k}_{12}^T \boldsymbol{F}_2 & \cdots & \boldsymbol{k}_{1m}^T \boldsymbol{F}_m \\ \boldsymbol{k}_{21}^T \boldsymbol{F}_1 & \boldsymbol{k}_{22}^T \boldsymbol{F}_2 & \cdots & \boldsymbol{k}_{2m}^T \boldsymbol{F}_m \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{k}_{p1}^T \boldsymbol{F}_1 & \boldsymbol{k}_{p2}^T \boldsymbol{F}_2 & \cdots & \boldsymbol{k}_{pm}^T \boldsymbol{F}_m \end{bmatrix}$$

and for the specific format of $\boldsymbol{K}$ then

$$\boldsymbol{KF} = \begin{bmatrix} k_{1_{11}} F_{1_1} & k_{1_{12}} F_{1_2} & \cdots & k_{1_{1m}} F_{1_m} \\ k_{1_{21}} F_{1_1} & k_{1_{22}} F_{1_2} & \cdots & k_{1_{2m}} F_{1_m} \\ \vdots & \vdots & \ddots & \vdots \\ k_{1_{m1}} F_{1_1} & k_{1_{p2}} F_{1_2} & \cdots & k_{1_{pm}} F_{1_m} \end{bmatrix} \equiv \boldsymbol{K}_1 \boldsymbol{F}_1$$

from which we conclude that $\boldsymbol{KF} \equiv \boldsymbol{K}_1 \boldsymbol{F}_1$.

APPENDIX C
OBTAINING A LEFT MATRIX DESCRIPTION

Given a transfer function model $\boldsymbol{y}(t)=\boldsymbol{H}(z^{-1})\boldsymbol{u}(t-1)$, the problem consists of finding two polynomial matrices $\boldsymbol{A}(z^{-1})$ and $\boldsymbol{B}(z^{-1})$ so that $\boldsymbol{H}(z^{-1})=\boldsymbol{A}^{-1}(z^{-1})\boldsymbol{B}(z^{-1})$ holds. The simplest way of doing this is by making $\boldsymbol{A}(z^{-1})$ a diagonal matrix with its elements equal to the least common denominators of the corresponding row of $\boldsymbol{H}(z^{-1})$. Let

$$\begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{bmatrix} = \begin{bmatrix} \frac{\bar{B}_{11}}{\bar{A}_{11}} & \frac{\bar{B}_{12}}{\bar{A}_{12}} & \cdots & \frac{\bar{B}_{1p}}{\bar{A}_{1p}} \\ \frac{\bar{B}_{21}}{\bar{A}_{21}} & \frac{\bar{B}_{22}}{\bar{A}_{22}} & \cdots & \frac{\bar{B}_{2p}}{\bar{A}_{2p}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\bar{B}_{m1}}{\bar{A}_{m1}} & \frac{\bar{B}_{m2}}{\bar{A}_{m1}} & \cdots & \frac{\bar{B}_{mp}}{\bar{A}_{mp}} \end{bmatrix} \begin{bmatrix} u_1(t-1) \\ u_2(t-1) \\ \vdots \\ u_p(t-1) \end{bmatrix} \tag{C.1}$$

where the first row of $\boldsymbol{H}(z^{-1})$

$$\begin{aligned} \bar{A}_{11}(z^{-1}) &= 1+\bar{a}_{1_{11}}z^{-1}+\bar{a}_{2_{11}}z^{-2}+\ldots+\bar{a}_{na_{11_{11}}}z^{-nA_{11}} \\ \bar{A}_{12}(z^{-1}) &= 1+\bar{a}_{1_{12}}z^{-1}+\bar{a}_{2_{12}}z^{-2}+\ldots+\bar{a}_{na_{12_{12}}}z^{-nA_{12}} \\ &\vdots \\ \bar{A}_{1p}(z^{-1}) &= 1+\bar{a}_{1_{1p}}z^{-1}+\bar{a}_{2_{1p}}z^{-2}+\ldots+\bar{a}_{na_{11_{1p}}}z^{-nA_{1p}} \end{aligned} \tag{C.2}$$

