

**IDENTIFICATION AND MODEL PREDICTIVE
CONTROL OF THE RAW MATERIAL BLENDING
PROCESS IN CEMENT INDUSTRY**

Ph.D. Thesis by

Ayhan KURAL, M.Sc.

(Registration Number : 503920032012)

104274

Date of submission : 5 February 2001

Date of defence examination: 15 June 2001

Supervisor (Chairman): Prof. Dr. Can ÖZSOY (İ.T.Ü.)

Members of the Examining Committee Prof.Dr. Ahmet KUZUCU(İ.T.Ü.)

Assoc.Prof.Dr. Kenan Kutlu (İ.T.Ü.)

Prof.Dr. Talha DİNİBÜTÜN (Doğuş Ü.)

Prof.Dr. İbrahim YÜKSEL(Uludağ Ü.)

JUNE 2001

**ÇİMENTO ENDÜSTRİSİNDE HAMMADE
HARMANLAMA PROSESİNİN TANILANMASI VE
MODEL ÖNGÖRÜLÜ KONTROLÜ**

İ. T. Ü. Fen Bilimleri Enstitüsü
Çimento Endüstrisinde Hamma

DOKTORA TEZİ
Y. Müh. Ayhan Kural
(Enstitü No: 503920032012)

Tezin Enstitüye Verildiği Tarih : 5 Şubat 2001
Tezin Savunulduğu Tarih : 15 Haziran 2001

Tez Danışmanı :

Prof.Dr. Can ÖZSOY (İ.T.Ü.)

Diğer Jüri Üyeleri

Prof.Dr. Ahmet KUZUCU (İ.T.Ü.)

Doç.Dr. Kenan KUTLU (İ.T.Ü.)

Prof.Dr. Talha DİNİBÜTÜN (Doğuş Ü.)

Prof.Dr. İbrahim YÜKSEL (Uludağ Ü.)

HAZİRAN 2001

FOREWORD

I wish to express my sincere appreciation and gratitude to my supervisor, Prof. Can Özsoy, for his guidance, support and encouragement and to my friend, İlker Murat Koç for his help and support during this thesis.

I would like to thank to Mr. Atalay Şahinoğlu, General Manager of Nuh Cement Factory, and Mr. Mehmet Çetinkaya, Technical Manager of Nuh Cement Factory, to give me the permission to receive all data and information about the raw material blending process in Nuh Cement Factory.

I also wish to thank to Prof. Dr. Ahmet Kuzucu for his suggestions and to my colleague, Mr. Bülent Bölüt for his help and support.

Thanks are also my teachers and colleagues in Automatic Control Department for their understanding, patience and moral support throughout the whole period of my study.

Finally, I would like to thank to my family who have supported me throughout my academic career.

February 2001

Ayhan Kural

TABLE OF CONTENTS

SHORTENINGS	v
LIST OF TABLES	vi
LIST OF FIGURES	vii
LIST OF SYMBOL	viii
SUMMARY	x
ÖZET	xiii
1. INTRODUCTION	1
2. RAW MATERIAL BLENDING SYSTEM IN CEMENT INDUSTRY	4
2.1. Raw Material Blending Process	4
2.2. Structure of the Cement Manufacturing Process and Chemical Equations	6
2.3. Analysis Systems of the Raw Material Blending Process	8
3. SYSTEM IDENTIFICATION	10
3.1. Introduction	11
3.2. System Identification Principles	12
3.2.1. Selection of the sampling frequency	17
3.2.2. Some commonly used input signals for system identification	18
3.2.3. Signal conditioning	23
3.3. Effects of Random Disturbances Upon Parameter Identification	24
3.4. Structure of the Recursive Identification Method	24
3.4.1. Estimation of the model complexity	26
3.4.2. Complexity estimation with white noise	26
3.5. Parameter Adaptation Algorithm (P.A.A)	27
3.5.1. Choice of the adaptation gain	29
3.5.2. Constant forgetting factor	30
3.6. Parameterization of Multivariable Systems	31
3.6.1. Properties of the parameterization	33
3.6.2. Canonical parameterization (Guidorzi)	35
3.6.3. Connection between state space and ARX model	35
3.7. Validation of the Model	37
3.7.1. Whiteness test	38
4. IDENTIFICATION OF THE RAW MATERIAL BLENDING PROCESS	41
4.1. Process Models	41
4.2. Model Structure Estimation of the Raw Material Blending Process	43
4.3. Parameter Estimation (RLS) of the Raw Material Blending Process	45
4.4. The Whiteness Test of the Raw Material Blending Process	46

4.5. Identification Results	47
5. MODEL BASED PREDICTIVE CONTROL (MPC)	52
5.1. Historical Perspective	55
5.2. MPC Strategy	58
5.3. Discrete Time Step Response Models	60
5.4. The Dynamic Matrix Equation	62
5.5. Prediction Vector	63
5.6. The Dynamic Matrix Control Algorithm	65
5.7. Extension to Multivariable Processes	67
5.8. Tuning Predictive Controller	68
6. MODEL PREDICTIVE CONTROL (MPC) OF THE RAW MATERIAL BLENDING PROCESS	70
6.1. Introduction	70
6.2. Prediction Models	70
6.3. Objective Function	72
6.4. Control Law for the Raw Material Blending Process	72
6.5. Simulation Results	73
7. CONCLUSION AND COMMENTS	80
BIBLIOGRPHY	83
APPENDIX A	89
AUTOBIOGRAPHY	99

SHORTENINGS

PGNAA	: Prompt Gamma Neutron Activation Analyzers
XRF	: X-ray Fluorescence
ML	: Lime Module
MA	: Aluminum Module
MS	: Silica Module
SISO	: Single-input Single-output
MIMO	: Multi-input Multi-output
MISO	: Multi-input Single-output
ARMA	: Autoregressive Moving Average
ARMAX	: Auto Regressive Moving Average with Exogenous Input
ARX	: Auto Regressive with Exogenous Input
RLS	: Recursive Least Squares Algorithm
LS	: Least Square
BIC	: Bayesian Information Criterion
PAA	: Parameter Adaptation Algorithm
PRBS	: Pseudorandom Binary Sequence
DC	: Stationary
MV	: Mean Value
MPC	: Model Based Predictive Control
GPC	: Generalized Predictive Control
DMC	: Dynamic Matrix Control
EPSAC	: Extended Prediction Self-Adaptive Control
PFC	: Predictive Functional Control
EHAC	: Extended Horizon Adaptive Control
UPC	: Unified Predictive Control
MPHC	: Model Predictive Heuristic Control
MAC	: Model Algorithmic Control
MUSMAR	: Multistep Multivariable Adaptive Control
MURHAC	: Multipredictor Receding Horizon Adaptive Control

LIST OF TABLES

	<u>Page No</u>
Tablo 2.1. Blending feedstream data	5



LIST OF FIGURES

	<u>Page No</u>
Figure 2.1 : Simple schema of blending control in Nuh Cement Factor.....	5
Figure 3.1 : The system identification loop	12
Figure 3.2 : The parameter estimation principle for sampled model	14
Figure 3.3 : Generation of a PRBS of length $2^5 - 1 = 31$ samplingperiod.....	21
Figure 3.4 : Choice of the maximum duration of a pulse in a PRBS.....	22
Figure 3.5 : Structure of recursive identification method.....	24
Figure 3.6 : Structure of recursive least square (R.L.S)	25
Figure 4.1 : $V_{Lsi}(\hat{n}_i)$ as function of $\hat{n}_i$	44
Figure 4.2 : Typical evolution of $V_{Lsi}(n_i)$, $d_iX(N)$ and $CV_{Lsi}(n_i)$ as function of n_i	45
Figure 4.3 : Measured input-output data of the process.....	47
Figure 4.4 : Filtered input-output data of the process.....	48
Figure 4.5 : Measured and predicted outputs of the model 1	48
Figure 4.6 : Step response of the model 1	48
Figure 4.7 : Measured and predicted outputs of the model 2	49
Figure 4.8 : Step response of model 2	49
Figure 4.9 : Measured and predicted outputs of the model 3	50
Figure 4.10 : Step response of the model 3	51
Figure 5.1 : Controlled and manipulated variables for a moving horizon	58
Figure 5.2 : Basic structure of MPC	59
Figure 6.1 : The block diagram of closed-loop state space models in the MPC	71
Figure 6.2 : Inputs-output of controlled process for model 1 at the beginning...	74
Figure 6.3 : Inputs-output of controlled process for model 1 adjusted parameters	75
Figure 6.4 : Input-output values of controlled process for model 1($n=10, U=3$) ..	76
Figure 6.5 : Input-output val. of controlled process for model 1, $W_2=[0.2 \ 0.9]$...	77
Figure 6.6 : Input-output values of controlled process for model 1 with noise ...	78
Figure A.1 : Inputs-output of controlled process for model 2 at the beginning.....	90
Figure A.2 : Inputs-output of controlled process for model 2 adjusted parameters	91
Figure A.3 : Input-output values of controlled process for model 2($n=12, U=4$)...	92
Figure A.4 : Input-output values of controlled process for model 2, $W_2=[0.6 \ 0.4]$	93
Figure A.5 : Input-output values of controlled process for model 2 with noise ...	94
Figure A.6 : Inputs-outputs of controlled process for model 3 at the beginning...	95
Figure A.7 : Inputs-outputs of controlled process for model 3 adjusted parameters	96
Figure A.8 : Inputs-outputs of controlled process for model 3($n=13, U=6$)	97
Figure A.9 : Inputs-outputs of controlled proc. for model 3, $W_2=[0.03 \ 0.1]$	98
Figure A.10 : Inputs-outputs of controlled process for model 3 with noise	99

LIST OF SYMBOLS

$\text{CaO, SiO}_2, \text{Fe}_2\text{O}_3, \text{Al}_2\text{O}_3$	: Oxides (calcium, silicium, iron, aluminium)
$\text{C}_3\text{S, C}_2\text{S, C}_3\text{A, C}_4\text{AF}$	: Cement compounds
ML, MA, MS	: Module values (lime, aluminium, silica)
A, B, C, D	: Plant parameters matrices of time series model
$y(t)$	: Plant output
$\hat{y}(t)$	: Predicted output
$u(t)$	: Plant input
p, n_y	: Number of outputs
m, n_u	: Number of inputs
n_a, n_b, n_c, n_d	: Order of plant parameter matrices
$e(t)$	: Error (gaussian white noise)
$e^0(t)$	: Prediction error
t	: Time
d	: Time delay
q^{-1}, z^{-1}	: Backward shift operator
f_b	: Bandwith of the system
f_s	: Sampling frequency
t_r	: Rise time
T_s	: Sampling period
T	: Period
ω_0	: Natural resonance frequency
ω_j	: Angular frequency
φ_j	: Phase angle
f_{PRBS}	: Frequency of PRBS
N	: Number of cells in PRBS, number of data
R	: Test duration of PRBS
$\theta(t)$	: Parameter vector
$\hat{\theta}(t)$	: Estimated parameter vector
$\phi(t)$	: Observation vector
$V_{\text{Lsi}}(\hat{n}_i), CV_{\text{Lsi}}(\hat{n}_i)$	: Least square loss functions
$\hat{d} X(N)$	: increasing function of $\hat{n}_i$
$\hat{d}$	: Number of parameters of $\hat{d} X(N)$
$\text{BIC}_{\text{Lsi}}(\hat{n}_i)$	: Bayesian information function
$\hat{n}_i, \hat{n}_{ij}$	: Observability indices of multivariable model
$J(t)$	: Least square criterion
$P(t)$	: Adaptation gain matrix
λ_1, λ_2	: Forgetting factors
$H_{q+1,M}$	: Hankel matrix
M	: Matrix coefficients of Henkel
$h(i,k)$	: Row of Hankel matrix
$L(i)$	: Number of Minimality
$S(i)$	: Line blocs

F, G, H, D	: System parameter matrices of state space model
x(t)	: System state vector of state space model
V	: Nonsingular matrix
R(i)	: Autocorrelation
R N(i)	: Normalized autocorrelation
Γ_{ei}	: Autocorrelation function
τ	: Time constant
w	: Unmeasured disturbances
z	: Measurement noise
$\varepsilon(t)$	: Error
y^p	: Prediction vector
y_{sp}	: Set point
$\bar{y}_p$	: Noise free plant output
Ψ	: Objective function
A_i	: The step response coefficients matrix
E	: Predicted value of the process error
Δu	: Control moves
K_{MPC}	: Feedback gain matrix
m	: Model horizon
U	: Control horizon
n	: Prediction horizon
W₁	: Weighting matrix for predicted errors
W₂	: Weighting matrix for control moves

IDENTIFICATION AND MODEL PREDICTIVE CONTROL OF THE RAW MATERIAL BLENDING PROCESS IN CEMENT INDUSTRY

SUMMARY

Raw material blending process is the one of the main processes in a cement factory. The task of the raw material blending process in a cement factory is to mix the raw materials in order to produce cement raw meal for the kiln. The process is multivariable and coupled one, because the feeder tanks do not contain chemically homogeneous raw materials. The chemical compositions of the raw materials vary from time to time. Furthermore, each of the raw materials contains varying amount of the constituent oxides. The disturbances coming from the variations in compositions of the raw material from long-term average compositions cause the changes of the system parameters. Therefore, it is essential to construct multivariable stochastic dynamical model using a priori information of the process and characteristic of the disturbance to represent the process. Then, we need to design controller, which calculates optimal raw feed ratios to achieve reference oxide and module values of the raw meal.

Raw mill blending process in Nuh Cement Factory in Turkey has been dealt with in the thesis. Here, three different feedstreams are available and, after mixing on a conveyor belt, fed to the raw mill by weigh-feeders, before being thoroughly ground and mixed in the raw mill. In this study, two of feedstreams containing low grade and iron ore are have been controlled. A sample of this raw mix has been collected at the input of the raw mill grinder and analyzed every one min by PGNA, Thus, the measurements consist of the output concentrations of the four basic oxides.

The chemical composition control of the raw meal is necessary because the relative amounts of C_3S , C_2S , C_3A and C_4AF formed in the kiln strongly depend on the oxide composition of the ground mix. The purpose of the control is to maintain standard percentages or relative rates of these oxides. The most important four oxides are CaO , SiO_2 , Al_2O_3 and Fe_2O_3 . The relative rates used in Nuh Cement Factory in Turkey can be expressed by the so-called module values and these modules are lime module (ML), aluminum module (MA) and silica module (MS).

Linear multivariable stochastic time-series models in which, the inputs are the feed ratios of raw material components (low grade and iron ore) and the outputs are iron oxide or/and lime module, have been constructed in the identification section. Knowledge of process dynamics and characteristic of the disturbance, which is affecting the system, have been used for deriving a stochastic model of system. The data groups have been collected from the computer controlled system under varying operating conditions in Nuh Cement Factory. ARX models have been parameterized

by the canonical parameterization (Guidorzi). The Least Square Method has been used for determining the observability index ($\hat{n}_i$), which gives a minimal order of a state realization. The parameters in such an assumed model have been estimated with recursive equations, which are minimizing the sum of the squared errors. Whiteness test has been used for validation of the models. This test consists of verifying if the prediction error signal $e_i(t)$ can be considered as a white noise (zero mean, Gaussian).

At the end of identification, the predicted output $y_p(t)$ have resembled the measured output. The deviations of $y_p(t)$ from $y(t)$ are due to modeling errors. In the validation tests, the absolute value of the autocorrelation function of the error signal have been found smaller than the chosen zero threshold value 0.15. This means that the proposed models are reliable.

State space model equations representing the process are given by :

$$x(t+1) = F x(t) + G u(t) \quad (1)$$

$$Y(t) = H x(t) + D u(t) + z(k) \quad (2)$$

where F, G, H, and D are the system parameter matrices, u is the input vector representing raw mix proportions (low grade, iron ore), y is the output vector representing iron oxide and/or lime module values of the raw meal and z is the noise representing modeling errors.

In the control part, a multivariable model predictive controller, using discrete time step response model and dynamic matrix control algorithm have been designed to compute optimal raw feed ratios (low grade and iron ore) for three models to achieve the target value of the lime module and iron oxide values of the raw meal despite disturbances. State space models have been turned into the step response model for predictive controller to calculate control actions in the computer program (MATLAB).

The control law is based on minimizing the objective function. The objective function, Ψ , is the sum of the square of the errors between setpoint values and predicted values of iron oxide and lime module (n steps into the future) of the raw meal.

The purpose of the controller is to choose the control moves $\Delta u(t_i)$ for U moves into the future such that the objective function is minimized. Thus, model predictive controller deals with not only the quality control but quantity control as well. This is accomplished by minimizing quadratic objective function consisting of the energy term (Δu^2). The control law using dynamic matrix control algorithm is given by :

$$\Delta u = (A^T W_1^2 A + W_2^2)^{-1} A^T W_1^2 E = K_{MPC} E \quad (3)$$

where W_1 and W_2 are weighting matrices, K_{MPC} is called the feedback gain matrix and E is the vector of predicted future errors over the horizon n.

At the end of the control, optimal raw mix proportions have been found as $u_1(\text{low grade}) = 59.6(\%)$ and $u_2(\text{iron ore})=3.642(\%)$ for model 1, $u_1(\text{low grade}) = 60.14(\%)$ and $u_2(\text{iron ore}) = 6.57(\%)$ for model 2 and $u_1(\text{low grade}) = 57.54(\%)$ and $u_2(\text{iron ore}) = 5.97(\%)$ for model 3.

Consequently, identification and model predictive control of the cement raw material blending process in cement factory have been examined. Good parametric models having minimum order and minimum number of parameters have been obtained in the identification part. In the control part, we have obtained perfect control, which gives fast response and perfect tracking compensating the disturbances rapidly for each model. In other words, the efficient raw mix proportion and significant decrease in the variance of controlled outputs has been provided at the end of this study.



ÇİMENTO ENDÜSTRİSİNDE HAMMADDE HARMANLAMA PROSESİNİN TANILANMASI VE MODEL ÖNGÖRÜLÜ KONTROLÜ

ÖZET

Hammadde harmanlama prosesi, çimento fabrikasındaki ana proseslerden biridir ve görevi ise, çimentoyu oluşturan hammaddeyi karıştırarak harmanlanan karışımı döner fırın prosesi için hazırlamaktır. Proses çok giriş çıkışlıdır ve besleme tanklarındaki hammaddeler kimyasal olarak homojen bir yapı içermediği için etkileşimli bir yapıya sahiptir. Çimentoyu oluşturan hammaddelerin kimyasal kompozisyonları zamanla değişmektedir ve her bir hammadde değişik miktarlarda oksit bileşenleri içermektedir. Hammaddelerin kimyasal kompozisyonlarında uzun dönemli ortalama değerlerine göre meydana gelen sapmalar prosese bozucu olarak etkilemekte ve bu durum proses parametrelerinin değişmesine yol açmaktadır. Bu sebeple, prosesi temsil etmek üzere prosesin daha önceki giriş çıkış değerlerini ve bozucu karakteristiğini kullanan çok giriş ve çıkışlı rastgele bozuculara haiz dinamik bir modele ve karışımdaki oksit ve standart modül değerlerini referans değerlerine yaklaştırmak üzere en uygun hammadde besleme oranlarını hesaplayacak bir kontrolör tasarımına ihtiyaç duyulmaktadır.

Tezde, Nuh Çimento fabrikasındaki hammadde harmanlama prosesi ele alınmıştır. Burada üç ayrı hammadde besleme silosu mevcuttur. Hammaddeler konveyorde karıştırıldıktan sonra ağırlık besleyicileri vasıtası ile harmanlama değirmenine gönderilmektedir. Harmanlama değirmeninde ise bu karışım iyice öğütülüp tekrar karıştırılmaktadır. Üç besleme silosundan, düşükmal ve demir cevherini içeren ikisi kontrol için kullanılmaktadır. Karışıma ait örnekler harmanlama milinin girişinden alınmakta ve PGNAA hızlı analiz cihazı ile dakikada bir analiz edilmektedir. Böylece karışıma ait dört temel oksit yüzdesi ölçülmektedir.