and

$$\begin{aligned}
\overline{B}_{11}(z^{-1}) &= \overline{b}_{0_{11}} + \overline{b}_{1_{11}} z^{-1} + \overline{b}_{2_{11}} z^{-2} + \ldots + \overline{b}_{nb_{11_{11}}} z^{-nB_{11}} \\
\overline{B}_{12}(z^{-1}) &= \overline{b}_{0_{12}} + \overline{b}_{1_{12}} z^{-1} + \overline{b}_{2_{12}} z^{-2} + \ldots + \overline{b}_{nb_{12_{12}}} z^{-nB_{12}} \\
&\vdots \\
\overline{B}_{1p}(z^{-1}) &= \overline{b}_{0_{1p}} + \overline{b}_{1_{1p}} z^{-1} + \overline{b}_{2_{1p}} z^{-2} + \ldots + \overline{b}_{nb_{1p_{1p}}} z^{-nB_{1p}}
\end{aligned} \tag{C.3}$$

Suppose no common denominators exist in matrix $\boldsymbol{H}$. Multiply matrix $\boldsymbol{H}$ by $\boldsymbol{A}$ such that the resulting matrix, $\boldsymbol{B}$, is a polynomial matrix.

$$\boldsymbol{AH} = \boldsymbol{B}$$

$$\underbrace{\begin{bmatrix} \overline{A}_{11}\overline{A}_{12}\ldots\overline{A}_{1p} & 0 & \cdots & 0 \\ 0 & \overline{A}_{21}\overline{A}_{22}\ldots\overline{A}_{2p} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \overline{A}_{m1}\overline{A}_{m2}\ldots\overline{A}_{mp} \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \dfrac{\overline{B}_{11}}{\overline{A}_{11}} & \dfrac{\overline{B}_{12}}{\overline{A}_{12}} & \cdots & \dfrac{\overline{B}_{1p}}{\overline{A}_{1p}} \\ \dfrac{\overline{B}_{21}}{\overline{A}_{21}} & \dfrac{\overline{B}_{22}}{\overline{A}_{22}} & \cdots & \dfrac{\overline{B}_{2p}}{\overline{A}_{2p}} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\overline{B}_{m1}}{\overline{A}_{m1}} & \dfrac{\overline{B}_{m2}}{\overline{A}_{m1}} & \cdots & \dfrac{\overline{B}_{mp}}{\overline{A}_{mp}} \end{bmatrix}}_{\boldsymbol{H}}$$

$$= \underbrace{\begin{bmatrix} \overline{A}_{12}\overline{A}_{13}\ldots\overline{A}_{1p}\overline{B}_{11} & \overline{A}_{11}\overline{A}_{13}\ldots\overline{A}_{1p}\overline{B}_{12} & \cdots & \overline{A}_{11}\overline{A}_{12}\ldots\overline{A}_{1(p-1)}\overline{B}_{1p} \\ \overline{A}_{22}\overline{A}_{23}\ldots\overline{A}_{2p}\overline{B}_{21} & \overline{A}_{21}\overline{A}_{23}\ldots\overline{A}_{2p}\overline{B}_{22} & \cdots & \overline{A}_{21}\overline{A}_{22}\ldots\overline{A}_{2(p-1)}\overline{B}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{A}_{m2}\overline{A}_{m3}\ldots\overline{A}_{mp}\overline{B}_{m1} & \overline{A}_{m1}\overline{A}_{m3}\ldots\overline{A}_{mp}\overline{B}_{m2} & \cdots & \overline{A}_{m1}\overline{A}_{m2}\ldots\overline{A}_{m(p-1)}\overline{B}_{mp} \end{bmatrix}}_{\boldsymbol{B}}$$

defining

$$\begin{aligned}
A_{11} &= \overline{A}_{11}\overline{A}_{12}\ldots\overline{A}_{1p} \\
A_{22} &= \overline{A}_{21}\overline{A}_{22}\ldots\overline{A}_{2p} \\
&\vdots \\
A_{mm} &= \overline{A}_{m1}\overline{A}_{m2}\ldots\overline{A}_{mp}
\end{aligned}$$

then $\boldsymbol{A}$ is

$$\boldsymbol{A}=\begin{bmatrix} A_{11} & 0 & \cdots & 0 \\ 0 & A_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{mm} \end{bmatrix}$$

and defining

$$\begin{array}{lll} B_{11}=\overline{A}_{12}\overline{A}_{13}\ldots\overline{A}_{1p}\overline{B}_{11} & B_{21}=\overline{A}_{11}\overline{A}_{13}\ldots\overline{A}_{1p}\overline{B}_{12} & B_{p1}=\overline{A}_{11}\overline{A}_{12}\ldots\overline{A}_{1(p-1)}\overline{B}_{1p} \\ B_{21}=\overline{A}_{22}\overline{A}_{23}\ldots\overline{A}_{2p}\overline{B}_{21} & B_{22}=\overline{A}_{21}\overline{A}_{23}\ldots\overline{A}_{2p}\overline{B}_{22} & B_{p2}=\overline{A}_{21}\overline{A}_{22}\ldots\overline{A}_{2(p-1)}\overline{B}_{2p} \\ \vdots & \vdots & \vdots \\ B_{m1}=\overline{A}_{m2}\overline{A}_{m3}\ldots\overline{A}_{mp}\overline{B}_{m1} & B_{m2}=\overline{A}_{m1}\overline{A}_{m3}\ldots\overline{A}_{mp}\overline{B}_{m2} & B_{pm}=\overline{A}_{m1}\overline{A}_{m2}\ldots\overline{A}_{m(p-1)}\overline{B}_{mp} \end{array}$$

matrix $\boldsymbol{B}$ is

$$\boldsymbol{B}=\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1p} \\ B_{21} & B_{22} & \cdots & B_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mp} \end{bmatrix}$$

and the deterministic auto-regressive moving-average (DARMA) representation of the process is obtained

$$\boldsymbol{A}\boldsymbol{y}(t)=\boldsymbol{B}\boldsymbol{u}(t-1)$$

Matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ do not have to be coprime in general and may still be used to implement the predictive controller. However this will generate a controller with higher degree and may in some cases result in less efficient algorithms. To guarantee coprimeness of the matrix fraction description the greatest common divisor must be found and extracted from matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

APPENDIX D
CLOSED-LOOP TRANSFER MATRICES

D.1 Disturbances in the Closed-loop

Figure D.1 shows the closed-loop system subject to a series of disturbance signals. These signals represent external disturbances and measurement noise, common to real systems.

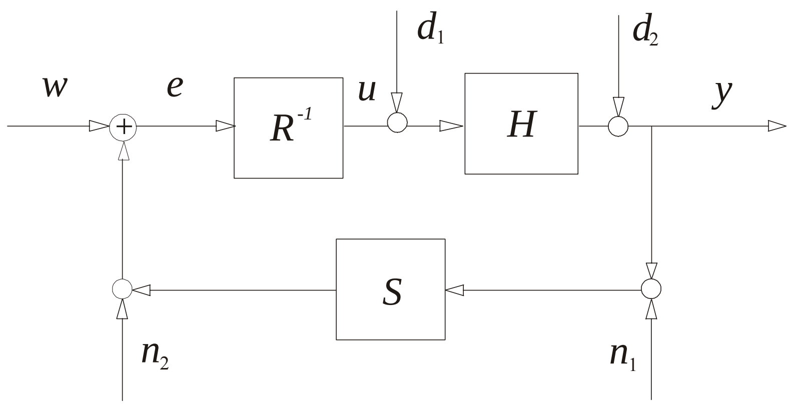

Figure D.1 Closed-loop system with external disturbance and noise signals

In the sequel all the different transfer functions will be derived for all the signals seen in Figure D.1. Thus we have

$$(\boldsymbol{I} + \boldsymbol{H}\boldsymbol{R}^{-1}\boldsymbol{S})\boldsymbol{y} = -\boldsymbol{H}\boldsymbol{d}_1 + \boldsymbol{d}_2 - \boldsymbol{H}\boldsymbol{R}^{-1}\boldsymbol{S}\boldsymbol{n}_1 + \boldsymbol{H}\boldsymbol{R}^{-1}(\boldsymbol{w} - \boldsymbol{n}_2)$$

$$(\boldsymbol{I} + \boldsymbol{R}^{-1}\boldsymbol{S}\boldsymbol{H})\boldsymbol{u} = -\boldsymbol{R}^{-1}\boldsymbol{S}\boldsymbol{H}\boldsymbol{d}_1 - \boldsymbol{R}^{-1}\boldsymbol{S}(\boldsymbol{d}_2 + \boldsymbol{n}_1) + \boldsymbol{R}^{-1}(\boldsymbol{w} - \boldsymbol{n}_2)$$