Karışımın kimyasal kompozisyonunun kontrolü gereklidir, çünkü döner fırın prosesindeki C_3S , C_2S , C_3A ve C_4AF ' nin nispi değerleri büyük ölçüde öğütülen karışımın oksit kompozisyonuna bağlıdır. Bu sebeple kontrolün amacı, oksit değerlerini veya birbirlerine göre nispi oranlarını istenen referans değerlerde tutmaya çalışmaktır. Dört önemli oksit CaO , SiO_2 , Al_2O_3 ve Fe_2O_3 ' dir. Bu dört oksidin birbirine göre oranlamasından oluşan eşitliklere modül adı verilmektedir. Nuh Çimento fabrikasında kullanılan modüller ise, kireç modülü (ML), alüminyum modülü (MA) ve silika modülüdür (MS).

Tezin tanılama bölümünde, lineer, rastgele bozuculara haiz olan, çok giriş ve çıkışlı zaman serisi modelleri geliştirilmiştir. Girişler düşük mal ve demir cevheri ve çıkışlar ise demir oksit ve/veya kireç standartıdır. Geliştirilen bu modeller sistemin daha önceki giriş çıkış değerlerini ve bozucu karakteristiklerini kullanan modellerdir.

Tezde kullanılan giriş ve çıkış verileri Nuh Çimento Fabrikasında değişik çalışma koşulları altında çalışan bilgisayar kontrollü prosesten toplanmıştır. Geliştirilen ARX modeller Guidorzi' nin kanonik formunu kullanarak parametrize edilmiştir. Prosesin minimum mertebesi veren katsayı, en küçük kareler metodunu kullanan fonksiyon kullanılarak bulunmuştur. Sistem parametreleri hataların karelerinin toplamının minimumunu bulan ardışık eşitlikler çözülerek kestirilmiştir. Son olarak, geliştirilen modellerin değerlendirilmesi için beyazlık testi kullanılmıştır. Bu test, kestirim sonucu oluşan hata değerlerinin beyaz gürültü (0 ortalamalı, Gaussian dağılımlı) fonksiyonuna benzer bir dağılım gösterip göstermediğini kontrol etmektedir.

Tanımlama sonucunda hesaplanan çıkış değerleri, Nuh Çimento fabrikasından alınmış olan ve daha önceden ölçülmüş olan çıkış değerlerine oldukça benzemektedir. Aradaki farklar ise modelleme hatalarından kaynaklanmaktadır. Model değerlendirme testi sonucunda, hata sinyalinin normalleştirilmiş otokorelasyon fonksiyonunun mutlak değeri, seçilen sıfırıncı mertebe tutucunun değeri olan 0.15' den küçük bulunmuştur. Dolayısıyla güvenilir modeller elde edilmiştir.

Prosesi temsil eden durum uzay modeli eşitlikleri aşağıda verilmiştir :

$$x(t+1) = F x(t) + G u(t) \quad (1)$$

$$Y(t) = H x(t) + D u(t) + z(k) \quad (2)$$

Bu eşitlikte F, G, H, ve D sistem parametre matrisleri, u düşük mal ve demir cevherini temsil eden giriş vektörü, y harmanlanmış karışımın demir oksit ve/veya kireç standartı değerlerini temsil eden çıkış vektörü ve z ise modelleme hatalarını temsil eden gürültü fonksiyonudur.

Kontrol kısmında bozuculara karşı duyarsız çalışarak, harmanlanan karışımındaki istenen referans demiroksit ve kireç standartı değerlerine ulaşmak üzere en uygun düşük mal ve demir cevheri besleme oranları hesaplayacak, basamak cevabı modelini ve dinamik matris algoritmasını kullanan çok giriş ve çıkışlı, model öngörülü kontrolör tasarlanmıştır. Kontrol aşamasında, model öngörülü kontrolörün MATLAB de yazılmış bilgisayar programında kumanda değerlerini hesaplaması için, elde edilen durum uzay modelleri basamak cevabı modelleri haline dönüştürülmüştür.

Kontrol kanunu amaç fonksiyonunun minimumunun bulunmasına göre tasarlanmıştır. Amaç fonksiyonu Ψ , harmanlanmış karışımındaki demir oksit ve/veya kirecin, istenen referans değerleri ile kontrol sonucu n adım sonra hesaplanan değerleri arasında oluşan hatanın karelerinin toplamını veren bir fonksiyondur.

Kontrolör, amaç fonksiyonunun minimumunu bulacak şekilde gelecekteki U tane kumunda değerini hesaplamaktadır. Böylece kullanılan model öngörülü kontrolör sadece çimento kalitesini değil aynı zamanda hammadde miktarlarını da ele almaktadır. Bu da amaç fonksiyonundaki enerji terimi Δu^2 nin minimize edilmesiyle sağlanmaktadır. Dinamik matris kontrol algoritmasını kullanan kontrolör için kontrol kanunu aşağıdaki eşitlikte verilmiştir.

$$\Delta u = (A^T W_1^2 A + W_2^2)^{-1} A^T W_1^2 E = K_{MPC} E \quad (3)$$

Burada W_1 ve W_2 ağırlık matrisleri, K_{MPC} geri besleme matris kazancı ve E ise gelecekteki n adım için öngörülen hata vektörüdür.

Kontrol sonucunda, optimal hammadde besleme oranları model 1 için düşük mal %59.6 ve demir cevheri %3.642, model 2 için düşük mal %60.14 ve demir cevheri %6.57 ve model 3 için düşük mal %57.54 ve demir cevheri %5.97 olarak hesaplanmıştır.

Sonuç olarak, bu çalışmada çimento endüstrisinde hammadde harmanlama prosesinin tanılanması ve model öngörülü kontrolü incelenmiştir. Tanılama kısmında minimum parametre sayısına ve mertebeye sahip iyi parametric modeller geliştirilmiş ve kontrol kısmında ise, her bir model için rastgele bozuculara karşı duyarsız, referans değerleri iyi bir şekilde izleyen ve çabuk oturan sistem cevapları elde edilmiştir. Başka bir deyişle, kontrol edilen çıkışlardaki sapmaları önemli ölçüde azaltan en uygun hammadde besleme oranları elde edilmiştir.



1. INTRODUCTION

The cement industry has a vital importance in the world from many points of view. The cement can be seen almost everywhere in the everyday life and the industrial society could not be thought without it. This industry is one of the biggest power consumers among many countries. The size of cement production and consumption has been evaluated for some time to characterize the development of a country. The technology of the cement production has already been known for about 100 years. Only the machinery and the technological parameters (capacity, effectiveness, quality) have drastically changed since that time.

The task of the raw material blending process in a cement factory is to mix the raw materials in order to produce cement raw meal for the kiln. Oxide compositions of the raw meal determine significantly the high quality of cement with specified strength characteristics. On the other hand, the chemical compositions of the raw materials vary from time to time and they are not measured. Furthermore, each of the raw materials contains varying amount of the constituent oxides. The disturbances coming from the variations in compositions of the raw material from long-term average compositions cause the changes of the system parameters. Therefore, it is essential to construct experimental dynamical model using a priori information of the process to represent the raw meal blending process having stochastic disturbances. Then, we need to design controller, which calculates optimal raw feed ratios to achieve desired control performance.

A raw meal with a good fineness and well-controlled chemical composition can improve the cement quality and the kiln operation performance. On the other hand, good raw materials used for correction of the composition, such as bauxite, are very expensive. The approaches to the solution of this fundamental blending problem have varied widely. Before the advent of more sophisticated techniques such as X-ray analysis and prompt gamma neutron activation analysis (PGNAA) blending was

done in several stages with intermediate wet chemical analysis used to determine the proportion of each batch to be blended. High-energy costs have in recent years made the wet method of cement manufacturing less profitable and most of the plants are presently being designed for the dry process. This old method was a wet process involving large homogenization basins and hence incurred significant expense in compressors and pumps. An alternative is to use a control strategy for determining the proper feed ratios of raw material components.

As digital computers and fast chemical analysis equipment were introduced into the cement industry, several control systems of varying complexity were developed for direct digital control of the blending process [1], [2], [3] and [4]. The theory of multivariable adaptive control was also applied to raw material blending [5-6]. Stochastic modeling, which is using experimental process data and characteristics of the various disturbances, and self-tuning control of a continuous cement raw material mixing system, were presented by [5]. In this study, the optimal control strategy was obtained as a minimum variance strategy. The control problem was solved using a self-tuning minimum variance regulator, resulting in a %34 decrease in the variance of the controlled variable. Adaptive composition control of cement raw mill guaranteeing a robust operation was handled by [6]. It consists of two parts: an estimator and a feedback regulator. The controller system tries to eliminate the disturbances and guide the process in such a way that the mill-silo output would be near to the required reference values of module. A time varying Kalman filter was developed by [7] whose outputs are the optimal estimates of flowstream concentrations. Real-time optimization problem for the raw materials was solved by [8]. A two level adaptive control policy combined with a heuristic auxiliary system was proposed for robustness of the raw mix control system by [9]. The singular value decomposition algorithm (SVD), which is the one of the most basic and important tools in the analysis and solution of the problems in numerical linear algebra for calculation of the raw mix proportion, was introduced by [10]. The SVD technique provides a reliable determination of the rank of matrix, thereby leading to accurate solutions of linear equations. A new generic optimal controller structure, which is equivalent to those used at internal model principal or pole placement technique, was dealt with by [11]. Recently, Constrained self-tuning composition control algorithm was presented for MIMO system by [12].

In this study, the identification and model predictive control of the cement raw material blending system in cement factory have been examined. The data groups have been collected from the computer controlled system under varying operating conditions in Nuh Cement Factory. The system has no prehomogenization of the raw materials. Instead, efficient mixing and continuously operating homogenization silos for raw meal have been obtained by using control strategy. In the identification part, three different linear multivariable stochastic time-series models in which, the inputs are the feed ratios of raw material components (low grade and iron ore) and the outputs are iron oxide or/and lime module of the raw meal, have been constructed. Knowledge of process dynamics and characteristic of the disturbance, which is affecting the system, have been used for deriving a stochastic model of system. ARX model has been parameterized by the canonical parameterization (Guidorzi). The Least Square Method has been used for determining the observability index ($\hat{n}_i$), which gives a minimal order of a state realization. The parameters (raw material oxide concentrations) in such an assumed model have been estimated using recursive equations, which are minimizing the sum of the squared errors. Whiteness test has been used for validation of the models. This test consists of verifying if the prediction error signal $e_i(t)$ can be considered as a white noise (zero mean, Gaussian distribution).

In the control part, a multivariable unconstrained model predictive controller, using discrete time step response model and dynamic matrix control algorithm have been designed to compute optimal raw feed ratios for three models to reach closer standard lime module and/or iron oxide percentage values of the raw meal and to decrease the variation of the oxide compositions of the raw meal despite disturbances.

2. RAW MATERIAL BLENDING SYSTEM IN CEMENT INDUSTRY

The cement manufacturing process consists of especially of mining, crushing and grinding, burning, and grinding with gypsum. Two basic processes, the wet process and the dry process, are used for cement manufacturing. The dry cement manufacturing process consumes less energy than the wet process, but in the dry process it is more difficult to produce a cement raw meal of high quality. In the wet process, proper proportions of the raw materials are mixed with enough water to form a paste called slurry. In this form the raw materials are further proportioned, mixed, ground and pulverized, and then pumped into a rotary, inclined furnace, called a kiln. The dry process is similar, except that the raw materials are proportioned, stored, ground, mixed, pulverized, and fed into the kiln in a dry state. Inside the kiln the raw mix will undergo a sequence of reactions [13,14]. Sintering takes place at the final stage of the reaction, i.e., at 1400-1450 °C, and a substance called clinker, having its own physical and chemical properties, is formed. The clinker is cooled, crushed, and mixed with a predetermined percentage of gypsum to regulate the setting time of cement. Finally, the finished product, known as the Portland cement, is stored in large storage bins called silos, from which it is fed to an automatic packing machine.

2.1 Raw Material Blending Process

A simplified schematic diagram of the raw mill blending process in Nuh Cement Factory in Turkey is shown in figure (2.1). Although model predictive control algorithm has been presented in the thesis could be adapted to other rather different cement plants and even to non-cement applications in materials blending, the specifics of the particular plant (Nuh Cement Factory in Turkey) used to test the algorithm are utilized in the development of the model.

Here, three different feedstreams are available and, after mixing on a conveyor belt, fed to the raw mill by weigh-feeders, before being thoroughly ground and mixed in the raw mill. In this study, two of feedstreams containing low grade and iron ore are have been controlled. A sample of this raw mix has been collected at the input of the raw mill grinder by an auto sampler and analyzed every one minute by PGNAA, then sent to the computer through a data communication line. Thus, the measurements consist of the output concentrations of the four basic oxides. The raw meal has been transferred to homogenization basins where further continuous mixing decreases the magnitude of the concentration variations about the silo average values. The complete filling of a basin requires a unit batch time of 16h..

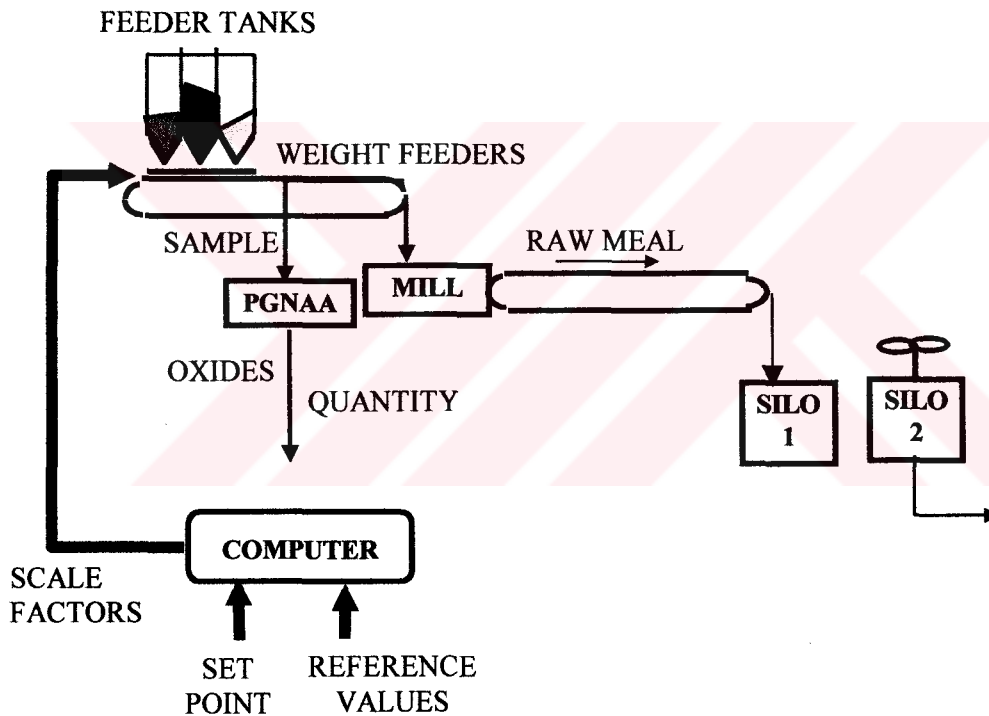


Figure 2.1. Simple schema of blending control in Nuh Cement Factory.

Nominal flow rates, concentrations, costs, and common names of the feedstreams are given in Table (2.1) [12,15]. Clearly, the large variations in the costs are a major incentive to calculate precise feedstream compositions using the estimation algorithm.

Table 2.1. Blending feedstream data.

Nominal Concentration (%)						
Name	Nominal Flow (%)	CaO	SiO ₂	Fe ₂ O ₃	Al ₂ O ₃	Cost (USD/Ton)
Low-Grade Material (Clay + Marn)	50.51	40.55	19.03	1.32	2.58	0.22
High-Grade Material (Lime+Marn)	38.31	48.15	7.59	1.58	1.59	1.54
Iron (Ore)	11.18	2.26	13.85	61.91	10.34	7.21

2.2 Structure of the Cement Manufacturing Process and Chemical Equations

The cement manufacturing process is composed of two main parts: (1) blending and grinding of raw materials, and (2) kiln firing and finish grinding to a fine powder containing four compounds in the following approximate weight fractions: tri-calcium silicat, C_3S ($\approx 60\%$), di-calcium silicat, C_2S ($\approx 25\%$), tri-calcium aluminate, C_3A ($\approx 5\%$), and tetra-calcium aluminoferrite, C_4AF ($\approx 10\%$). The concentrations of these four cement compounds, in weight percent (wt %) or expressed as a decimal fraction, are related to the concentrations of the four ignited oxides after firing (CaO, SiO₂, Fe₂O₃, and Al₂O₃) by Bogue equation [1].

$$\begin{bmatrix} C_3S \\ C_2S \\ C_3A \\ C_4AF \end{bmatrix} = \begin{bmatrix} 4.07 & -7.6 & -1.43 & -6.72 \\ -3.07 & 8.60 & 1.08 & 5.07 \\ 0.0 & 0.0 & -1.69 & 2.65 \\ 0.0 & 0.0 & 3.04 & 0.0 \end{bmatrix} \begin{bmatrix} CaO \\ SiO_2 \\ Fe_2O_3 \\ Al_2O_3 \end{bmatrix} \quad (2.1)$$

The final strength and quality of cement are strong functions of the concentrations of the cement compounds. Examination of equation (1) reveals the high sensitivity of the cement compounds to ignited oxide variations. Any composition restrictions imposed on the cement compounds result in even more stringent restrictions on ignited oxide compositions. Thus the precise regulation of the output concentrations of the raw mill is absolutely essential in the manufacture of high quality cement

with specified strength characteristics. The constancy of the kiln feed composition is of great importance not only from the standpoint of product quality control and the magnifying effects of the kiln reactions, but also because constant kiln feed composition allows better kiln control and permits the build up of a stable coating on the inside of the kiln, protection the refractory lining.

In the most general cement blending operation, raw material feedstreams from both on-site quarries and outside sources are mixed together using weigh feeder conveyor belts and are fed to a grinding mill. The ability of the quarry to provide materials of uniform composition is insufficient to the extent that time varying control of the weigh feeder flow rates is necessary. Plant data indicate that oxide concentrations vary randomly in time in the short (~1h) the long (~1yr) terms.

Control of the raw mix is one of the most crucial parts of the cement manufacturing process. In addition, most plants require some form of homogenizing silo for kiln feed. Such silos, as well as being energy intensive, frequently do not provide the degree of blending that is expected.

The mixing system has previously been discussed [16-18]. The chemical composition control of the raw meal is necessary because the relative amounts of C_3S , C_2S , C_3A and C_4AF formed in the kiln strongly depend on the oxide composition of the ground mix. Denote the most important four oxides CaO , SiO_2 , Al_2O_3 and Fe_2O_3 by C , S , A and F , respectively. The purpose of the control is to maintain standard percentages or relative rates of these oxides. The relative rates used in Nuh Cement Factory in Turkey can be expressed by the so-called module values:

Lime standard (or module):

$$ML = \frac{100 C}{2.8S + 1.1A + 0.8F} \quad (2.2)$$

Aluminum module:

$$MA = \frac{A}{F} \quad (2.3)$$

Silica module:

$$MS = \frac{S}{A + F} \quad (2.4)$$

A high ML requires high heat consumption for clinker burning inside the kiln, and thus gives more strength to the cement. A higher MS decreases the liquid phase content, which impairs the burnability of the clinker and reduces the cement setting time. The value of MA determines the composition of liquid phase in clinker.