$$(\boldsymbol{I} + \boldsymbol{S}\boldsymbol{H}\boldsymbol{R}^{-1})\boldsymbol{v} = \quad \boldsymbol{S}\boldsymbol{H}\boldsymbol{d}_1 + \boldsymbol{S}(\boldsymbol{d}_2 + \boldsymbol{n}_1) + \boldsymbol{S}\boldsymbol{H}\boldsymbol{R}^{-1}(\boldsymbol{w} - \boldsymbol{n}_2)$$

$$(\boldsymbol{I} + \boldsymbol{S}\boldsymbol{H}\boldsymbol{R}^{-1})\boldsymbol{e} = -\boldsymbol{S}\boldsymbol{H}\boldsymbol{d}_1 - \boldsymbol{S}(\boldsymbol{d}_2 + \boldsymbol{n}_1) + (\boldsymbol{w} - \boldsymbol{n}_2)$$

It is immediate to see that $\boldsymbol{n}_2$ may be discarded from the analysis once it is redundant to signal $\boldsymbol{w}$. Before proceeding, some helpful properties of transfer matrices will be shown.

PROPERTY D.1 From the Matrix Inversion Lemma it is known that $(\boldsymbol{A}+\boldsymbol{BCD})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{DA}^{-1}\boldsymbol{B}+\boldsymbol{C}^{-1})^{-1}\boldsymbol{DA}^{-1}$ where all the inverted matrices possess inverses. As a consequence we have:

D.1.1 $(\boldsymbol{I}+\boldsymbol{B})^{-1} = \boldsymbol{I} - \boldsymbol{B}(\boldsymbol{I}+\boldsymbol{B})^{-1} = \boldsymbol{I} - (\boldsymbol{I}+\boldsymbol{B})^{-1}\boldsymbol{B}$

D.1.2 $(\boldsymbol{I}+\boldsymbol{BD})^{-1} = \boldsymbol{I} - \boldsymbol{B}(\boldsymbol{I}+\boldsymbol{DB})^{-1}\boldsymbol{D}$

D.1.3 $(\boldsymbol{I}+\boldsymbol{DB})^{-1}\boldsymbol{D} = \boldsymbol{D}(\boldsymbol{I}+\boldsymbol{BD})^{-1}$ ■

Matrices $(\boldsymbol{I}+\boldsymbol{B})^{-1}$ and $(\boldsymbol{I}+\boldsymbol{B})^{-1}\boldsymbol{B}$ (or $\boldsymbol{B}(\boldsymbol{I}+\boldsymbol{B})^{-1}$) are called complimentary matrices. Stability of a transfer matrix implies in stability of its complimentary transfer matrix. This is a consequence of the fact that the sum of two stable transfer matrices generates a stable matrix. That way only one the two complimentary transfer matrices needs to be analyzed.

All transfer matrices for $\boldsymbol{v}$ can be discarded once $\boldsymbol{G}_{vd_1} = -\boldsymbol{G}_{ed_1}$, $\boldsymbol{G}_{v(d_2+n_1)} = -\boldsymbol{G}_{e(d_2+n_1)}$ and $\boldsymbol{G}_{vw}$ and $\boldsymbol{G}_{ew}$ are complimentary transfer matrices. Transfer matrices $\boldsymbol{G}_{yd_2}$ and $\boldsymbol{G}_{yn_1}$ are also complimentary and we have chosen to discard the first one. That way signals $\boldsymbol{d}_2$ and $\boldsymbol{n}_1$ become redundant and choosing to discard $\boldsymbol{d}_2$ we are left with:

$$\boldsymbol{y} = -(\boldsymbol{I}+\boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{Hd}_1 - (\boldsymbol{I}+\boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{HR}^{-1}\boldsymbol{Sn}_1 + (\boldsymbol{I}+\boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{HR}^{-1}\boldsymbol{w}$$

$$\boldsymbol{u} = -(\boldsymbol{I}+\boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}\boldsymbol{SHd}_1 - (\boldsymbol{I}+\boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}\boldsymbol{Sn}_1 + (\boldsymbol{I}+\boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}\boldsymbol{w}$$

$$\boldsymbol{e} = -(\boldsymbol{I}+\boldsymbol{SHR}^{-1})^{-1}\boldsymbol{SHd}_1 - (\boldsymbol{I}+\boldsymbol{SHR}^{-1})^{-1}\boldsymbol{Sn}_1 + (\boldsymbol{I}+\boldsymbol{SHR}^{-1})^{-1}\boldsymbol{w}$$

Then we have

$\boldsymbol{G}_{yd_1} = -(\boldsymbol{I} + \boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{H}$, $\boldsymbol{G}_{yn_1} = -(\boldsymbol{I} + \boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{HR}^{-1}\boldsymbol{S}$, $\boldsymbol{G}_{yw} = -(\boldsymbol{I} + \boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{HR}^{-1}$

$\boldsymbol{G}_{ud_1} = -(\boldsymbol{I} + \boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}\boldsymbol{SH}$, $\boldsymbol{G}_{un_1} = -(\boldsymbol{I} + \boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}\boldsymbol{S}$, $\boldsymbol{G}_{uw} = (\boldsymbol{I} + \boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}\boldsymbol{R}^{-1}$

$\boldsymbol{G}_{ed_1} = -(\boldsymbol{I} + \boldsymbol{SHR}^{-1})^{-1}\boldsymbol{SHd}_1$, $\boldsymbol{G}_{en_1} = -(\boldsymbol{I} + \boldsymbol{SHR}^{-1})^{-1}\boldsymbol{S}$, $\boldsymbol{G}_{ew} = (\boldsymbol{I} + \boldsymbol{SHR}^{-1})^{-1}$

D.2 Closed-loop Transfer Matrices of Predictive Control

Substituting $\boldsymbol{H} = \boldsymbol{BA}^{-1}$ and bearing in mind that $\boldsymbol{A}$ and $\boldsymbol{R}$ are invertible matrices we have $\boldsymbol{G}_{yd_1} = -\boldsymbol{B\Omega}^{-1}\boldsymbol{R}$ where $\boldsymbol{\Omega} = \boldsymbol{RA} + \boldsymbol{SB}$. To prove this recall from Property **D**.1.3 that $\boldsymbol{G}_{yd_1} = -(\boldsymbol{I} + \boldsymbol{HR}^{-1}\boldsymbol{S})^{-1}\boldsymbol{H} = -\boldsymbol{H}(\boldsymbol{I} + \boldsymbol{R}^{-1}\boldsymbol{SH})^{-1}$. Then its immediate to conclude that $\boldsymbol{G}_{yn_1} = -\boldsymbol{B\Omega}^{-1}\boldsymbol{S}$ and $\boldsymbol{G}_{yw} = -\boldsymbol{B\Omega}^{-1}$. Similar manipulations and substitutions give $\boldsymbol{G}_{uw} = \boldsymbol{A\Omega}^{-1}$, $\boldsymbol{G}_{un_1} = -\boldsymbol{A\Omega}^{-1}\boldsymbol{S}$, $\boldsymbol{G}_{ud_1} = \boldsymbol{I} - \boldsymbol{A\Omega}^{-1}\boldsymbol{R}$, $\boldsymbol{G}_{ew} = \boldsymbol{RA\Omega}^{-1}$, $\boldsymbol{G}_{en_1} = -\boldsymbol{RA\Omega}^{-1}\boldsymbol{S}$ and $\boldsymbol{G}_{ed_1} = -\boldsymbol{SB\Omega}^{-1}\boldsymbol{R}$. We then have:

$$\boldsymbol{y} = -\boldsymbol{B\Omega}^{-1}\boldsymbol{Rd}_1 - \boldsymbol{B\Omega}^{-1}\boldsymbol{Sn}_1 - \boldsymbol{B\Omega}^{-1}\boldsymbol{w}$$

$$\boldsymbol{u} = (\boldsymbol{I} - \boldsymbol{A\Omega}^{-1}\boldsymbol{R})\boldsymbol{d}_1 - \boldsymbol{A\Omega}^{-1}\boldsymbol{Sn}_1 + \boldsymbol{A\Omega}^{-1}\boldsymbol{w}$$

$$\boldsymbol{e} = -\boldsymbol{SB\Omega}^{-1}\boldsymbol{Rd}_1 - \boldsymbol{RA\Omega}^{-1}\boldsymbol{Sn}_1 + \boldsymbol{RA\Omega}^{-1}\boldsymbol{w}$$

Recalling that all matrices are polynomial matrices (in z^{-1}) then it is seen that transfer matrices $\boldsymbol{G}_{uw} = \boldsymbol{A\Omega}^{-1}$ and $\boldsymbol{G}_{yw} = -\boldsymbol{B\Omega}^{-1}$ are fundamental in the analysis of stability. All other matrices are the product of either one of these two matrices and a polynomial matrix. This way if either $\boldsymbol{G}_{uw} = \boldsymbol{A\Omega}^{-1}$ or $\boldsymbol{G}_{yw} = -\boldsymbol{B\Omega}^{-1}$ is unstable then the closed-loop system will be unstable. Note that stability of $\boldsymbol{G}_{yw} = -\boldsymbol{B\Omega}^{-1}$ does not imply stability of $\boldsymbol{G}_{uw} = \boldsymbol{A\Omega}^{-1}$ and both matrices should be checked in order to assume stability of the closed-loop.