The goal is to achieve a desired level of ML, MS, and MA of the raw mix, to produce a particular quality of the cement by controlling the mix proportions of the raw materials. To achieve an appropriate raw mix proportion is very difficult, due to the inconsistencies in the chemical composition of the raw material.

1.3 Analysis Systems of the Raw Material Blending Process

One of the major problems in designing raw materials blending systems is the difficulty of obtaining frequent rapid accurate representative chemical analysis of the raw materials. Most cement plants today use X-ray fluorescence (XRF) analyzers for frequent batch analysis of the kiln feed. However, the time delay between obtaining the sample and making any correction to the raw mill inputs can be several hours. Moreover, XRF analyzers require a high degree of preliminary sampling and are ineffective for frequent raw materials analysis.

A relatively new method of bulk raw materials analysis is available which may help deal with some of these problems. It is known as prompt gamma neutron activation analyzers (PGNAA). PGNAA analyzers work by bombarding materials, usually in solid or liquid form, with neutrons. Some atoms absorb neutrons and promptly emit high-energy gamma radiation, which can be detected and analyzed by energy dispersive methods. The sensitivity of PGNAA for the elements calcium, silicon, iron, and aluminum, which are of primary importance in cement manufacturing, is great enough that those elements can be rapidly detected, allowing control actions to be taken within only a few minutes. By use of appropriate calibration and

correction procedures and control algorithms, a computer can be used to store and analyze a sample's gamma spectrum, compute the elemental composition and oxide module, and immediately take the control actions necessary to minimize deviation from a specified set point. While similar claims have been made for the most sophisticated automatic on-line X-ray fluorescence analyzers, the latter have yet to prove themselves either reliable or cost effective in cement plants. Batch sample XRF analyzers are excellent analytical tools, but the sampling, sample preparation, and required analysis times make their use in process control of limited effectiveness. Some features of PGNAAs are compared with those of batch XRF analysis in [19]. Results of simulations of the system configurations are also presented in [19]. Best system performances are achieved by employing two analyzers in parallel on two raw material streams prior to mill, placement of a PNGAA analyzer upstream of the mill and a PGNAAs analyzer downstream from the mill, respectively. By using PGNAAs analyzers, feedback, feed forward, or predictive control strategies can be effectively employed in raw materials proportioning. Such strategies can potentially supplant the expensive and problematic raw materials sampling systems commonly employed in cement plants, and, in many cases, eliminate the need for raw materials preblending altogether. For existing cement plants experiencing difficulty in producing acceptable kiln feeds, as well as for plants still on the drawing board, this advance represents possibly large savings in capital and operating costs.

3. SYSTEM IDENTIFICATION

3.1 Introduction

System Identification has its roots in standard statistical techniques and many of the basic routines have direct interpretations as well known statistical methods such as Least Squares and Maximum Likelihood. The control community took an active part in the development and application of these basic techniques to dynamic systems right after the birth of “modern control theory” in the early 1960’s. Maximum likelihood estimation was applied to difference equations (ARMAX models) by Aström and Bohlin [20] and thereafter a wide range of estimation techniques and model parameterizations flourished. By now, the area is well matured with established and well-understood techniques. Industrial use and application of the techniques has become standard by Ljung [21].

The literature on System Identification is extensive. For a practical user oriented introduction we may mention Ljung and Glad [22]. Texts that go deeper into the theory and algorithms include Eykhoff [23], Goodwin and Payne [24], Ljung [25], and Söderström and Stoica [26]. A classical treatment is Box and Jenkins [27]. In addition, there is a substantial literature on other approaches, such as “set membership” (compute all those models that reproduce the observed data within a certain given error bound), estimation of models from given frequency response measurement Schoukens and Pintelon [28], on-line model estimation Ljung and Söderström [29], non-parametric frequency domain methods Brillinger [30], etc. To follow the development in the field, the IFAC series of Symposia on System Identification (Budapest, 1991, Copenhagen, 1994) is a good source. The techniques apply to very general models. Most common models are difference equations descriptions, such as ARX and ARMAX models, as well as all types of linear state-space models. Lately, black-box nonlinear structures, such as Artificial Neural Networks, Fuzzy models and so on, have been much used.

According to the classical definition, identification is the determination, on the basis of input and output, of a system within a specified class of systems, to which the system under test is equivalent. The process of system identification includes constructing models and estimating unknown plant parameters from experimental data. This definition is quite general and allows many degrees of freedom in the practical formulation of the identification problem. For example, one may select a linear or a non-linear model. It is possible to have different parameterizations for a given model. Additional choices can be made to select the input signal.

There are two major classes of mathematical models, which are used for identification: linear and non-linear models. Due to the complexity inherent in non-linear systems and to the fact that there is no well-established general theory for such systems, it is assumed that the physical process being considered behaves in a linear fashion at least within a small neighborhood of a nominal operating point.

Parameterization is an important issue for multi-input multi-output (MIMO) systems, since they admit more than one parameterization depending on the observability indices of these systems. Thus, in order to obtain a minimal parameterization for MIMO systems, the structure of the system, i.e., the observability indices related to each output must be determined [31-40].

The selection of an appropriate input signal is an important step in identification problems [41-44]. A basic criterion for this selection is that the input-output data should be informative enough to discriminate between different models among the class of models being considered. Without this discrimination there is no guarantee that the obtained parameters are the true parameters of the system. This criterion can be expressed mathematically in terms of the system being identified. Uncorrelated pseudo random binary sequences are examples of suitable input signals that can be used for designing an informative experiment.

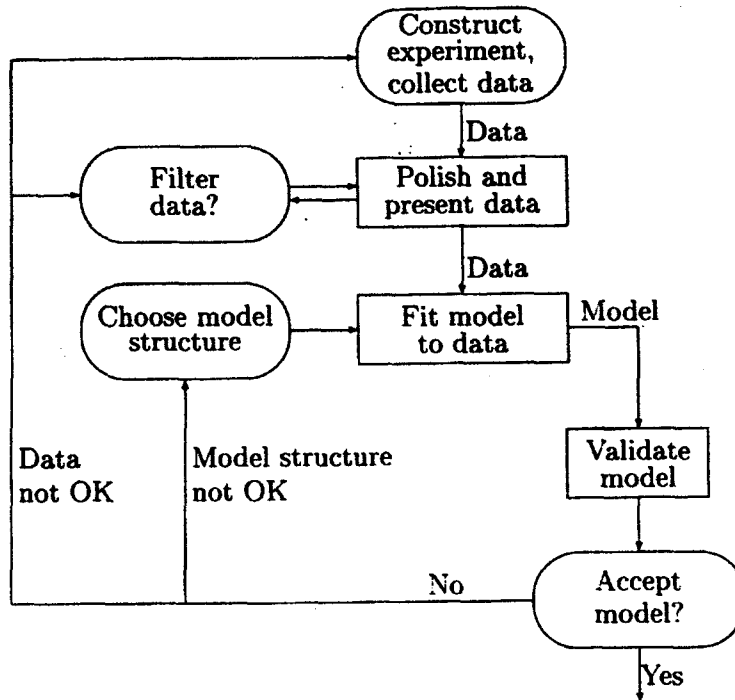


Figure 3.1. The system identification loop.

3.2 System Identification Principles

System identification is the process of determining a dynamical model for a plant starting from input/output measurement. This model is multivariable when it clarifies the variations between several outputs (or measurements) from variations of several inputs (or commands).

The knowledge of the dynamic of the plant to be controlled is the crucial step for designing and tuning a suitable controller, achieving good performances. System identification from input/output data is the most efficient way to obtain such plant models.

The notion of the mathematical model of a system or phenomenon is a fundamental concept. In general, a multitude of model types exist, each one dedicated to a particular application. For example, the “knowledge” type models (based on the laws of physics, chemistry, etc.) permit a fairly complete system description and are used for plant simulation and design. These models are in general extremely complex and can only rarely be directly used for the design of control systems. The dynamic

control models which give the relation between the input and output variations of a system are, as indicated above, the type of model suitable for the design and tuning of control systems. Although indications concerning the structure of these control models can be obtained from the structure of the knowledge type model, it is in general very difficult to determine the significant parameter values from these models.

This is why in the majority of practical situations; it is necessary to implement a methodology for direct identification of these dynamic (control) models from experimental data.

There are two types of dynamic models:

- 1) non-parametric models (e.g.: frequency response, step response).
- 2) parametric models (e.g.: transfer function, differential or difference equation).

Henceforward, identification of sampled discrete-time parametric dynamic models are the most suitable for the design, and tuning of digital control systems.

System identification is an experimental approach for determining the dynamic model of a system. It includes four steps:

- 1) Input/output data acquisition under an experimentation protocol.
- 2) Choice or estimation of “model” complexity.
- 3) Estimation of the model parameters.
- 4) Validation of the identified model (structure and values of parameters)

A complete identification operation must necessarily comprise the four stages indicated above. The specific methods used at each stage depend on the type of model desired (parametric or non-parametric, continuous-time or discrete-time).

The availability of a digital computer permits the implementation of algorithms, which automatically estimate the parameters of the sampled system models. It should be stressed that the identification of the parametric sampled models enables (by simulation) non-parametric models of the “step response” or “frequency response”

type to be obtained, with a far higher degree of accuracy than when a direct approach is carried out. In addition, this is obtained using extremely weak excitation signals.

The identification of parametric sampled models leads to models of very general use and offers numerous advantages over other approaches. High performance identification algorithms having a recursive formulation tailored to real-time identification problems and to their implementation on microcomputer have been developed. The fact that these identification methods can operate with extremely weak excitation signals is a quality very much appreciated in practical situations.

DISCRETIZED PLANT

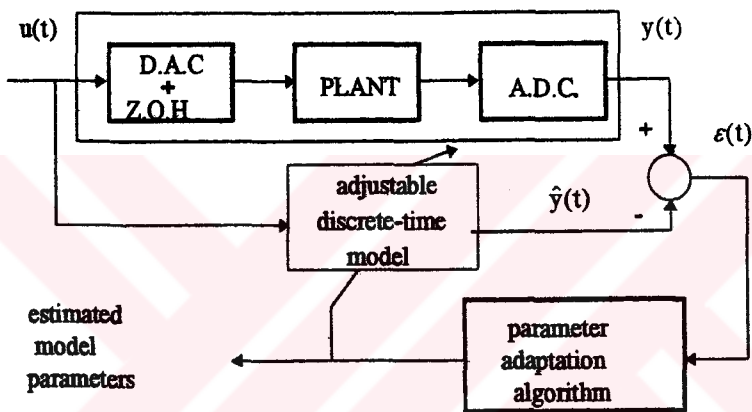


Figure 3.2. The parameter estimation principle for sampled model.

A discrete-time model with adjustable parameters is implemented on the computer. The error between the system output $y(t)$ at instant t and the output predicted by the model $\hat{y}(t)$ (known as the prediction error $e(t)$) is used by a parameter adaptation algorithm which at each sampling instant, will modify the model parameters in order to minimize this error.

The input is in general a very low level pseudo-random binary sequence generated by the computer (sequence of rectangular pulses with randomly variable duration). Once the model is obtained, an objective validation can be made by carrying out statistical tests on the prediction error $e(t)$ and the predicted output $\hat{y}(t)$. The validation test enables the best model to be chosen (for a given plant), i.e. the best structure and the best algorithm for the estimation of the parameters.

Finally, computing and graphically representing the step responses and the frequency responses of the identified model, the characteristics of the continuous-time model (step response or frequency response) can be traced.

This modern approach to system model identification eliminates all the faults of the “classical” methods and also offers other possibilities such as:

- Tracking of the variations of the system parameters in real time permitting retuning of the controllers during operation.
- Identification of the disturbance models.
- Modeling of the “transducer” noises in view of their elimination.
- Detection and measurement of the vibration frequencies.
- Spectral analysis of the signals.

One of the key elements for implementing this system model identification approach is the parameter adaptation algorithm (PAA) drives the parameters of the adjustable prediction model from the data available on the system at each sampling instant. This algorithm has a “recursive” structure, i.e. the new value of the estimated parameters is equal to the previous value plus a correction term which will depend on the most recent measurements.

In general a “parameter vector” is defined, the components of which are the different parameters, which must be identified. All the parameter adaptation algorithms have the following recursive structure [44]:

$$\begin{bmatrix} \text{New parameters} \\ \text{estimation} \\ \text{(vector)} \end{bmatrix} = \begin{bmatrix} \text{Previous parameters} \\ \text{estimation} \\ \text{(vector)} \end{bmatrix} + \begin{bmatrix} \text{Adaptation} \\ \text{gain} \\ \text{(matrix)} \end{bmatrix} \times \begin{bmatrix} \text{Measurement} \\ \text{function} \\ \text{(vector)} \end{bmatrix} \times \begin{bmatrix} \text{Prediction error} \\ \text{function} \\ \text{(scalar)} \end{bmatrix}$$

The measurement function vector is also known as the “observation vector”. Note that non-recursive parametric identification algorithms also exist (which process in block the input/output data files obtained over a certain time horizon).

Recursive identification offers the following advantages over these non-recursive techniques:

- Obtaining of an estimation for the model as the system evolves.
- Considerable data compression, since the recursive algorithms process at each instant only one input/output pair instead of the whole input/output data set.
- Requires a considerably weaker memory and computation power.
- Easy implementation on microcomputers.
- Possibility to implement real-time identification systems.
- Possibility to track the parameters of time variable systems.

The discretized plant in the absence of disturbances is described by the backward shift operator $[z^{-1} u(t) = u(t-1)]$:

$$y(t) = \frac{z^{-d} B(z^{-1})}{A(z^{-1})} u(t) \quad (3.1)$$

where

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{na} z^{-na} \quad (3.2)$$

$$B(z^{-1}) = b_1 z^{-1} + \dots + b_{nb} z^{-nb} \quad (3.3)$$

Equation (3.1) can be written in the form:

$$y(t+1) = - \sum_{i=1}^{na} a_i y(t+1-i) + \sum_{i=1}^{nb} b_i u(t-d-i+1) = \theta^T \phi(t) \quad (3.4)$$

in which the vector of parameters and measurements :

$$\theta^T = [a_1, \dots, a_{na}, b_1, \dots, b_{nb}] \quad (3.5)$$

$$\phi^T = [-y(t) \dots -y(t - n_a + 1), u(t - d), \dots u(t - d - n_b + 1)] \quad (3.6)$$

In the case of multi input/multi output model, the research of the model is a fundamental step. Indeed, the only possible knowledge of the number of system states is not enough, it is necessary to know more, by the mean of the observability indices, which are a set of invariants attached with each system outputs, whom the sum of which gives a minimal dimension of a state realization.

3.2.1 Selection of the sampling frequency

The sampling frequency for digital control systems is chosen according to the bandwidth of the system to be identified or according to the desired bandwidth for the closed loop system (they are in general of the same order of magnitude). No matter how the dynamic characteristics of the plant or the performances of the desired closed loop system are specified, they can always be related to the bandwidth of the system.

The basic rule for the choice of the sampling frequency is:

$$f_s = (6 \text{ to } 25) * f_b \quad (3.7)$$

where f_s is the sampling frequency and f_b is the bandwidth of the system.

This rule can be translated also in the time domain. In the time domain the basic rule is that on the step response (for damped systems) one should have more than 3 samples and less than 12 on the rise time t_r , i.e. the sampling period T_s is given by:

$$T_s = t_r / (3 \text{ to } 12) \quad (3.8)$$

More specifically for first order continuous time systems characterized by a time constant T , the sampling period is selected as:

$$T/4 < T_s < T$$

For second order continuous system characterized by a natural resonance frequency ω_0 (in rad/s) and a damping factor between 0.7 and 1, the sampling period should

satisfy the relation:

$$0.25 < \omega_0 * T_s < 1.5$$

Note that use of too high sampling frequencies can cause serious numerical troubles since all the poles and zeros are clustered in the z-plane around the point (1, j0).

3.2.2 Some commonly used input signals for system identification

The input signal used in an identification experiment can have a significant influence on the resulting parameter estimates. The input spectrum must satisfy certain properties in order to guarantee that the system can be identified. With a few exceptions we usually deal with linear models only. It will then be sufficient to characterize signals in terms of first- and second-order moments (mean value and covariance function). Note however, that two signals can have equal mean and covariance function but still have drastically different realizations. This will lead to the concept of persistent excitation.

There follow some examples of typical input signals.

- Step function.
- Pseudorandom binary sequence.
- Autoregressive moving average process (white noise).
- Sum of sinusoids.

A step function :

A step function is given by:

$$u(t) = \begin{cases} 0 & t < 0 \\ u_0 & t \geq 0 \end{cases}$$

The user has to choose only the amplitude u_0 . For systems with a large signal-to-noise ratio, an input step can give valuable information about the dynamics. Rise

time, overshoot and static gain are directly related to the step response. Also the major time constants and a possible resonance can be at least crudely estimated from a step response.

A sum of sinusoids :

In this class of input signals $u(t)$ is given by:

$$u(t) = \sum_{j=1}^m a_j \sin(\omega_j t + \phi_j) \quad (3.9)$$

where the angular frequencies $\{\omega_j\}$ are distinct,

$$0 \leq \omega_1 < \omega_2 < \dots < \omega_m \leq \pi$$

For a sum of sinusoids the user has to choose the amplitudes $\{a_j\}$, the frequencies $\{\omega_j\}$ and the phases $\{\phi_j\}$. A term with $\omega_1 = 0$ corresponds to $a_1 \sin \phi_1$. A term with $\omega_m = \pi$ will oscillate with a period of two sampling intervals.

An autoregressive moving average sequence:

There are many ways of generating pseudorandom numbers on a computer [41,42]. Let $\{e(t)\}$ be a pseudo-random sequence which is similar to white noise in the sense that

$$\frac{1}{N} \sum_{t=1}^N e(t)e(t+\tau) \rightarrow 0 \quad \text{as } N \rightarrow \infty \quad (\tau \neq 0)$$

This relation is to hold for τ at least as large as the dominating time constant of the unknown system. From the sequence $\{e(t)\}$ a rather general input $u(t)$ can be obtained by linear filtering as follows:

$$u(t) + c_1 u(t-1) + \dots + c_m u(t-m) = e(t) + d_1 e(t-1) + \dots + d_m e(t-m) \quad (3.10)$$

Signals such as $u(t)$ are often called ARMA (autoregressive moving average) processes. When all $c_i = 0$ it is called an MA (moving average) process, while for an

AR (autoregressive) process all $d_i = 0$. Occasionally the notation ARMA (m_1, m_2) is used, where m_1 and m_2 denote the number of c_i - and d_i - coefficients, respectively.

With this approach the user has to select the filter parameters m , $\{c_i\}$, $\{d_i\}$ and the random generator for $e(t)$. The latter includes the distribution of $e(t)$ which often is taken as Gaussian or rectangular, but other choices are possible.

The filtering can be written as:

$$C(q^{-1})u(t) = D(q^{-1})e(t) \quad (3.11)$$

where q^{-1} is the backward shift operator ($q^{-1} e(t) = e(t-1)$, etc.)

$$C(q^{-1}) = 1 + c_1 q^{-1} + \dots + c_m q^{-m} \quad (3.12)$$

$$D(q^{-1}) = 1 + d_1 q^{-1} + \dots + d_l q^{-l} \quad (3.13)$$

The filter parameters should be chosen so that the polynomials $C(z)$ and $D(z)$ have all zeros outside the unit circle. The requirement on $C(z)$ guarantees that $u(t)$ is a stationary signal. It follows from spectral factorization theory that the requirement on $D(z)$ does not impose any constraints. There will always exist an equivalent representation such that $D(z)$ has all zeros on or outside the unit circle, as long as only the spectral density of the signal is being considered. The above requirement on $D(z)$ will be most useful in a somewhat different context, when deriving optimal predictors. Different choices of the filter parameters $\{c_i, d_i\}$ lead to input signals with various frequency contents and various shapes of time realizations.