APPENDIX E
OBTAINING MATRIX $\boldsymbol{K}$

Proof. Expand expression $\Delta\boldsymbol{u} = (\boldsymbol{G}^T \boldsymbol{\Phi} \boldsymbol{G} + \boldsymbol{\Lambda})^{-1} \boldsymbol{G}^T \boldsymbol{\Phi}(\boldsymbol{r} - \boldsymbol{y}^0)$ along the prediction and control horizons to get

$$
\begin{bmatrix}
\begin{bmatrix} \Delta u_1(t) \\ \Delta u_1(t+1) \\ \vdots \\ \Delta u_1(t+Nu_1) \end{bmatrix} \\
\begin{bmatrix} \Delta u_2(t) \\ \Delta u_2(t+1) \\ \vdots \\ \Delta u_2(t+Nu_2) \end{bmatrix} \\
\vdots \\
\begin{bmatrix} \Delta u_p(t) \\ \Delta u_p(t+1) \\ \vdots \\ \Delta u_p(t+Nu_p) \end{bmatrix}
\end{bmatrix}
=
\begin{bmatrix}
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \\ k_{21} & k_{22} & \cdots & k_{2Ny_1} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_1 1} & k_{Nu_1 2} & \cdots & k_{Nu_1 Ny_1} \end{bmatrix}_{11} &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \\ k_{21} & k_{22} & \cdots & k_{2Ny_2} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_1 1} & k_{Nu_1 2} & \cdots & k_{Nu_1 Ny_2} \end{bmatrix}_{12} & \cdots &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \\ k_{21} & k_{22} & \cdots & k_{2Ny_m} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_1 1} & k_{Nu_1 2} & \cdots & k_{Nu_1 Ny_m} \end{bmatrix}_{1m} \\
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \\ k_{21} & k_{22} & \cdots & k_{2Ny_1} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_2 1} & k_{Nu_2 2} & \cdots & k_{Nu_2 Ny_1} \end{bmatrix}_{21} &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \\ k_{21} & k_{22} & \cdots & k_{2Ny_2} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_2 1} & k_{Nu_2 2} & \cdots & k_{Nu_2 Ny_2} \end{bmatrix}_{22} & \cdots &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \\ k_{21} & k_{22} & \cdots & k_{2Ny_m} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_2 1} & k_{Nu_2 2} & \cdots & k_{Nu_2 Ny_m} \end{bmatrix}_{2m} \\
\vdots & \vdots & \ddots & \vdots \\
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \\ k_{21} & k_{22} & \cdots & k_{2Ny_1} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_p 1} & k_{Nu_p 2} & \cdots & k_{Nu_p Ny_1} \end{bmatrix}_{p1} &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \\ k_{21} & k_{22} & \cdots & k_{2Ny_2} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_p 1} & k_{Nu_p 2} & \cdots & k_{Nu_p Ny_2} \end{bmatrix}_{p2} & \cdots &
\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \\ k_{21} & k_{22} & \cdots & k_{2Ny_m} \\ \vdots & \vdots & \ddots & \vdots \\ k_{Nu_p 1} & k_{Nu_p 2} & \cdots & k_{Nu_p Ny_m} \end{bmatrix}_{pm}
\end{bmatrix}
\begin{bmatrix}
\begin{bmatrix} r_1(t+1) - y_1{}^0(t+1) \\ r_1(t+2) - y_1{}^0(t+2) \\ \vdots \\ r_1(t+Ny_1) - y_1{}^0(t+Ny_1) \end{bmatrix} \\
\begin{bmatrix} r_2(t+1) - y_2{}^0(t+1) \\ r_2(t+2) - y_2{}^0(t+2) \\ \vdots \\ r_2(t+Ny_2) - y_2{}^0(t+Ny_2) \end{bmatrix} \\
\vdots \\
\begin{bmatrix} r_m(t+1) - y_m{}^0(t+1) \\ r_m(t+2) - y_m{}^0(t+2) \\ \vdots \\ r_m(t+Ny_m) - y_m{}^0(t+Ny_m) \end{bmatrix}
\end{bmatrix}
$$