A pseudorandom binary sequence:

A pseudorandom binary sequence (PRBS) is a signal that shifts between two levels in a certain fashion. It can be generated by using shift registers for realizing a finite state system [43,23] and is a periodic signal. In most cases the period is chosen to be of the same order as the number of samples in the experiment, or larger. When applying a PRBS, the user must select the two levels, the period and the clock period. The clock period is the minimal number of sampling intervals after which the sequence is allowed to shift.

The convergence towards zero of the plant model error does not necessarily imply in every case that the estimated model parameters will converge towards the true parameters of the plant model. In order to correctly identify the parameters of the plant model, the input applied to the plant should contain enough components of distinct frequencies. More specifically if you have to identify a model with n parameters ($n = n_a + n_b$), the input signal should contain at least $(n+1)/2$ distinct sinusoidal components. In other words, good parameter identification requires the application of a frequency “rich” input. The standard solution in practice is provided by the use of “pseudo-random binary sequences” (PRBS), which approach the characteristics of a white noise and as such have a very rich frequency content.

Pseudo-random binary sequences are sequences of rectangular pulses, modulated in width, which approximate a discrete-time white noise (spectral density constant up to $0.45 f_s$) and thus have a spectral content “rich” in frequencies. One of the major advantages of using this standard signal is the fact that the magnitude of the signal is constant and can be selected independently of the frequency content of the signal, which is defined by the pulse width modulation.

The name “pseudo-random” comes from the fact that a PRBS is characterized by a “sequence length” within which the variations in pulse width vary randomly, but that over a large time horizon, it is periodic; the period being defined by the length of the sequence.

The PRBS are generated by means of shift registers with feedback (implemented in hardware or software). The maximum length of a sequence is $2^N - 1$ in which N is the number of cells of the shift register. The next figure presents the generation of a PRBS of length $31 = 2^5 - 1$ obtained by means of a 5 cells shift register.

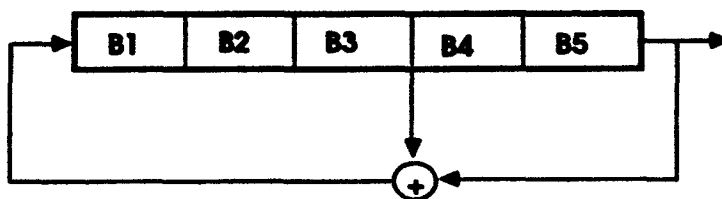


Figure 3.3. Generation of a PRBS of length $2^5 - 1 = 31$ sampling periods.

Note also a very important characteristic element of the PRBS: the maximum duration of a PRBS impulse is equal to N (number of cells) multiplied by the sampling period T_s . This property is to be considered when choosing a PRBS for system identification.

In order to correctly identify the steady state gain of the plant, the duration of at least one of the pulses must be greater than the risetime t_r of the plant. The maximum duration of a pulse being $N.T_s$, the following condition results:

$$N.T_s > t_r$$

which is illustrated in the next figure.

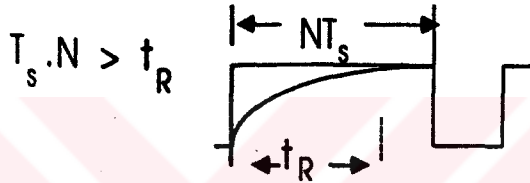


Figure 3.4. Choice of the maximum duration of a pulse in a PRBS.

From the above condition, you may determine N and therefore the length of the sequence, which is $2^N - 1$.

Furthermore, in order to cover the entire frequency spectrum generated by a particular PRBS, the length of a test must be at least equal to the length of the sequence.

In a large number of cases, the duration of the test (Z) is chosen equal to the length of the sequence. If the duration of the test is specified, you may select N and the clock frequency of the PRBS generator such as:

$$(2^N - 1).T_s < Z \quad (Z = \text{test duration})$$

Note that the condition for the selection of N can result in fairly large values of Z corresponding to sequence lengths of prohibitive duration.

This is why in a large number of practical situations, a submultiples of the sampling frequency is chosen as the clock frequency for the PRBS,

$$f_{PRBS} = f_s / p \quad p = 1, 2, 3, \dots$$

then the condition for the selection of N becomes:

$$p \cdot N \cdot T_s > t_r$$

Although the magnitude of the P.R.B.S. may be very low, the resulting output must be larger than the residual noise level. If the signal/noise ratio is too low, the length of the test must be augmented in order to obtain a satisfactory parameter estimation.

Typical values of the PRBS level are between 1 to 5% of the steady state value of the plant input.

Note that in a large number of applications, the significant increase in the PRBS level may be undesirable in view of the non-linear character of the plants to be identified (we are concerned with the identification of a linear model around an operating point).

3.2.3 Signal conditioning

Signal conditioning is an unavoidable step once the data acquisition of input/output data has been done in order to prepare the files for plant model identification.

The structures of the models used for identification correspond to dynamic models (which express output variations as a function of input variations around an operating point). It is therefore necessary, for a correct identification, to eliminate the DC components (corresponding to the operating point) from the input-output data or the slow drift of the operating point during identification.

Elimination of stationary DC components is carried out in two stages:

- a) Computation of the mean value (M.V.) of the I/O files.
- b) The mean value of the I/O files is subtracted from the I/O data and a new I/O file is created.

$$y'(t) = y(t) - \text{M.V. } y \quad (3.14)$$

$$u'(t) = u(t) - M.V. u \quad (3.15)$$

3.3 Effects of Random Disturbances Upon Parameter Identification

The measured output of the plant is in general contaminated by noise. This is due either to the effect of the random disturbances acting at different points of the plant, or to measurement noises. These random disturbances are frequently modeled by ARMA model, ARMAX (Auto Regressive Moving Average with Exogenous Input) model or ARX model.

These disturbances introduce errors into the identification of the plant model parameters when the recursive, (or non recursive) least squares algorithm is used. This type of error is called the “bias” of parameters. To eliminate this bias, predictors and adaptation errors must be chosen, such that:

$$E \{e(t) e(t-1)\} = 0 \text{ for } \hat{\theta} = 0 \quad (3.16)$$

The criteria for the generation of algorithms satisfying condition and asymptotically resulting in unbiased parameter estimation is :

$e(t+1)$ (or adaptation error) is a white noise for $\hat{\theta} = 0$

3.4 Structure of the Recursive Identification Method

All the recursive identification methods fit within the diagram given in figure (3.5)

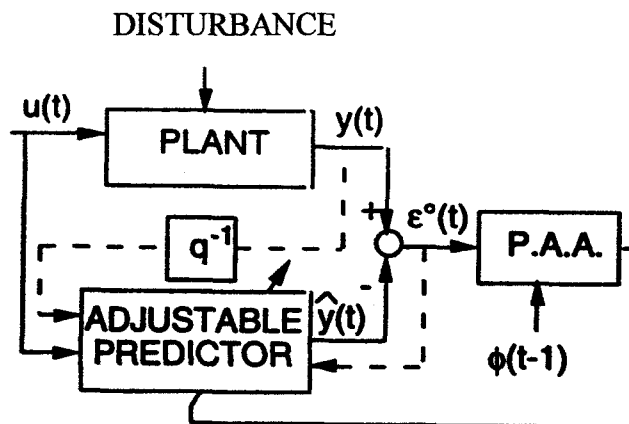


Figure 3.5. Structure of recursive identification method.

They all use the same structure for the parameter adaptation algorithm (PAA) with the different possible choices for the “adaptation gain”. They are distinguished from each other by the:

- predictor structure
- nature of the components of the observation vector (ϕ)
- dimension of the adjustable parameter vector $\hat{\theta}(t)$ and of the observation vector (ϕ)
- way in which the prediction errors and respectively the adaptation errors are generated

The convergence properties, in the presence of random disturbances, will depend on the different choices. In this study, the equation error (recursive least squares) method has been used for identification. This method tends to obtain a “white” prediction error (white noise) for a class of disturbance models by modeling the disturbance.

The structure, which has been used in this study, is ARX type (the special case of ARMAX, $C(q^{-1}) = 1$).

$$A(q^{-1})y(t) = q^{-d} B(q^{-1})u(t) + e(t) \quad (3.17)$$

in which the term $e(t)$ models the disturbances.

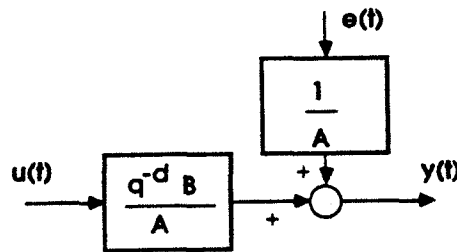


Figure 3.6. Structure of recursive least square (R.L.S.).

3.4.1 Estimation of the model complexity

The aim of the complexity estimation is to estimate the orders of each polynomial of the plant model (degrees of A and B), as well as the pure time delay (d), directly from the Input/output data, without identifying the coefficients of the model. When there is no noise. We can apply the rank test means to look for the number of columns to add to be able to express as $\tilde{y}(0)$ a linear combination of the other columns. This corresponds to look for the order of the model, i.e. $\max(n_a, n_b+d)$. But this method does not work in practice, when there is some noise.

3.4.2 Complexity estimation with white noise

Let the plant model be described by:

$$y(t) = -a_1 y(t-1) + b_1 u(t-1) \quad (3.18)$$

The present output will be noted $y(0)$, and previous values will be indexed $-i$. A matrix is then built as follows:

- the first row contains $y(0)$ and all the variables present in the corresponding equation,
- the other rows are made of the previous values, with the same time progression.

$$\begin{bmatrix} y(0) & \vdots & y(-1) & u(-1) \\ y(-1) & \vdots & y(-2) & u(-2) \\ y(-2) & \vdots & y(-3) & u(-3) \end{bmatrix}$$

$$\begin{matrix} \uparrow \\ \tilde{y}(0) \end{matrix} \quad \begin{matrix} \uparrow \\ R(1) \end{matrix} = \begin{matrix} \uparrow \\ R(n) \end{matrix}$$

This matrix may be divided in two parts: the first column, noted $\tilde{y}$ is a vector that contains the present and previous output the remaining part is a matrix, noted $R(1)$.

We can use a least square estimation with the following test value for an estimation $\tilde{n}$ of the order, if the noise is a white noise.

$$V_{LS}(\hat{n}) = \text{Min} \frac{1}{N} \|\hat{y}(0) - R(\hat{n})\hat{\theta}\|^2 \quad (3.19)$$

where $\hat{\theta}$ express a linear dependence of $\tilde{y}(0)$ upon the columns of $R(\tilde{n})$.

3.5 Parameter Adaptation Algorithm (PAA)

The general structure of parameter adaptation algorithm will be presented in the context of the Recursive Least Squares identification method [44].

The “a priori” adjustable predictor for Recursive Least Squares is given in the general case by:

$$\hat{y}^\circ(t+1) = -\sum_{i=1}^{na} \hat{a}_i y(t+1-i) + \sum_{i=1}^{nb} \hat{b}_i u(t-d-i+1) = \hat{\theta}^T \phi(t) \quad (3.4)$$

in which

$$\hat{\theta}^T(t) = [\hat{a}_1(t), \dots, \hat{a}_{na}(t), \hat{b}_1(t), \dots, \hat{b}_{nb}(t)] \quad (3.5)$$

$$\phi^T = [-y(t) \dots -y(t-n_a+1), u(t-d), \dots u(t-d-n_b+1)] \quad (3.6)$$

and the “a priori” prediction error is given by:

$$e^\circ(t+1) = y(t+1) - \hat{y}^\circ(t+1) = y(t+1) - \hat{\theta}^T(t) \phi(t) \quad (3.20)$$

The aim is to find a recursive parameter adaptation algorithm, which minimizes the “least squares” criterion:

$$\min_{\theta} J(t) = \sum_{i=1}^t [y(i) - \hat{\theta}(t)^T \phi(i-1)]^2 \quad (3.21)$$

The term $\hat{\theta}(t)^T \phi(i-1)$ corresponds :

$$\hat{\theta}(t)^T \phi(i-1) = -\hat{a}_1(t)y(i-1) + \hat{b}_1(t)u(i-1) = \hat{y}(\theta(t)) \quad (3.22)$$

This is the prediction of the output at instant i based on the parameter estimate at instant t obtained using t measurements.

A first Formulation of the recursive least squares (RLS) parameter adaptation algorithm (PAA) is given below:

$$\hat{\theta}(t+1) = \hat{\theta}(t) - P(t+1)\phi(t)e^\circ(t+1) \quad (3.23)$$

$$P(t+1) = P(t) - \frac{P(t)\phi(t)^T P(t)}{1 + \phi(t)^T P(t)\phi(t)} \quad (3.24)$$

$$e^\circ(t+1) = y(t+1) - \hat{\theta}(t)^T \phi(t) \quad (3.20)$$

An equivalent form of this algorithm is obtained by introducing the expression of $P(t+1)$ given by equation (3.30) in equation (3.29),

$$[\hat{\theta}(t+1) - \hat{\theta}(t)] = P(t+1)\phi(t)e(t+1) = P(t)\phi(t) \frac{e^\circ(t+1)}{1 + \phi(t)^T P(t)\phi(t)} \quad (3.25)$$

is then obtained.

However the “a posteriori” prediction error is given by:

$$\begin{aligned} e(t+1) &= y(t+1) - \hat{\theta}(t+1)^T \phi(t) = y(t+1) - \hat{\theta}(t)^T \phi(t) - [\hat{\theta}(t+1) - \hat{\theta}(t)]^T \phi(t) \\ &= e^\circ(t+1) - \phi(t)^T P(t)\phi(t) \frac{e^\circ(t+1)}{1 + \phi(t)^T P(t)\phi(t)} = \frac{e^\circ(t+1)}{1 + \phi(t)^T P(t)\phi(t)} \end{aligned} \quad (3.26)$$

which expresses the relation between the “a posteriori” prediction error and the “a priori” prediction error. Using this relation in equation (3.32) an equivalent form of the parameter adaptation algorithm of the recursive least squares is obtained:

$$\hat{\theta}(t+1) = \hat{\theta}(t) - P(t) \phi(t) e(t+1) \quad (3.27)$$

$$P(t+1)^{-1} = P(t)^{-1} - \phi(t) \phi(t)^T \quad (3.28)$$

$$P(t+1) = P(t) - \frac{P(t) \phi(t) \phi(t)^T P(t)}{1 + \phi(t)^T P(t) \phi(t)} \quad (3.29)$$

$$e(t+1) = \frac{y(t+1) - \hat{\theta}(t)^T \phi(t)}{1 + \phi(t)^T P(t) \phi(t)} \quad (3.30)$$

The recursive least squares algorithm is an algorithm with a decreasing adaptation gain. This is clearly seen if the estimation of a single parameter is considered. In this case $P(t)$ and $\phi(t)$ are scalars and equation (3.29) becomes:

$$P(t+1) = \frac{P(t)}{1 + \phi(t)^2} \leq P(t) \quad (3.31)$$

The recursive least squares algorithm in fact gives less and less weight to the new prediction errors and thus to the new measurements.

Consequently this type of variation of the adaptation gain is not suitable for the estimation of time varying parameters and other variation profiles for the adaptation gain must therefore be considered.

3.5.1 Choice of the adaptation gain

The recursive formula for the inverse of the adaptation gain $P(t+1)^{-1}$ given by equation (3.35) is generalized by introducing two weighting sequences $\lambda_1(t)$ and $\lambda_2(t)$, and the adaptation algorithm becomes:

$$\hat{\theta}(t+1) = \hat{\theta}(t) - P(t+1)\phi(t)e(t+1) \quad (3.27)$$

$$P(t+1)^{-1} = \lambda_1 P(t)^{-1} - \lambda_2 \phi(t)\phi(t)^T \quad (3.32)$$

$$0 \leq \lambda_1(t) \leq 1 ; \quad 0 \leq \lambda_2(t) \leq 2 ; \quad P(0) > 0$$

$$e(t+1) = \frac{y(t+1) - \hat{\theta}(t)^T \phi(t)}{1 + \phi(t)^T P(t) \phi(t)} \quad (3.30)$$

Note that $\lambda_1(t)$ and $\lambda_2(t)$ in equation (3.32) have the opposite effect. $\lambda_1(t) < 1$ tends to increase the adaptation gain (the gain inverse decreases), $\lambda_2(t) > 0$ tends to decrease the adaptation gain (the gain inverse increases). Each choice of sequences $\lambda_1(t)$ and $\lambda_2(t)$ corresponds a variation ‘profile’ of the adaptation gain and an interpretation in terms of the error criterion which is minimized by the PAA. Using the matrix inversion lemma, one obtains from equation (3.32):

$$P(t+1) = \frac{1}{\lambda_1(t)} \left[P(t) - \frac{P(t)\phi(t)\phi(t)^T P(t)}{\frac{\lambda_1(t)}{\lambda_2(t)} + \phi(t)^T P(t)\phi(t)} \right] \quad (3.33)$$

3.5.2. Constant forgetting factor

In this case:

$$\lambda_1(t) = \lambda_1 ; \quad 0 < \lambda_1 < 1 ; \quad \lambda_2(t) = \lambda_2 = 1$$

The typical values for λ_1 are:

$$\lambda_1 = 0.95 \text{ to } 0.99$$

The criterion to be minimized will be:

$$J(t) = \sum_{i=1}^t \lambda_1^{(t-i)} \left[y(i) - \hat{\theta}(t)^T \phi(i-1) \right]^2 \quad (3.34)$$

The effect of $\lambda_1 < 1$ is to introduce increasingly weaker weighting on the old data ($i < t$). This is why λ_1 is known as the “forgetting factor”. The maximum weight is given to the most recent error. This type of profile is suited to the identification of slowly time varying systems.

3.6 Parameterization of Multivariable Systems

Consider a linear system with m inputs and p outputs represented by:

$$A(z)y(t) = B(z)u(t) \quad (3.35)$$

where $y(t)$ and $u(t)$ are input vectors and outputs of dimension $p \times 1$ and $m \times 1$, respectively, t is the time sampling period and, z is the forward shift operator, $A(z)$ and $B(z)$ are polynomial matrices of dimension $p \times p$ and $p \times m$ respectively

$$A(z) = \sum_{i=0}^q A_i z^i \quad (3.36)$$

$$B(z) = \sum_{i=0}^q B_i z^i \quad (3.37)$$

In the A_i and B_i matrices, it exists parameters called “trivial”, which have the value 0 or 1, whom the position depends on the system structure. The parameterization of the system consists to determine the position of this trivial parameters, or in other words, to determine the position of the non-trivial parameters which are to identify.

Parameterization of the matrix $A(z)$ and $B(z)$:

Let M_i the impulse response of the system:

$$G(z) = \sum_{t=1}^{\infty} M_i z^{-t} \quad (3.38)$$

where $G(z) = A(z)^{-1} B(z)$ is the transfer matrix of the system.

By replacing $y(t)$ and $u(t)$ in (3.42) by M_t and the impulse $\delta(t)$ ($\delta(0) = 1$ and $\delta(t) = 0$ for $t \neq 0$), respectively, we have:

$$\begin{bmatrix} A_0 & A_1 & \dots & A_q \end{bmatrix} \cdot \begin{bmatrix} M_1 & M_2 & M_3 & \dots & M_M \\ M_2 & M_3 & M_4 & \dots & M_{M+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M_{q+1} & M_{q+2} & M_{q+3} & \dots & M_{M+q} \end{bmatrix} = [0] \quad (3.39)$$

$$\begin{bmatrix} A_1 & A_2 & \dots & A_q \end{bmatrix} \cdot \begin{bmatrix} M_1 & 0 & 0 & \dots & 0 \\ M_2 & M_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M_q & M_{q-1} & M_{q-2} & \dots & M_1 \end{bmatrix} = [B_0 \ B_1 \ \dots \ B_{q-1}] \quad (3.40)$$

The parameterization of the model will be determined from equation (3.39) and (3.40).

The Hankel matrix is defined by:

$$H_{q+1,M} = \begin{bmatrix} M_1 & M_2 & M_3 & \dots & M_M \\ M_2 & M_3 & M_4 & \dots & M_{M+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ M_{q+1} & M_{q+2} & M_{q+3} & \dots & M_{M+q} \end{bmatrix} \quad (3.41)$$

$H_{q+1,M}$ is the coefficient matrix in (3.39). n is the McMillan degree of the system.

For M large, we have:

$$\text{rang} \{ H_{q+1,M} \} = n \quad (3.42)$$

From (3.39), the parameters of $[A_0 \ A_1 \ \dots \ A_q]$ are the coefficients of a dependence relationship between the lines of $H_{q+1,M}$. As $H_{q+1,M}$ is only determined by the system, this dependence relationship too determined by the system.

3.6.1 Properties of the parameterization

Identifiability:

A parameterization is identifiable if all its parameters are only determined from the input-output data.

From the section “Parameterization of the matrix $A(z)$ and $B(z)$ ”, the $A(z)$ parameters are the coefficients of the dependence relationship between lines of Hankel matrix. This Hankel matrix is only defined by the system, then the parameters of $A(z)$ too. The parameters of $A(z)$ are only defined by the parameters of $B(z)$ and the system step response.

Thus, in this parameterization, the parameters of $A(z)$ and $B(z)$ are determined according an only way. The parameterization is identifiable.

Representation Capacity:

We have to know if the parameters of $A(z)$ and $B(z)$ are enough to represent the system, i.e. if we can compute the impulse response of the system from these parameters.

From (3.39) and (3.40), we have:

$$\begin{bmatrix} 0 & & 0 & A_0 & A_{q-1} & A_q \\ 0 & & 0 & A_0 & A_1 & A_q \\ & & & & & \\ A_0 & A_{q-1} & A_q & 0 & & \\ A_1 & A_q & 0 & & & \\ & & & & & \\ A_q & 0 & & & & 0 \end{bmatrix} X \begin{bmatrix} M_1 \\ M_2 \\ M_3 \\ \\ M_{M+q} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \\ 0 \\ B_0 \\ B_{q-1} \end{bmatrix} \quad (3.43)$$

If the matrix A_q is non-singular, so is the coefficient matrix in the left hand side of the equation (3.49). It follows that the impulse response of the system can be determined from the matrices $A(z)$ and $B(z)$. Thus if the parameters of $A(z)$ and $B(z)$ satisfied (3.39) and (3.40), the model (3.35) determines uniquely the system.

Minimality:

In order to get $y_i(t)$ independent of its future value, $L(i)$ must satisfy the condition:

$$L(i) > \max\{S(i)\} \quad (3.44)$$

Let us consider the matrix $A(z)$. Each element of $A(z)$ is a z polynomial. We call “degree-column” of the i -th column of the matrix $A(z)$, the greatest degree of the polynomial on this column.

The matrix $A(z)$ is called “proper column”, if the matrix of the coefficients of terms which the degree is equal to the degree-column of the column including these terms is regular.

With the condition (3.50), the polynomial matrix $A(z)$ is always proper column with degree-column = $L(i)-1$. Thus:

$$\deg\{\det A(z)\} = \sum_{i=1}^p \text{degree column} = \sum_{i=1}^p [L(i)-1] \quad (3.45)$$

Thus model is minimal if and only if:

$$\sum_{i=1}^p [L(i)-1] = \sum_{i=1}^p n_i = n \quad (3.46)$$

where n_i is the number of $S(i)$ elements. This condition with the condition (3.44) imply that the n independent lines have to be chosen into the line blocs indexed with:

$$S(i) = \{1, 2, 3, \dots, n_{i-1}, n_i\} \quad \forall i \quad (3.47)$$

(i.e., if the row $h(i,k)$ with $k > 1$ then so the row $h(i, k-1)$ too) and that:

$$L(i) = n_i + 1 \quad (3.48)$$

(i.e., the row $h(i, L(i))$ is the first row in the Hankel matrix, which is linearly dependent on the chosen n rows)

3.6.2 Canonical parameterization (Guidorzi)

Canonical parameterization of Guidorzi has been applied in the thesis. The details about Guidorzi have been discussed in the next section.

This parameterization results in the following equation:

$$y_i(t + \hat{n}_i + 1) = \sum_{j=1}^p \sum_{k=1}^{\hat{n}_{ij}} \hat{a}_{ijk} y_j(t + k) + \sum_{j=1}^m \sum_{k=1}^{\hat{n}_i} \hat{b}_{ijk} u_j(t + k) + \hat{e}_i(t + \hat{n}_i + 1) \quad (3.49)$$

3.6.3 Connection between state space and ARX model

The sampling state representation of the model can be represented by :

$$x(t+1) = F x(t) + G u(t) \quad (3.50)$$

$$Y(t) = H x(t) \quad (3.51)$$

where F is the system dynamical matrix of dimension $n \times n$, G is the input distribution matrix of dimension $n \times m$ and H is the output distribution matrix of dimension $p \times n$; $u(t)$ is the vector of the inputs at the instant t , $y(t)$ is the outputs of this and $x(t)$ is the system state vector, whose the dimension is equal to sum of the observability indices.

Parameterization coefficients can be replaced for state space model as follows :

$$F_{ij} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ a_{ij1} & \cdots & a_{ijn_{ij}} & 0 & \cdots & 0 \end{bmatrix} \quad (3.52)$$

for $i = j$:

$$F_{ii} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ a_{ij1} & \cdots & \cdots & \cdots & a_{ijn_i} \end{bmatrix} \quad (3.53)$$

$$H = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & & & & & & & & \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix} \quad (3.54)$$

$$G = \begin{bmatrix} g_{li1} & \cdots & g_{mi1} \\ \vdots & & \vdots \\ g_{lin_i} & \cdots & g_{min_i} \end{bmatrix} \quad (3.55)$$

where the scalars b_{ijk} are linked to the scalars g_{ijk} , entries of $\tilde{G}$, by the bijection

$$\tilde{G} = VG$$

$$\tilde{G} = \begin{bmatrix} b_{il1} & \cdots & b_{im1} \\ \vdots & & \vdots \\ b_{iln_i} & \cdots & b_{imn_i} \end{bmatrix} \quad (3.56)$$

The matrix V is structurally nonsingular. In fact, in every case , $\det(V) = 1$.

$$V_{ij} = \begin{bmatrix} -a_{ij2} & -a_{ij3} & \cdots & -a_{ijn_i} & 0 \\ -a_{ij3} & \cdots & \cdots & 0 & \cdots \\ \vdots & & \ddots & & \vdots \\ -a_{ijn_{ij}} & 0 & \ddots & & \vdots \\ 0 & & & & \vdots \\ \vdots & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix} \quad (3.57)$$

For $i = j$:

$$V_{ii} = \begin{bmatrix} -a_{ij2} & -a_{ij3} & \cdots & -a_{ijn_i} & 1 \\ -a_{ij3} & \cdots & \cdots & 1 & \cdots \\ \vdots & & & & \\ -a_{ijn_i} & 1 & \cdots & & \\ 1 & & & & \end{bmatrix} \quad (3.58)$$

3.7 Validation of the Model

In system identification both the determination of model structure and model validation are important aspects [44-52]. An overparameterized model structure can lead to unnecessarily complicated computations for finding the parameter estimates and for using the estimated model. An underparameterized model may be very inaccurate.

The choice of model structure in practice is influenced greatly by the intended use of the model. A stabilizing regulator can often be based on a crude low-order model, whereas more complex and detailed models are necessary if the model is aimed at giving physical insight into the process.

In practice one often performs identification for an increasing set of model orders (or more generally, structural indices). Then one must know when the model order is appropriate, i.e. when to stop. Needless to say, any real-life data set cannot be modeled exactly by a linear finite-order model. Nevertheless such models often give good approximations of the true dynamics. However, the methods for finding the 'correct' model order are based on the statistical assumption that the data come from a true system within the model class considered.

The principle of the validation method is as follows:

- if the "plant + disturbance" structure chosen is correct, i.e. representative of reality
- if an appropriate identification method for the structure chosen has been used
- if the degrees of the polynomials $A(q^{-1})$, $B(q^{-1})$, $C(q^{-1})$ and the value of d (delay)

have been correctly chosen.

The validation method implements this principle. It is made up of several steps:

- 1) Creation of an I/O file for the identified model (using the same input sequence as for the system).
- 2) Creation of a prediction error file for the identified model (minimum 100 data).
- 3) “Whiteness” (independence) test on the prediction errors sequence (also known as residual prediction errors).

For a good model, $y_m(t)$ should resemble the measured output. The deviations of $y_m(t)$ from $y(t)$ are due both to modeling errors and to the disturbances. It is therefore important to realize that if the data are noisy then $y_m(t)$ should differ from $y(t)$. It should only describe the part of the output that is due to the input signal. There are several statistical tests on the prediction errors $e(t, \hat{\theta}_N)$. The prediction errors evaluated at the parameter estimate $\hat{\theta}_N$ are often called the residuals.

The methods for model structure determination based on tests of the residuals are tied to model structures and identification methods where the disturbances are explicitly modeled. This means that they are not suitable if, for example, an instrumental variable method is used for parameter estimation, but they are well adapted to validate models obtained by a prediction error method.

This section is concerned with the validation of models identified using identification method based on the “whitening” of the prediction error (Recursive Least Squares). The prediction error $e(t)$ asymptotically tends towards a white noise, which implies:

$$\lim_{t \rightarrow \infty} E\{e(t)e(t-i)\} = 0 \quad i = 1, 2, 3 \quad (3.16)$$

3.7.1 Whiteness test

Let $\{e(t)\}$ be the centered sequence of the residual prediction errors (centered: measured values mean value). One computes:

$$R(0) = \frac{1}{N} \sum_{t=1}^N e^2(t) ; \quad R N(0) = \frac{R(0)}{R(0)} = 1 \quad (3.59)$$

$$R(i) = \frac{1}{N} \sum_{t=1}^N e(t)e(t-i) ; \quad R N(i) = \frac{R(i)}{R(0)} ; \quad i = 1, 2, 3, \dots, n_a \dots \quad (3.60)$$

with $i_{\max} \geq \max(n_a, n_b + d)$

If the residual prediction error sequence is perfectly white (theoretical situation) and the number of samples is very large then $R N(0) = 1$; $R N(i) = 0$.

In real situations, however, this is never the case. $e(t)$ contains residual structural errors (order errors, non-linear effects, non gaussian noises) and on the other hand, the number of samples is in general relatively small (several hundreds). Also it should be kept in mind that one always seeks to identify “good” simple models (with few parameters).

One considers as a practical validation criterion:

$$R N(0) = 1 ; \quad |R N(i)| < \frac{2.17}{\sqrt{N}}$$

The basic practical numerical value for the validation criterion:

$$R N(0) = 1 ; \quad |R N(i)| \leq 0.15 ; \quad i \geq 1 \quad (3.61)$$

The following remarks are imperative:

- A “too good” validation criterion indicates that model simplifications may be possible.
- If several identified models have the same complexity one chooses the model given by the methods which leads to the smallest $|R N(i)|$.

Note equally that a complete model validation implies, after the validation using the input/output sequence employed for identification, a validation using a plant input/output sequence other than the one used for identification.

Another remark, if the level of the residual prediction errors is very low, the bias on the RLS is negligible and the validation criterion loses its signification because the residual noise may not be gaussian (essentially round off errors).



4. IDENTIFICATION OF THE RAW MATERIAL BLENDING PROCESS

4.1 Process Models

Three different multivariable stochastic models (ARX), which have same inputs and different outputs, have been constructed to find an estimate of the raw material blending plant parameters in cement industry. ARX model has been parameterized by using Guidorzi canonical form. In this parameterization [31-33], the n independent rows are the first independent rows of the Hankel matrix. The p dependent rows are the i -th rows in the blocs indexed by:

$$L(i) = n_i + 1 \quad (3.48)$$

Moreover, by the way the base is chosen, the row $h(i, L(i))$ is independent of the rows below it in the Hankel matrix. This parameterization results in the following equation:

$$y_i(t + \hat{n}_i + 1) = \sum_{j=1}^p \sum_{k=1}^{\hat{n}_{ij}} \hat{a}_{ijk} y_j(t + k) + \sum_{j=1}^m \sum_{k=1}^{\hat{n}_i} \hat{b}_{ijk} u_j(t + k) + \hat{e}_i(t + \hat{n}_i + 1) \quad (3.49)$$

where $\hat{a}_{ijk}$ and $\hat{b}_{ijk}$ are the parameters, $\hat{e}_i(t)$ is a white noise and $\hat{n}_i$ is the observability index of the model.

The integers $\hat{n}_{ij}$ are determinated by:

$$\hat{n}_{ij} = \min\{\hat{n}_i + 1, \hat{n}_j\} \text{ for } i > j$$

$$\hat{n}_{ij} = \min\{\hat{n}_i, \hat{n}_j\} \text{ for } i \leq j \quad (4.1)$$

$$\hat{n}_{ij} = \hat{n}_i \text{ for } i = j$$

Corresponding ARX model to equation (3.49) with m inputs and p outputs is as follows:

$$A(z) y(t) = B(z) u(t) \quad (3.35)$$

where

$$A(z) = \begin{bmatrix} \tilde{a}_{11}(z) & \cdots & \tilde{a}_{1p}(z) \\ \vdots & & \vdots \\ \tilde{a}_{p1}(z) & \cdots & \tilde{a}_{pp}(z) \end{bmatrix} \quad (3.36)$$

$$B(z) = \begin{bmatrix} \tilde{b}_{11}(z) & \cdots & \tilde{b}_{1m}(z) \\ \vdots & & \vdots \\ \tilde{b}_{p1}(z) & \cdots & \tilde{b}_{pm}(z) \end{bmatrix} \quad (3.43)$$

Parameterization coefficients can be replaced as follows for ARX model :

$$\begin{aligned} \tilde{a}_{ii}(z) &= z^{n_i} - a_{iin_1} z^{n_i-1} - \cdots - a_{ii2} z - a_{ii1} \\ \tilde{a}_{ij}(z) &= -a_{iin_{ij}} z^{n_{ij}-1} - \cdots - a_{ij2} z - a_{ij1} \\ \tilde{b}_{ij}(z) &= -b_{ijn_1} z^{n_i-1} - \cdots - b_{ij2} z - b_{ij1} \end{aligned} \quad (4.2)$$

where the system input vector u represents low grade and iron ore. The system output y represents iron oxide, and /or lime module.

The integers n_i 's in the equation (3.49) are the observability indices. We have :

$$\deg\{\det A(z)\} = \sum_{i=1}^p n_i = n \quad (3.45)$$

The parameterization is thus minimal. The number of the parameters is $\leq n(m+p)$.

4.2 Model Structure Estimation of the Raw Material Blending Process

Indeed, the only possible knowledge of the number of system states is not enough, it is necessary to know more, by the mean of the observability index (n_i) of a multivariable model, which are a set of invariants attached with each system. In this study, The Least Square Method has been used for determining the observability index ($\hat{n}_i$) that the sum of which gives a minimal order of a state realization. The method based on The Least Square (LS) function will be presented as follows:

The prediction error of the model is obtained by (3.49) :

$$\hat{e}_i(t + \hat{n}_i + 1) = y_i(t + \hat{n}_i + 1) - \sum_{j=1}^p \sum_{k=1}^{\hat{n}_{ij}} \hat{a}_{ijk} y_j(t + k) + \sum_{j=1}^m \sum_{k=1}^{\hat{n}_i} \hat{b}_{ijk} u_j(t + k) \quad (4.3)$$

where, $y(t)$ is the measured output of the system.

We can use a least square estimation with the following test value for an estimation $\hat{n}_i$ of the order, if the noise is a white noise. The LS criterion $V_{LSi}(\hat{n}_i)$ is the smallest power (over the parameter vector $\hat{\theta}_i$) of the prediction $\hat{e}_i(t)$.

$$V_{LSi}(\hat{n}_i) = \min_{\hat{\theta}_i} \frac{1}{N} \sum_{t=1}^N [\hat{e}_i(t)]^2 \quad (4.4)$$

where N is the number of data and is $\hat{\theta}_i$ the vector containing the parameters $\hat{a}_{ijk}$ and $\hat{b}_{ijk}$.

$V_{LSi}(\hat{n}_i)$ is a non-increasing function of $\hat{n}_i$ as illustrated in figure (4.1). $V_{LSi}(\hat{n}_i)$ is also function of N as well as other factors such as the experimental conditions but, for notational simplicity, it is written in the present form $V_{LSi}(\hat{n}_i)$. The index $\hat{n}_j$ can be estimated by the value of $\hat{n}_i$ where the decreasing rate of $V_{LSi}(\hat{n}_i)$ is no longer significant.

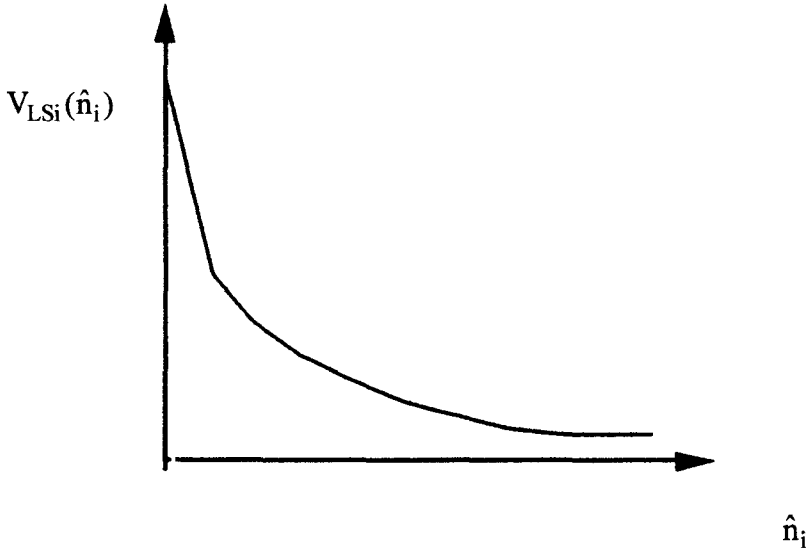


Figure 4.1. $V_{LSi}(\hat{n}_i)$ as function of $\hat{n}_i$.

In practice, it is often difficult to decide by visual inspection whether or not the decreasing rate of $V_{LSi}(\hat{n}_i)$ is significant. A solution to the problem consists of detaining an information criterion whose the general form is:

$$CV_{LSi}(\hat{n}_i) = V_{LSi}(\hat{n}_i) + \hat{d}_i X(N) \quad (4.5)$$

where $\hat{d}_i$ is the dimension of or the number of parameters in the equation (3.49), and $X(N)$ is a function which decreases as N increases.

In equation (4.5), the penalty term $\hat{d}_i X(N)$ is an increasing function of $\hat{n}_i$. The criterion $CV_{LSi}(\hat{n}_i)$ has a minimum due to the different evolution of the two terms in the right hand side of (4.5). The value of $\hat{n}_i$ which minimizes $CV_{LSi}(\hat{n}_i)$ is an estimation of $\hat{n}_i$. $CV_{LSi}(\hat{n}_i)$ function is illustrated in figure (4.2).

In figure (4.2),

1: $V_{LSi}(\hat{n}_i)$

2: $\hat{d}_i X(N)$

3: $CV_{LSi}(\hat{n}_i)$.