and because the first control move alone is of interest, subsequent terms ($i = 2$ to Nu_ℓ) are discarded and the above expression becomes

$$\begin{bmatrix} \Delta u_1(t) \\ \Delta u_2(t) \\ \vdots \\ \Delta u_p(t) \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \end{bmatrix}_{11} & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \end{bmatrix}_{12} & \cdots & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \end{bmatrix}_{1m} \\ \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \end{bmatrix}_{21} & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \end{bmatrix}_{22} & \cdots & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \end{bmatrix}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_1} \end{bmatrix}_{p1} & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_2} \end{bmatrix}_{p2} & \cdots & \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1Ny_m} \end{bmatrix}_{pm} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} r_1(t+1) - y_1{}^0(t+1) \\ r_1(t+2) - y_1{}^0(t+2) \\ \vdots \\ r_1(t+Ny_1) - y_1{}^0(t+Ny_1) \end{bmatrix} \\ \begin{bmatrix} r_2(t+1) - y_2{}^0(t+1) \\ r_2(t+2) - y_2{}^0(t+2) \\ \vdots \\ r_2(t+Ny_2) - y_2{}^0(t+Ny_2) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} r_m(t+1) - y_m{}^0(t+1) \\ r_m(t+2) - y_m{}^0(t+2) \\ \vdots \\ r_m(t+Ny_m) - y_m{}^0(t+Ny_m) \end{bmatrix} \end{bmatrix}$$

Eliminating the, now unnecessary, first index of each element of the rows of the new matrix then $[k_{11} \quad k_{12} \quad \cdots \quad k_{1N_{y_j}}]_{\ell j}$ becomes

$[k_1 \quad k_2 \quad \cdots \quad k_{N_{y_j}}]_{\ell j}$ which are the elements of matrix $\boldsymbol{K}$. *Q.E.D*

LIST OF REFERENCES

Bitmead, R.R., Gevers, M., Wertz, V., *Adaptive Optimal Control – The Thinking Man's GPC*, Prentice Hall of Australia Pty Ltd, Australia (1990).

Camacho, E.F., Bordons, C., *Model Predictive Control*, Springer-Verlag London Limited, Great Britain (1999).

Clarke, D.W., Mohtadi, C., Tuffs, P.S., "Generalized Predictive Control - Part II. Extensions and Interpretations," *Automatica*, 23, 2, pp 149-160 (1987).

Clarke, D.W., Scattolini, R., "Constrained Receding-horizon Predictive Control," *Proceedings IEE*, 138, 4, pp 347-354 (1991).

Crisalle, O.D., Seborg, D.E., Mellichamp, D.A., "Theoretical Analysis of Long-Range Predictive Controllers," *American Control Conference*, Vol. 1, pp 570-576, Pittsburgh, PA (1989).

Cutler, C.R., Ramaker, B.C., "Dynamic Matrix Control - A Computer Control Algorithm," *Automatic Control Conference*, San Francisco, CA (1980).

De Keyser, R.M.C., Van Cuawenberghe, A.R., "Extended Prediction Self-adaptive Control," *IFAC: Symposium on Identification and System Parameter Estimation*, York, UK, pp 1317-1322 (1985).

Dorf, R.C., Bishop, R.H., *Modern Control Systems*, Addison Wesley Longman, Inc., 8th ed., Reading, MA (1998).

Gantmacher, F.R., *The Theory of Matrices*, Chelsea Publishing Company, Vol. 1, New York, NY (1977).

Garcia, C.E., Prett, D.M., Morari, M., "Model Predictive Control: Theory and Practice – A Survey, *Automatica*, 25, 3, pp 335-348 (1989).

Hrissagis, K., Crisalle, O.D., "Robust Unconstrained Predictive Control Design with Guaranteed Nominal Performance, *AIChE Journal*, 42, 5, pp 1293-1303 (1996).

Kaylath, T., *Linear Systems*, Prentice Hall, Englewood Cliffs, NJ, (1980).