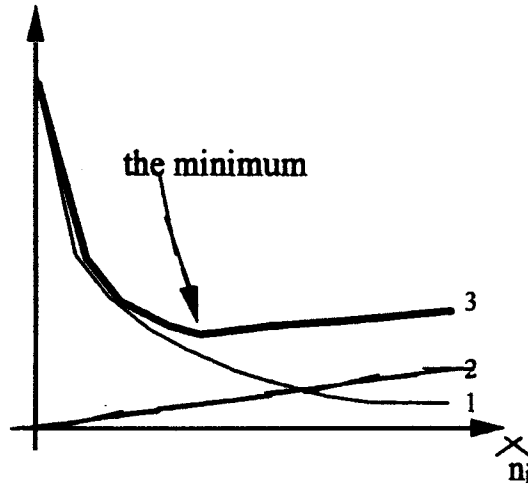


Figure 4.2. Typical evolution of $V_{LSi}(\hat{n}_i)$, $d_i X(N)$ and $C V_{LSi}(\hat{n}_i)$ as function of $\hat{n}_i$.

$V_{LSi}(\hat{n}_i)$ is computed as function $\hat{n}_i$. The index n_i is next estimated by minimizing the BIC (Bayesian Information Criterion).

$$BIC_{LSi}(\hat{n}_i) = V_{LSi}(\hat{n}_i) + \hat{d}_i \frac{\log(N)}{N} \quad (4.6)$$

where $CV_{LSi}(\hat{n}_i) = BIC_{LSi}(\hat{n}_i)$. The choice $X(N) = \log(N)/N$ in (4.6) leads to a consistent estimator in the white noise case.

4.3 Parameter Estimation (RLS) of the Raw Material Blending Process

The aim is to find a recursive parameter adaptation algorithm, which minimizes the “least squares” criterion defined by (3.34): This method gives an estimate of the raw material compositions ($\hat{\theta}$) which minimizes the output error which is the difference between the measured and computed module or oxide percentage values.

The computation is done via the following recursive equations :

$$\hat{\theta}_i(t+1) = \hat{\theta}_i(t) - P_i(t+1) \phi_i(t) e_i(t+1) \quad (3.27)$$

$$P_i(t+1) = \frac{1}{\lambda_1} \left[P_i(t) - \frac{P_i(t)\phi_i(t)\phi_i(t)^T P_i(t)}{\lambda_1 + \phi_i(t)^T P_i(t)\phi_i(t)} \right] \quad (3.33)$$

$$e_i(t+1) = y_i(t+1) - \hat{y}_i(t+1) = y_i(t+1) - \hat{\theta}_i(t)^T \phi_i(t) \quad (3.30)$$

where $\phi_i(t)$ is the regressor vector corresponding to the equation of the i -th output (obtained after removing the noise part in (3.49)), $e(t+1)$ is the prediction error and $P_i(t+1)$ is the matrix gain when using an increase gain, λ_1 ($\lambda_1=0.99$) is the forgetting factor. Note that the argument t in the parameter $\hat{\theta}_i$ vector implies that this vector is time-varying since with a recursive identification method, the parameter vector of the model is time varying.

4.4 The Whiteness Test of the Raw Material Blending Process Model

This test consists of verifying if the prediction error signal (4.3) can be considered as a white noise. The test is realizing by computing the normalized autocorrelation function:

$$\sqrt{N} \frac{\Gamma_{ei}(\tau)}{\Gamma_{ei}(0)} \quad \tau = 1, 2, \dots, \tau_{\max} \quad (4.7)$$

where,

$$\Gamma_{ei} = \frac{1}{N} \sum_{t=1}^{N-\tau} e_i(t+\tau)e_i(t) \quad (4.8)$$

is the autocorrelation function of the signal $e_i(t)$.

If $e_i(t)$ is a white noise then $\sqrt{N} \frac{\Gamma_{ei}(\tau)}{\Gamma_{ei}(0)}$ is asymptotically gaussian with zero mean and variance = 1. The normalization of the autocorrelation allows to determinate the zero threshold corresponding to a predefined interval. A typical value for the zero threshold for the normalized variable, $\sqrt{N} \frac{\Gamma_{ei}(\tau)}{\Gamma_{ei}(0)}$ for $\tau \neq 0$, ranges from 2 and 2.13

corresponds to the 97 % confidence interval). Using this test, the absolute value of all autocorrelation functions with $\tau \geq 1$ must be smaller than the chosen zero threshold value 0.15. The typical value $\tau_{\max}$ is n.

4.5 Identification Results

PGNAA analyzer in Nuh Cement Factory can analyze the crushed solids up 3 or 4 inch topsize with sufficient accuracy in a minute. In the identification part, parametric multivariable models having minimum order and minimum number of parameters have been obtained. Developed multivariable difference equation (time series) models have a direct connection with a canonical state space form of the models. Forgetting factor λ has been chosen equal to 0.99 in the parameter estimation algorithm. At the end of simulation, the predicted output $y_p(t)$ have resembled the measured output. The deviations of $y_p(t)$ from $y(t)$ are both to the modeling errors and to the disturbances. In the validation tests, the absolute value of the autocorrelation function value is smaller than the chosen zero threshold value 0.15. This means that the proposed models are reliable. Furthermore, with these developed models, the stable operation conditions have been obtained for the constant feed ratios of the raw material. The measured input-output data groups and filtered data groups of the process are shown in figure (4.3), (4.4) and the predicted and the measured outputs are compared in figure (4.5), (4.7), (4.9) for three models. Also, step responses of the models are shown in figure (4.6), (4.8), (4.10).

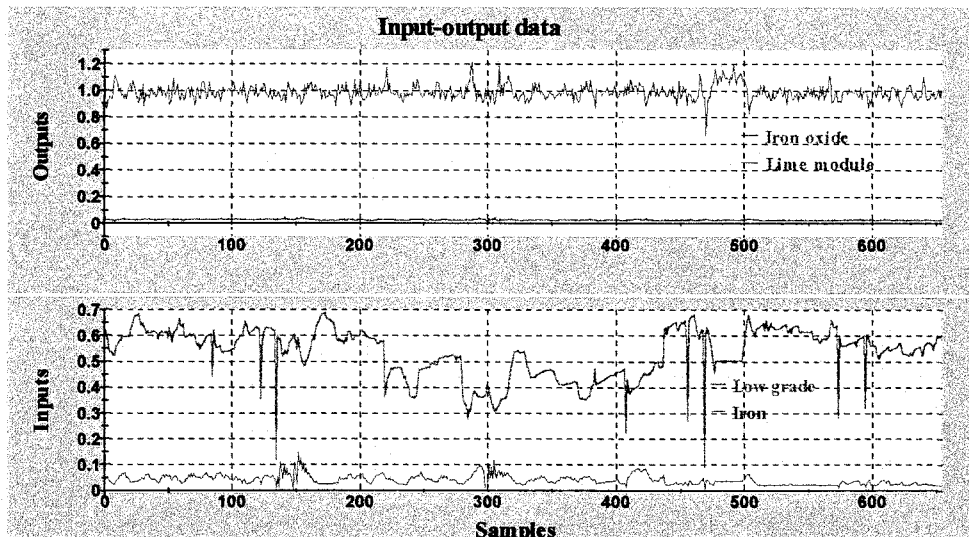


Figure 4.3. Measured input-output data of the process.

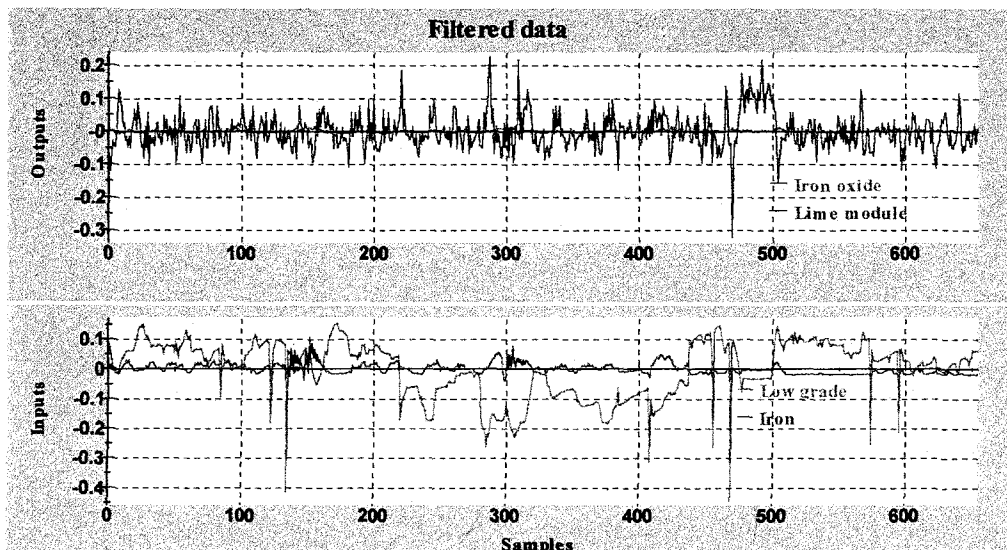


Figure 4.4. Filtered input-output data of the process.

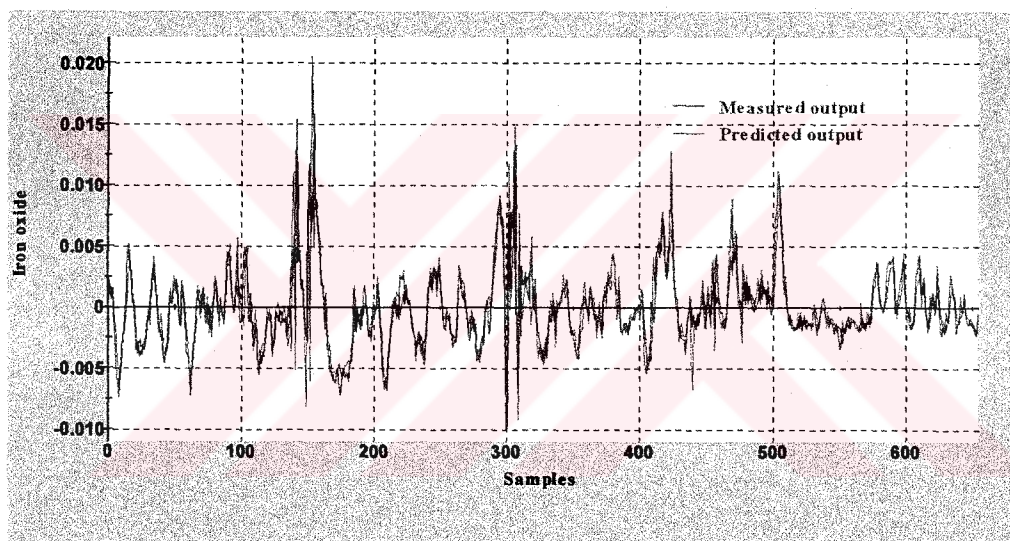


Figure 4.5. Measured and predicted outputs of the model 1.

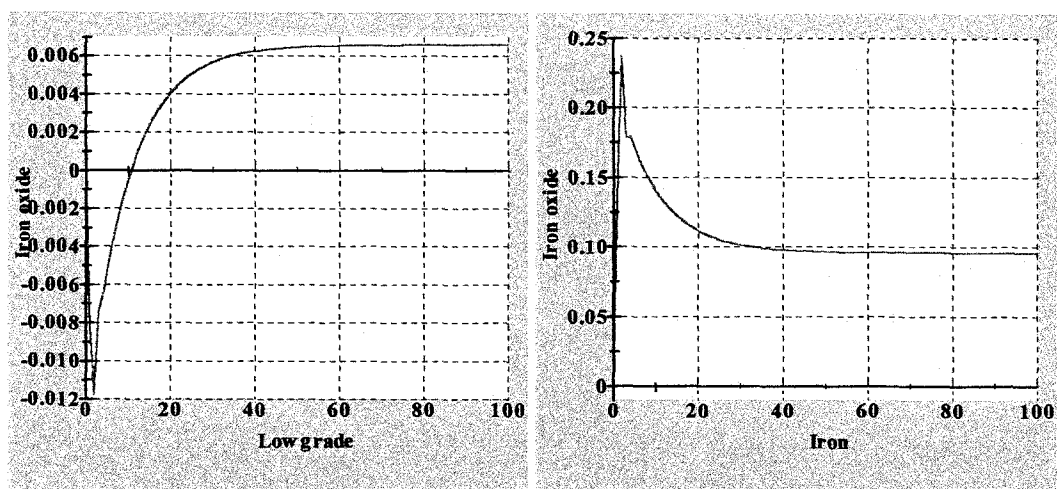


Figure 4.6. Step response of model 1.

Estimated parameter values of time series model 1 (MISO).

$$A_0 = 1 \quad A_1 = -0.538 \quad A_2 = -0.244 \quad A_3 = -0.077$$

$$B_1 = [-0.008 \quad 0.153] \quad B_2 = [0.001 \quad 0.001] \quad B_3 = [0.008 \quad -0.141]$$

State space representation of model 1:

$$F = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0.077 & 0.244 & 0.538 \end{bmatrix}$$

$$G = \begin{bmatrix} -0.008 & 0.153 \\ -0.004 & 0.084 \\ 0.004 & -0.059 \end{bmatrix}$$

$$H = [1 \quad 0 \quad 0]$$

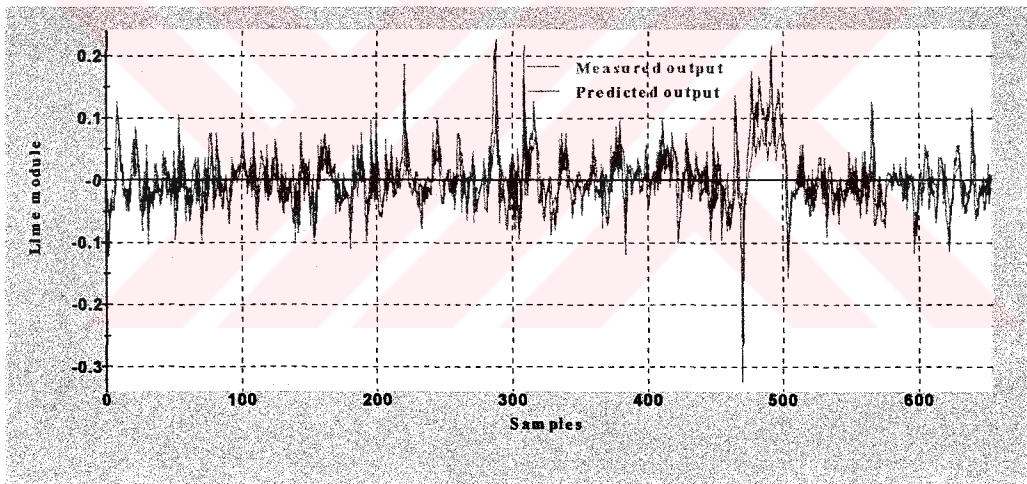


Figure 4.7. Measured and predicted outputs of model 2.

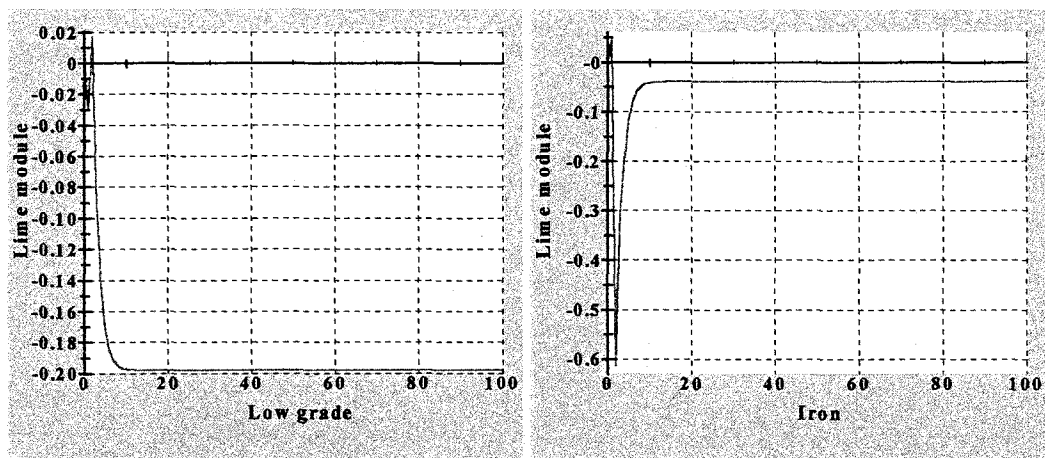


Figure 4.8. Step response of model 2.

Estimated parameter values of time series model 2 (MISO).

$$A_0 = 1 \quad A_1 = -0.472 \quad A_2 = -0.076 \quad A_3 = -0.039$$

$$B_1 = [-0.029 \quad 0.044] \quad B_2 = [0.060 \quad -0.678] \quad B_3 = [-0.128 \quad 0.615]$$

State space representation of model 2:

$$F = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.039 & 0.076 & 0.472 \end{bmatrix}$$

$$G = \begin{bmatrix} -0.029 & 0.044 \\ 0.046 & -0.658 \\ -0.108 & 0.308 \end{bmatrix}$$

$$H = [1 \quad 0 \quad 0]$$

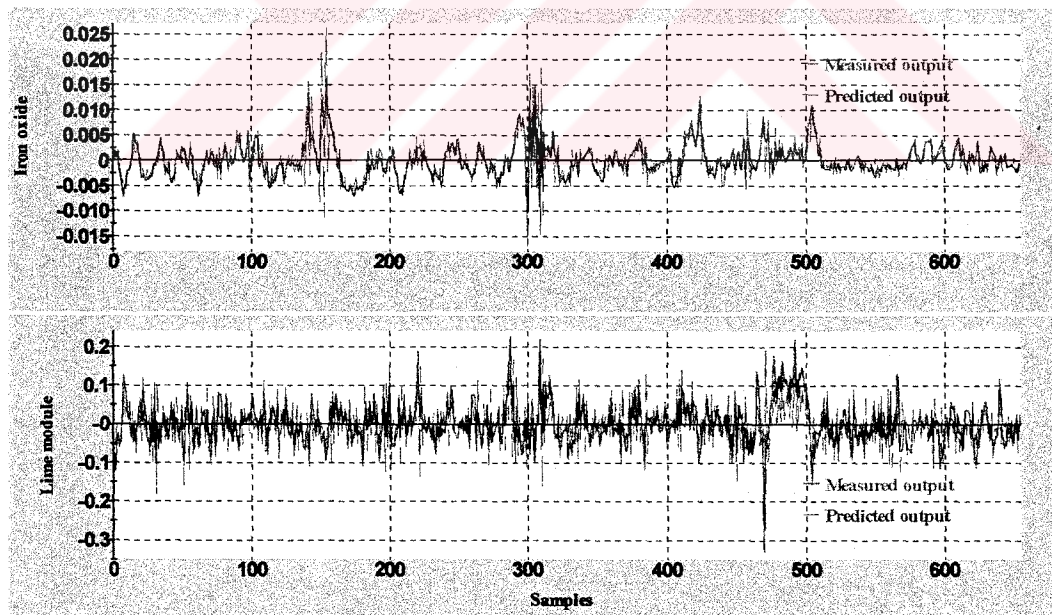


Figure 4.9. Measured and predicted outputs of model 3.

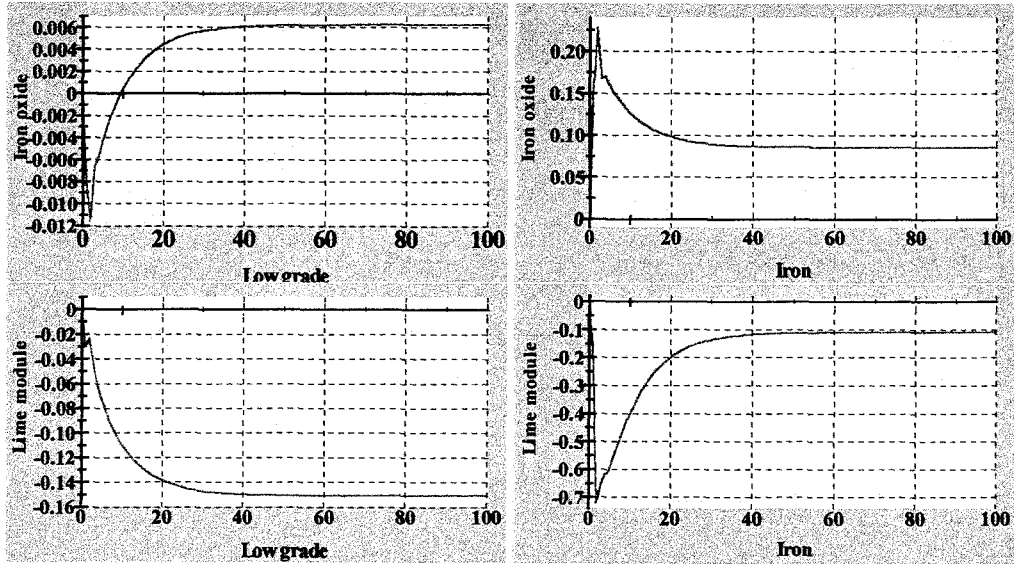


Figure 4.10. Step response of model 3.

Estimated parameter values of time series model 3 (MIMO).

$$A_0 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad A_1 = \begin{bmatrix} -0.522 & 0 \\ 3.845 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} -0.254 & 0.000 \\ -0.063 & -0.448 \end{bmatrix} \quad A_3 = \begin{bmatrix} -0.095 & -0.004 \\ -0.641 & -0.065 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} -0.008 & 0.145 \\ 0.000 & 0.000 \end{bmatrix} \quad B_2 = \begin{bmatrix} -0.001 & 0.007 \\ -0.061 & 0.401 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0.009 & -0.140 \\ 0.007 & -0.186 \end{bmatrix}$$

State space representation of model 3:

$$F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0.095 & 0.254 & 0.522 & 0.004 & 0.000 \\ 0 & 0 & 0 & 0 & 1 \\ 0.641 & 0.063 & -3.845 & 0.065 & 0.448 \end{bmatrix}$$

$$G = \begin{bmatrix} -0.008 & 0.145 \\ -0.004 & 0.082 \\ 0.005 & -0.061 \\ -0.030 & -0.158 \\ 0.007 & -0.565 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

5. MODEL BASED PREDICTIVE CONTROL (MPC)

The concept of predictive control was introduced by Richalet [53] and Cutler and Ramaker [54] in the late seventies. Model Predictive Control has developed considerably in the late eighties, especially since the seminal contribution by Clarke and co-workers [55], both within the research control community and in industry. The reason for this success can be attributed to the fact that Model Predictive Control is perhaps the most general way of posing the process control problem in the time domain. Model Predictive Control formulation integrates optimal control, stochastic control, control of processes with dead time, multivariable control and future references when available. Another advantage of Model Predictive Control is that because of the finite control horizon used, constraints, and in general non-linear processes, which are frequently found in industry, can be handled. Although Model Predictive Control has been found to be quite a robust type of control in most reported applications, stability and robustness proofs have been difficult to obtain because of the finite horizon used. This has been a drawback for a wider dissemination of Model Predictive Control in the control research community. Some new and very promising results in this context allow one to think that this control technique will experience greater expansion within this community in the near future. On the other hand, although a number of applications have been reported both in industry and research institutions, Model Predictive Control have not yet reached in industry the popularity that its potential would suggest. One of the reasons for this is that its implementation requires some mathematical complexities, which are not a problem in general for the research control community, where mathematical packages are normally fully available, but represent a drawback for the use of the technique by the control engineers in practice.

Model Based Predictive Control (MPC), is not a specific control strategy but more of a very ample range of control methods developed around certain common ideas. These design methods lead to linear controllers, which have practically the same

structure and present adequate degrees of freedom. The ideas appearing in greater or lesser degree in all the predictive control family are basically:

- Explicit use of a model to predict the process output at future time instants (horizon).
- Calculation of a control sequence minimizing a certain objective function.
- Receding strategy, so that at each instant the horizon is displaced towards the future, which involves the application of the first control signal of the sequence calculated at each step.

The various MPG algorithms only differ amongst themselves in the model used to represent the process and the noises and the cost function to be minimized. This type of control is of an open nature within which many works have been developed, being widely received by the academic world and by industry. There are many applications of predictive control successfully in use at the present time, not only in the process industry but also applications to the control of a diversity of processes ranging from robot manipulators [56] to clinical anesthesia [57]. Applications in the cement industry, drying towers and in robot arms are described in [58], whilst developments for distillation columns, PVC plants, steam generators or servos are presented in [59]. The good performance of these applications shows the capacity of the MPC to achieve highly efficient control systems able to operate during long periods of time with hardly any intervention.

MPC presents a series of advantages over other methods, amongst which stand out:

- it is particularly attractive to staff with only a limited knowledge of control, because the concepts are very intuitive and at the same time the tuning is relatively easy.
- it can be used to control a wide variety of processes without the designer having to take special precautions. It can be used to control ‘simple’ processes as well as ‘difficult’ processes, such as processes with a large time delay, processes that are non-minimum phase and processes that are open-loop unstable.

- it is not restricted to single-input, single-output (SISO) processes. Predictive controllers can be derived for and applied to multi-input, multi-output (MIMO) processes [60].
- it intrinsically has compensation for dead times.
- it introduces feed forward control in a natural way to compensate for measurable disturbances and for tracking reference trajectories.
- its extension to the treatment of constraints is conceptually simple and these can be included systematically during the design process. Since in real life process constraints are common practice, this issue is rather important. Although it is well known that predictive controllers have this property, little attention is paid in the literature to predictive controller design when there are constraints. This feature in particular is believed to be one of the most attractive aspects of predictive controller design.
- it is a totally open methodology. That is, within the framework of predictive control there are many ways to design a predictive controller. As a result, over ten different predictive controllers, each with different properties, have been proposed in the literature over the last decade. Some well-known predictive controllers are GPC (Generalized Predictive Control) [55,61], DMC (Dynamic Matrix Control) [54], EPSAC (Extended Prediction Self-Adaptive Control) [62], PFC (Predictive Functional Control) [63], EHAC (Extended Horizon Adaptive Control) [64] and UPC (Unified Predictive Control) [65].
- In contrast to linear quadratic and pole-placement controllers, predictive controllers can also be derived for nonlinear processes. A nonlinear model of the process is then used explicitly to design the controller.
- Because predictive controllers make use of predictions, pre-scheduled reference trajectories (for example, used in robot control) or set points can be dealt with.

Unavoidably, predictive controller design has some drawbacks. Since predictive controllers belong to the class of model-based controller design methods a model of the process must be available. In general, in designing a control system two phases

can be distinguished: modeling and controller design. Predictive control provides only a solution for the controller design part. A model of the process must be obtained by other methods.

A second drawback is due to the fact that the predictive control concept is an open methodology. It has already been mentioned that, as a result of this, many different predictive controllers can be derived, each having different properties. Although, at first glance, the differences between these controllers seem rather small, these 'small' differences can yield very different behavior the closed-loop systems. As a result, it can be quite difficult to select which predictive controller must (or can) be used to solve a particular control problem. Sometimes, especially in the process industry, one cannot afford the expense designing a control system that one knows will not work in another process and whose cost cannot therefore be spread over a large number of applications. Therefore, a unified approach to predictive controller design is needed which allows treatment of each problem within the same framework and results in significant reduction in design costs. The resulting unified predictive controller unifies, among others, a number of well-known predictive controllers such as GPC, DMC, EPSAC and EHAC. Consequently, another advantage of the unified approach is that once this unified predictive controller is analyzed, conclusions can be drawn with respect to the above-mentioned predictive controllers.

In practice, MPC has proved to be a reasonable strategy for industrial control, in spite of the original lack of theoretical results at some crucial points such as stability or robustness.

5.1 Historical Perspective

From the end of the 1970's various articles appeared showing an incipient interest in MPC in the industry, principally the Richalet et al. publications [53,66] presenting Model Predictive Heuristic Control (MPHC) (later known as Model Algorithmic Control (MAC)) and those of Cutler and Ramakr [54] with Dynamic Matrix Control (DMC). A dynamic process model is explicitly used in both algorithms (impulse response in the first one and step response in the second) in order to predict the effect of the future control actions at the output; these are determined by

minimizing the predicted error subject to operational restrictions. The optimization is repeated at each sampling period with up to date information about the process. These formulations were heuristic and algorithmic and took advantage of the increasing potential of digital computers at the time.

These controllers were closely related to the minimum time optimal control problem and to linear programming. The receding horizon principle, one of the central ideas of MPC, was proposed by Propoi as long ago as 1963 [67], within the frame of “open-loop optimal feedback”, which was extensively dealt with in the seventies.

MPC quickly became popular particularly in chemical process industries, due to the simplicity of the algorithm and to the use of the impulse or step response model which, although possessing many more parameters than the formulations in the state space or input-output domain, is usually preferred as being more intuitive and requiring less a priori information for its identification. The majority of these applications were carried out on multivariable systems including constraints. In spite of this success, these formulations lacked formal theories providing stability and robustness results; in fact, the finite-horizon case seemed too difficult to analyze apart from in very specific cases.

Another line of work arose independently around adaptive control ideas, developing strategies essentially for monovariable processes formulated with input/output models. Peterka's Predictor-Based Self-Tuning Control [68] can be included here, which was designed to minimize, for the most recent predicted values, the expected value of a quadratic criterion on a given control horizon (finite or asymptotically infinite) or Ydstie's Extended Horizon Adaptive Control (EHAC) [64]. This method tries to keep the future output (calculated by a Diophantine equation) close to the reference at a period of time after the process delay and permits different strategies. Extended Prediction Self Adaptive Control (EPSAC) by De Keyser et al. [62] proposes a constant control signal starting from the present moment while using a sub-optimal predictor instead of solving a Diophantine equation. Generalized Predictive Control (GPC) developed by Clarke et al. in 1987 [55] also appears within this context. This uses ideas from Generalized Minimum Variance and is perhaps the most popular method at the moment. There are numerous predictive controller formulations based on the same common ideas, amongst which can be included:

Multistep Multivariable Adaptive Control (MUSMAR) [69], Multipredictor Receding Horizon Adaptive Control (MURHAC) [70], Predictive Functional Control (PFC) [64] or Unified Predictive Control (UPC) [65].

MPC has also been formulated in the state space context [72]. This not only allows for the use of well-known theorems of the state space theory, but also facilitates their generalization to more complex cases such as systems with stochastic disturbances and noise in the measured variables. By extending the step response model and using known state estimation techniques, processes with integrators can also be treated. The state estimation techniques arising from stochastic optimal control can be used for predictions without adding additional complications [72]. This perspective leads to simple tuning rules for stability and robustness: the MPC controller can be interpreted to be a compensator based on a state observer and its stability, performance and robustness are determined by the poles of the observer (which can be directly fixed by adjustable parameters) and the poles of the regulator (determined by the horizons, weightings, etc.).

Although the first works on GPC [73] proved some specific stability theorems using state-space relationships and studied the influence of filter polynomials on robustness improvement, the original lack of general stability results for finite horizon receding controllers was recognized as a drawback. Because of this, a fresh line of work on new predictive control methods with guaranteed stability appeared in the nineties. Two methods: (Clarke and Scattolini [74] and Mosca et al. [75]) were independently developed and were proved to be stable by imposing end-point equality constraints on the output after a finite horizon. Bearing in mind the same objective, Kouvaritakis et al. [76] presented stable GPC, a formulation, which guarantees closed-loop stability by stabilizing the process prior to the minimization of the objective function.

New results have also been obtained by using robust control design approaches by Campo and Morari [77] and Allwright [78]. The key idea is to take into account uncertainties about the process in an explicit manner and to design MPC in order to optimize the objective function for the worst situation of the uncertainties. These challenging results allow one to think that MPC will experience an even greater dissemination both in the academic world and within the control practitioner community.

5.2 MPC Strategy

Figure 5.1 illustrates the key features of a moving horizon control algorithm. Note that all the previous controlled variable (output) and manipulated variable (input) values are fixed and known. The manipulated variable moves and the resulting controlled variable values into the future remain unknown at this point. The moving horizon controller chooses the future manipulated variable values in order to regulate the controlled variable to its setpoint using the step response model and the previous inputs.

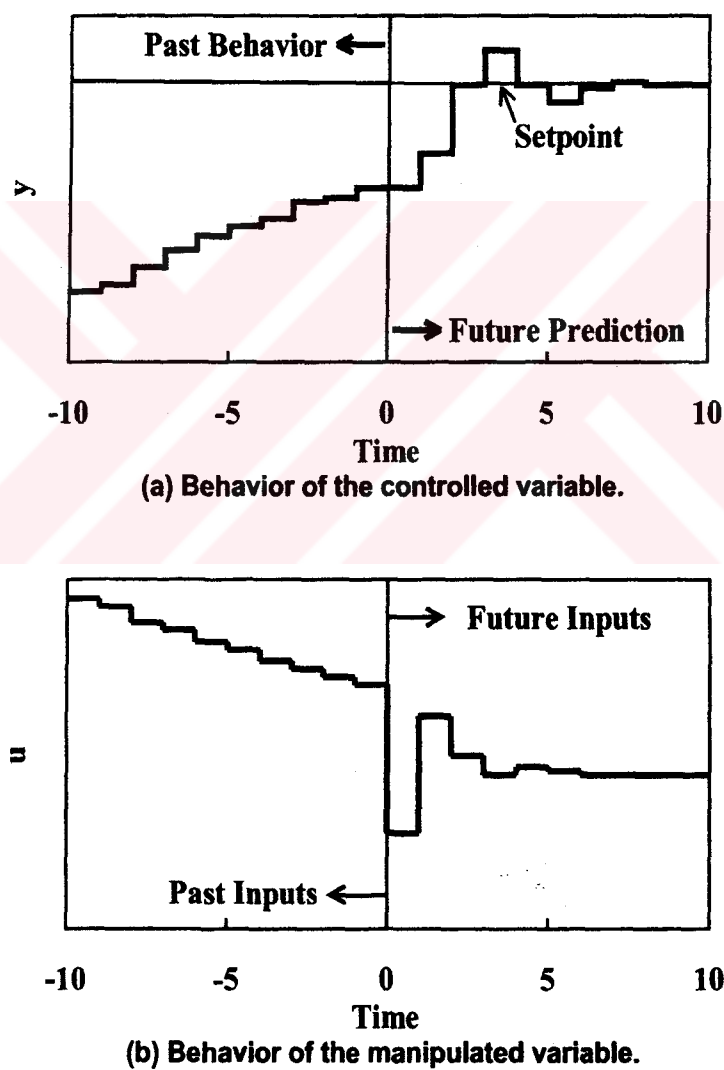


Figure 5.1. Controlled and manipulated variables for a moving horizon controller.

After one control interval has expired, a new controlled variable value is available as well as the last change in manipulated variable value. At the time $k+1$, the computation is repeated with the horizon moved by one time interval and the controller recalculates the sequence of manipulated variable values into the future to meet the control objective. In this manner, even though the complete sequence of control moves into the future is calculated at each control interval, only the first move is actually implemented before a new sequence of inputs is determined. The key feature of this approach is that at each control interval a sequence of control moves into the future is considered as well as the previous input sequence when determining the next change in the manipulated variable value.

In order to implement this strategy, the basic structure shown in figure 5.1 is used. A model is used to predict the future plant outputs, based on past and current values and on the proposed optimal future control actions. These actions are calculated by the optimizer taking into account the cost function (where the future tracking error is considered) as well as the constraints.

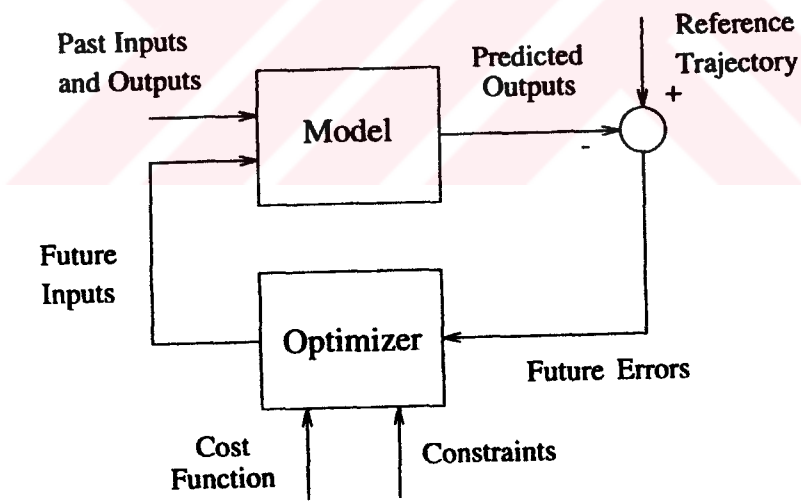


Figure 5.2. Basic structure of MPC.

The process model plays, in consequence, a decisive role in the controller. The chosen model must be capable of capturing the process dynamics so as to precisely predict the future outputs as well as being simple to implement and to understand. As MPC is not a unique technique but a set of different methodologies, there are many types of models used in various formulations.

One of the most popular in industry is the Step Response Model, which is very simple to obtain, as it only needs the measurement of the output when the process is excited with a step input. It is widely accepted in industrial practice because it is very intuitive and can also be used for multivariable processes, although its main drawbacks are the large number of parameters needed. Closely related to this kind of model is the Truncated Impulse Response Model, obtained when the input is an impulse.

The Transfer Function Model is, perhaps, most widespread in the academic community and is used in most control design methods, as it is a representation that requires only a few parameters and is valid for all kind of processes. The State-Space Model is also used in some formulations, as it can easily describe multivariable processes.

The optimizer is another fundamental part of the strategy as it provides the control actions. If the cost function is quadratic, its minimum can be obtained as an explicit function (linear) of past inputs and outputs and the future reference trajectory. In the presence of inequality constraints the solution has to be obtained by more computationally taxing numerical algorithms. The size of the optimization problems depends on the number of variables and the prediction horizons used and usually turn out to be relatively modest optimization problems, which do not require sophisticated computer codes to be solved. However the amount of time needed for the constrained and robust cases, it can be various orders of magnitude higher than that needed for the unconstrained case and the bandwidth of the process to which constrained MPC can be applied is considerably reduced.

5.3 Discrete Time Step Response Models

Model predictive control (MPC) is the most widely used form of multivariable control. It has been estimated that there are more than 3,000 industrial MPC applications worldwide. Dynamic Matrix Control (DMC) is the most popular form of MPC. This section will consider DMC applied to a SISO process and show how it can be extended to MIMO process [79]. DMC uses discrete time step response model of the process to calculate control action. The generalized discrete time step response

model of a process can be obtained from a step test using the following equation

$$a_i = \frac{y'(t_i)}{\Delta u(t_0)} \quad (5.1)$$

assuming that the process is at steady state at $t = t_0$ and a single step input change is made at $t = t_0$. Note that a_i is called a step response coefficient. The equation for the step response model can be obtained by rearranging equation (4.1):

$$y'(t_i) = y(t_i) - y(t_0) = a_i \Delta u(t_0) \quad (5.2)$$

Due to the flexible form of equation (5.1), a wide range of complicated dynamic behavior (e.g., inverse action) can easily be represented using this approach.