Keerthi, S.S., Gilbert, E.G., "Optimal Infinite-Horizon Feedback Laws for a General Class of Constrained Discrete-Time Systems: Stability and Moving-Horizon Approximations," *Journal of Optimization Theory and Applications*, 57, 2 (1988).

Kouvaritakis, B., Rossiter, J.A., Chang, A.O.T., "Stable Generalized Predictive Control: An Algorithm with Guaranteed Stability," *Proceedings IEE*, Part D, 139, 4, pp 349-362 (1992).

Kwon, W.H., Pearson, A.E., "A Modified Quadratic Cost Problem and Feedback Stabilization of a Linear System," *IEEE Transactions on Automatic Control*, AC-22, 5, pp 838-842 (1977).

Lay, D.C., *Linear Algebra and its Applications*, Addison Wesley Longman, Inc., 2nd ed., Reading, MA (1998).

Luyben, W.L., *Process Modelling, Simulation and Control for Engineers*, McGraw-Hill Book Co., 2nd ed., Singapore (1990).

Maciejowski, J.M., *Multivariable Feedback Design*, Addison-Wesley Publishers Ltd., Cornwall, Great Britain (1989).

Mayne, D.Q., Michalska, H., "Robust Receding Horizon Control of Constrained Nonlinear Systems," *IEEE Transactions on Automatic Control*, 38, pp 1623-1633 (1993).

Mayne, D.Q., Rawlings, J.B., Rao, C.V., Skokaert, P.O., "Constrained Model Predictive Control: Stability and optimality," *Automatica*, 36, pp 789-814 (2000).

Morari, M., Lee, J.H., "Model Predictive Control: Past, Present and Future," *Computers and Chemical Engineering*, 23, pp 667-682 (1999).

Mosca, E., Lemos, J.M., Zhang, J., "Stabilizing I/O Receding Horizon Control," *IEEE Conference on Decision and Control* (1990).

Muske, K.R., Rawlings, J., "Model Predictive Control with Linear Models," *AIChE Journal*, 39, pp 262-287 (1993).

Nunes, G.C., "Modelagem e Simulação Dinâmica de Separador Trifásico Água-Óleo-Gás," M.Sc. Thesis, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil (1994).

Ogata, K., *Discrete-Time Control Systems*, Prentice-Hall Inc., Englewood Cliffs, NJ (1987).

Peterka, V., "Predictor-based Self-tuning Control," *Automatica*, 20, 1, pp 39-50 (1984).

Primbs, J.A., Nevistic, V., "Feasibility and Stability of Constrained Finite Receding Horizon Control," *Automatica*, 36, pp 965-971 (2000).

Rawlings, J.B., Muske, K.R., "The Stability of Constrained Receding Horizon Control," *IEEE Transactions on Automatic Control*, 38, 10 (1993).

Richalet, J., Rault, A., Testud, J.L., Papon, J., "Model Predictive Heuristic Control: Application to Industrial Processes," *Automatica*, 14, 2, pp 413-428 (1978).

Skogestad, S., and Postlethwaite, I., *Multivariable Feedback Control – Analysis and Design*, John Wiley and Sons, New York, NY (1996).

Strum, R.D., Kirk, D.E., *Contemporary Linear Systems Using Matlab 4.0*, PWS Publishing Company, Boston, MA (1996).

Qin, S.J., Badgwell, T.A., "An Overview of Industrial Model Predictive Control Technology," *Proceedings of the 5th International Conference on Chemical Process Control*, CPC-V, AIChE Symposium Series No. 316, Vol. 93, pp 232-256. Tahoe City, CA, (1996)

Visyasagar, M., *Control System Synthesis - A Factorization Approach*, The MIT Press, Cambridge, MA (1985).

Zhou, K., *Essentials of Robust Control*, Prentice Hall, Upper Saddle River, NJ (1998).

BIOGRAPHICAL SKETCH

Giovani Cavalcanti Nunes was born in the city of Caruaru, state of Pernambuco in Brazil on April 21, 1961. He graduated with a B.S. in chemical engineering from the Universidade do Estado do Rio de Janeiro in Rio de Janeiro, Brazil in December of 1986. He joined PETROBRAS S.A. as a process engineer in March of 1987 and has been there ever since. In October, 1994, he graduated with a M.Sc. degree in chemical engineering from the Universidade Federal do Rio de Janeiro. He joined the graduate program at the University of Florida in August of 1997 and graduated with a Ph.D. in chemical engineering in July of 2001.