5.4 The Dynamic Matrix Equation

The Dynamic Matrix equation (5.2) can be used to predict the behavior of $y(t)$ for a series of $\Delta u(t)$ moves by applying the Principle of Superposition. The Principle of Superposition states that the total effect of a number of $\Delta u(t)$ moves on $y(t)$ is equal to the sum of the effect of each individual $\Delta u(t)$. Assume that $y(t_0)$ is known and $y(t)$ is at steady state at $t = t_0$ and that a series of $\Delta u(t_i)$ moves are made into the future, then using equation (5.2) applying the principle of superposition :

$$y(t_1) - y(t_0) = a_1 \Delta u(t_0)$$

$$y(t_2) - y(t_0) = a_2 \Delta u(t_0) + a_1 \Delta u(t_1)$$

$$y(t_3) - y(t_0) = a_3 \Delta u(t_0) + a_2 \Delta u(t_1) + a_1 \Delta u(t_2)$$

⋮

$$y(t_n) - y(t_0) = a_n \Delta u(t_0) + a_{n-1} \Delta u(t_1) + a_{n-2} \Delta u(t_2) + \dots \quad (5.3)$$

For example, consider the equation for $y'(t_3)$. The contribution of $\Delta u(t_0)$ to $y(t_3)$ uses a_3 when applying equation (5.2). In order to consider the effect of $\Delta u(t_1)$, equation (5.2) must be time shifted backwards, ΔT_s ; therefore, the coefficient applied to $\Delta u(t_1)$ is a_2 . Another way to consider this problem is that since $\Delta u(t_1)$ is applied at $t = t_1$, using equation (5.2) to model the effect of $\Delta u(t_1)$ requires that i in equation (5.2) be set equal to 2. Then similarly, to calculate the effect of $\Delta u(t_2)$ on $y(t_3)$, a_1 should be used. Equation (5.3) can be more compactly expressed by :

$$y'(t_n) = y(t_n) - y(t_0) = \sum_{i=1}^n a_i \Delta u(t_{n-i}) \quad (5.4)$$

Equation (5.3) or (5.4) can be put into matrix form. Consider the case in which U moves in $\Delta u(t_i)$ are made into the future and the step response models have m coefficients (a_i 's). Then from equation (5.3),

$$\begin{bmatrix} y'(t_1) \\ y'(t_2) \\ y'(t_3) \\ \vdots \\ y'(t_n) \end{bmatrix} = \begin{bmatrix} a_1 & 0 & 0 & \cdots & 0 \\ a_2 & a_1 & 0 & & \\ a_3 & a_2 & a_1 & & \\ \vdots & \vdots & & & \\ a_m & a_m & a_m & \cdots & a_1 \end{bmatrix} \begin{bmatrix} \Delta u(t_0) \\ \Delta u(t_1) \\ \Delta u(t_2) \\ \vdots \\ \Delta u(t_{U-1}) \end{bmatrix} \quad (5.5)$$

assuming that $n > m$. n is the prediction horizon or the number of ΔT_s steps into the future for which the model is used to predict the behaviour of the controlled variable, m is the model horizon or the number of ΔT_s steps used by the step response model. U is the control horizon or the number of $\Delta u(t)$ moves on $y(t)$. Note that a_i for $i > m$ is equal to a_m . Equation (5.5) can also be written as :

$$y' = A \Delta u \quad (5.6)$$

The matrix A is calculated from the coefficients of the step response model and is dimensioned (n, U) . There are n rows and U columns, and is called the dynamic matrix. If $n = U$, the matrix is square. The dynamic matrix can be used to calculate the dynamic behavior of the process in response to future changes in $\Delta u(t_i)$. The value of equation (5.6) stems from the fact that it can be used to calculate the dynamic behavior of y for a series of input changes.

5.5 Prediction Vector

Up until this point, we have assumed that $y(t_0)$ was at steady state and that manipulated variable changes are made only for $t \geq t_0$. For a control application, this assumption is not realistic since manipulated variable changes at $t < t_0$ are likely to exist as shown in figure (5.3). As a result, the effect of the previous input changes ($\Delta u(t)$ for $t < t_0$) must be taken into account in order to properly model the future behavior of the controlled variable ($y(t)$ for $t > t_0$).

The prediction vector, y^p , contains the values of $y(t)$ for $t > t_0$ if no future manipulated variable changes are made ($\Delta u(t) = 0$ for $t \geq t_0$). The prediction vector contains the effects of previous manipulated variable changes on future controlled variable values.

Assume that the process has a model horizon, m and $n=U$. That is, after m time steps, an input change has had its total steady-state effect on the process.

Applying equation (5.4) to calculate the prediction vector at $t = t_1$ results in,

$$y^p(t_1) = y(t_{-m}) + a_{m+1} \Delta u(t_{-m}) + a_m \Delta u(t_{-m+1}) + a_{m-1} \Delta u(t_{-m+2}) \\ + \dots + a_3 \Delta u(t_{-2}) + a_2 \Delta u(t_{-1}) + a_1 \Delta u(t_0)$$

where the negative subscripts indicate the number of sampling intervals before t_0 and assuming that the process is at steady state at $t = t_{-m}$. Also note that the coefficients of $\Delta u(t_{-m})$ and $\Delta u(t_{-m+1})$ are both a_m since $a_{m+1} = a_m$. Also, $\Delta u(t_0)$ is zero for the prediction vector. Therefore,

$$y^p(t_1) = y(t_{-m}) + a_m \Delta u(t_{-m}) + a_m \Delta u(t_{-m+1}) + a_{m-1} \Delta u(t_{-m+2}) \\ + \dots + a_3 \Delta u(t_{-2}) + a_2 \Delta u(t_{-1})$$

Likewise, $y^p(t_2)$ is given by :

$$y^p(t_2) = y(t_{-m}) + a_m \Delta u(t_{-m}) + a_m \Delta u(t_{-m+1}) + a_m \Delta u(t_{-m+2}) + a_{m-1} \Delta u(t_{-m+3}) \\ + \dots + a_4 \Delta u(t_{-2}) + a_3 \Delta u(t_{-1})$$

In this manner, $y^p(t_n)$ is given by :

$$y^p(t_n) = y(t_{-m}) + a_m \Delta u(t_{-m}) + a_{m-1} \Delta u(t_{-m+1}) + \dots + a_2 \Delta u(t_{-2}) + a_1 \Delta u(t_{-1}) \quad (5.7)$$

where n is the number of ΔT_s moves into the future that are modeled and where $n > m$. The prediction vector, y^p , can be expressed in matrix form,

$$\begin{bmatrix} y^p(t_1) \\ y^p(t_2) \\ \vdots \\ y^p(t_n) \end{bmatrix} = \begin{bmatrix} y(t_{-m}) \\ y(t_{-m}) \\ \vdots \\ y(t_{-m}) \end{bmatrix} + \begin{bmatrix} a_m & a_m & a_{m-1} & a_{m-2} & \dots & a_3 & a_2 \\ a_m & a_m & a_m & a_{m-1} & \dots & a_4 & a_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_m & a_m & a_m & a_m & \dots & a_m & a_m \end{bmatrix} \begin{bmatrix} \Delta u(t_{-m}) \\ \Delta u(t_{-m+1}) \\ \vdots \\ \Delta u(t_{-1}) \end{bmatrix}$$

or

$$y^p = y(t_{-m}) + A^p \Delta u^p \quad (5.8)$$

where $y(t_{-k})$ is the value of the controlled variable at $t = t_0 - k \Delta T_s$. Then the values of $y(t)$ for $t > t_0$ can be calculated by combining the prediction vector with the effects of future control moves (5.6).

$$y = y^p + A \Delta u \quad (5.9)$$

Equation (5.9) is subject to a number of factors that undermine its accuracy:

(1) errors in identifying the coefficients of the discrete time step response model, (2) unmeasured disturbances, (3) nonlinear behavior, and (4) not a steady-state process at $t = t_{-m}$. If equation (5.9) were used for control, offset would result due to these sources of process/model mismatch.

Using equation (5.8), $y^p(t_0)$ is calculated by

$$y^p(t_0) = y(t_{-m}) + a_m \Delta u(t_{-m}) + a_{m-1} \Delta u(t_{-m+1}) + \dots + a_2 \Delta u(t_{-2}) + a_1 \Delta u(t_{-1})$$

5.6 The Dynamic Matrix Control Algorithm

The control law using dynamic matrix control (DMC) algorithm is based on minimizing the objective function [80,81]. The objective function, Ψ , is the sum of the square of the errors between setpoint values and predicted values of iron oxide and lime module (n steps into the future) of the raw meal.

$$\Psi = \sum_{i=1}^n [y_{sp} - y(t_i)]^2 \quad (5.10)$$

Equation (5.9) shows that $y(t_i)$ is made up of two parts: the prediction vector (the effect of past inputs) and the effects of future inputs. Note that only the effect of future moves can be changed by the controller; therefore, combining y_{sp} , and y^p into :

$$E(t_i) = y_{sp} - y^p(t_i) \quad (5.11)$$

results in,

$$\Psi = \sum_{i=1}^n [E(t_i) - y_c(t_i)]^2 \quad (5.12)$$

where

$$y_c = A \Delta u \quad (5.13)$$

The objective of the controller is to choose the control moves, $\Delta u(t_i)$ for U moves into the future such that Ψ is minimized.

Perfect control (i.e., $\Psi = 0$), which is based upon assuming that $y_c(t_i)$ is the mirror image $E(t_i)$, is given by :

$$\Delta u = A^{-1} E \quad (5.14)$$

But this result is not realistic since it assumes that y_c can be moved instantaneously. In addition, equation (5.14) is valid only when the number of dependent variable is equal to the number of input variables. Instead, we can choose the set of control moves that minimizes the sum of the squares of the errors from setpoint. This solution can be obtained analytically by differentiating equation (5.9) with respect to Δu and setting the result equal to zero,

$$\frac{\partial \Psi}{\partial \Delta u} = A^T (A \Delta u - E) = 0 \quad (5.15)$$

Solving for Δu :

$$\Delta u = (A^T A)^{-1} A^T E = K_{MPC} E \quad (5.16)$$

which is the control law for the controller. $(A^T A)^{-1} A^T$ is the pseudoinverse matrix, K_{MPC} is called the feedback gain matrix and E is the predicted value of the process error.

This equation does not assume instantaneous changes in y_c and can be applied in cases in which the number of dependent variables is different from the number of independent variables. Note that $(A^T A)^{-1} A^T$ is equal to A^{-1} when A is a square matrix. Therefore, once the step response model and the prediction vector are calculated, the controller can be formulated directly using equation (5.16).

Equation (5.16) results in very aggressive control because it is based on minimizing the deviation from setpoint without regard to the changes in the manipulated variable levels. That is, if equation (5.16) is applied, excessively sharp changes in u will result which is not desirable operationally. Normal levels of process/model mismatch combined with the aggressive nature of equation (5.16) can easily yield unstable control performance. In addition, $(A^T A)^{-1}$ can be ill-conditioned due to process/model mismatch and deadtime in the process model. These problems can be overcome by adding the diagonal matrix, W_2^{-2} , to $A^T A$ in equation (5.16) resulting in the following:

$$\Delta u = (A^T A + W_2^{-2})^{-1} A^T E \quad (5.17)$$

where W_2 is the move suppression weighting matrix and is a diagonal matrix with positive elements,

$$W_2 = \begin{bmatrix} w_2 & 0 & 0 & \cdots & 0 \\ 0 & w_2 & 0 & \cdots & 0 \\ 0 & 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \cdots & w_2 \end{bmatrix} \quad (5.18)$$

The larger the value of w , the more Ψ is penalized for changes in the manipulated variable; therefore, w can be used as a tuning parameter and it will determine the aggressiveness of the controller. In general, the more nonlinear a process and the larger the magnitude of the disturbances, the larger the value of w that should be used.

5.7 Extension to Multivariable Processes

For a multi-input multi output (MIMO) process with n_u inputs and n_y outputs can be expressed in matrix form,

$$\begin{bmatrix} y'_1(t_1) & y'_2(t_1) & \cdots & y'_{n_y}(t_1) \\ y'_1(t_2) & y'_2(t_2) & \cdots & y'_{n_y}(t_2) \\ y'_1(t_3) & y'_2(t_3) & \cdots & y'_{n_y}(t_3) \\ \vdots & \vdots & & \vdots \\ y'_1(t_n) & y'_2(t_n) & \cdots & y'_{n_y}(t_n) \end{bmatrix} = \begin{bmatrix} a_{1,1,i} & a_{1,2,i} & \cdots & a_{1,n_u,i} \\ a_{2,1,i} & a_{2,2,i} & \cdots & a_{2,n_u,i} \\ \vdots & \vdots & & \vdots \\ a_{n_y,1,i} & a_{n_y,2,i} & \cdots & a_{n_y,n_u,i} \end{bmatrix} \begin{bmatrix} \Delta u_1(t_0) & \Delta u_2(t_0) & \cdots & \Delta u_{n_u}(t_0) \\ \Delta u_1(t_1) & \Delta u_2(t_1) & \cdots & \Delta u_{n_u}(t_1) \\ \vdots & \vdots & & \vdots \\ \Delta u_1(t_{U-1}) & \Delta u_2(t_{U-1}) & \cdots & \Delta u_{n_u}(t_{U-1}) \end{bmatrix} \quad (5.19)$$

The step response coefficients matrix is given by :

$$A_i = \begin{bmatrix} a_{1,1,i} & a_{1,2,i} & \cdots & a_{1,n_u,i} \\ a_{2,1,i} & a_{2,2,i} & \cdots & a_{2,n_u,i} \\ \vdots & \vdots & & \vdots \\ a_{n_y,1,i} & a_{n_y,2,i} & \cdots & a_{n_y,n_u,i} \end{bmatrix} \quad (5.20)$$

where a_{kji} is the i^{th} step response coefficient showing the effect of input j on output k .

An additional issue that must be addressed when applying the controller to MIMO processes is how to prioritize the various control objectives. Controllers use controlled variable weighting, which allows the user to assign a relative weighting to each of the controlled variables. A controlled variable weighting matrix, W_1 , is added to the DMC control law (5.17).

$$\Delta u = (A^T W_1^2 A + W_2^2)^{-1} A^T W_1^2 E = K_{MPC} E \quad (5.21)$$

where K_{MPC} is called the feedback gain matrix, W_1 and W_2 are diagonal weighting matrices and E is the predicted value of the process error.

5.8 Tuning Predictive Controller

The predictive control technique presented in the previous section includes a number of design parameters, which can be adjusted to give the desired response as well as an appropriate amount of controller effort. These parameters include :

m = model horizon

U = control horizon

n = prediction horizon

W_1 = weighting matrix for predicted errors

W_2 = weighting matrix for control moves

Δt = sampling period

The model horizon m should usually be selected so that $m \Delta t \geq$ open-loop settling time, which is equal to the time for the open-loop step response to be 99 % complete (95 %, in some cases). In the case of integrating processes, some modifications in the approach need to be made.

Parameter n is the number of predictions that are used in the optimization calculations. It is also the dimension of the gain vector K_{MPC} . Increasing n results in more conservative control action, which has a stabilizing effect but also increase the

computational effort. When $n = d + 1$ (d = time delay), a minimal prototype controller results.

The control horizon U is the number of future control actions that are calculated in the optimization step to reduce the predicted errors. Therefore the computational effort increases as U is increased. A suitable first guess is to choose U so that $U \Delta t \cong t_{60}$, which is the time for the open-loop response to be 60% complete. For $U > 5$, the value of n is not crucial since only the first move is implemented. However, too large a value of u results in excessive control action. A smaller value of U leads to a robust controller that is relatively insensitive to model errors.

The weighting matrices W_1 and W_2 contain a potentially large number of design parameters. However, it is usually sufficient to select $W_1 = I$ and $W_2 = f I$ (I is the identity matrix and f is a scalar design parameter). Larger values of f penalize the magnitude of Δu more, thus giving less vigorous control. When $f = 0$, the controller gains are very sensitive to U , largely because of the ill-conditioning of $A^T A$, and U must be made small.

The sampling period Δt must also be selected as part of the design procedure. It should be small enough to ensure that important dynamic information is not lost. On the other hand, if Δt is too small, then, as discussed above, m must be made very large, which is undesirable. If Δt is smaller than the time delay, it causes m to be rather large. The sampling period is not considered to be a tuning parameter because adjustment of U , n , and f is usually quite sufficient to obtain satisfactory closed-loop performance. However, it may be helpful to check the sensitivity of the response to Δt .

6. MODEL PREDICTIVE CONTROL (MPC) OF THE RAW MATERIAL BLENDING PROCESS

6.1 Raw Meal Blending Control Problem

In the raw material blending process, the control problem arises from the fact that the chemical compositions of the various raw materials vary from time to time and they are not measured. Furthermore, each of the raw materials contains varying amount of the constituent oxides. Only annual or monthly average values of raw material compositions are available and fluctuation of the actual raw material compositions from these long-term average compositions introduces disturbances to be eliminated by computer control. Therefore, the purpose of the control strategy is to calculate the optimal feed ratios of the raw materials to achieve the target value of the chemical compositions (oxide or module values) of the raw meal despite disturbance as the same meaning of penalizing the deviations of the module or oxide percentage from the desired values in cement industry. In the thesis, multivariable model predictive controller, using discrete time step response model and dynamic matrix control (DMC) algorithm have been designed to achieve desired control performance. Since predictive controllers belong to the class of model-based controller design methods a model of the process must be available. In general, in designing a control system two phases can be distinguished: modeling and controller design. Predictive control provides only a solution for the controller design part. A model of the process must be obtained by other methods.

6.2 Prediction Models

In the fourth section, developed three raw material blending process models had been presented and unknown plant parameters had been estimated. The block diagram of closed-loop state space models in the MPC is given by figure (6.1).

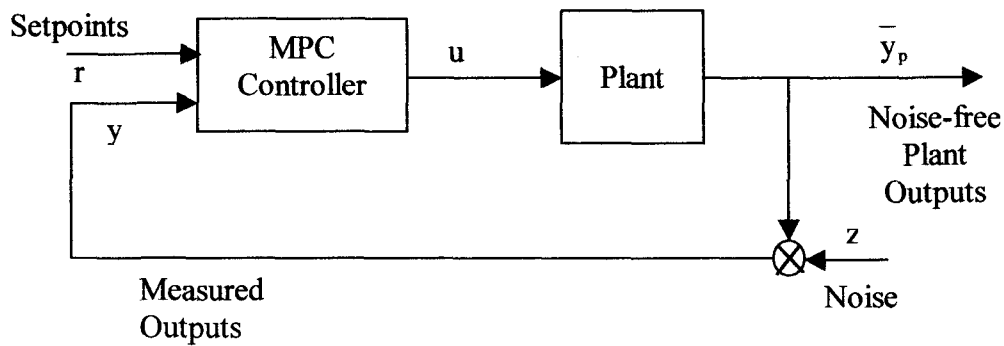


Figure 6.1. The block diagram of closed-loop state space models in the MPC.

In the figure (6.1), z is the noise representing modelling errors (white noise). The variable $\bar{y}_p(k)$ represents the plant output before the addition of the noise. The state space model equations for the plant are given by :

$$x(t+1) = F x(t) + G u(t) \quad (6.1)$$

$$Y(t) = H x(t) + D u(t) + z(k) \quad (6.2)$$

where F , G , H , and D are the system parameter matrices, u is the input vector representing raw mix proportions (low grade, iron ore) and y is the output vector representing iron oxide or/and lime module values of the raw meal).

The state space model has been turned into the step response model for predictive controller to calculate control actions in the computer program (MATLAB). Dynamic Matrix Equation for process model with two inputs and two outputs, which is obtained the step response model, can be expressed in matrix form :

$$\begin{bmatrix} y'_1(t_1) & y'_2(t_1) \\ y'_1(t_2) & y'_2(t_2) \\ y'_1(t_3) & y'_2(t_3) \\ \vdots & \vdots \\ y'_1(t_n) & y'_2(t_n) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \Delta u_1(t_0) & \Delta u_2(t_0) \\ \Delta u_1(t_1) & \Delta u_2(t_1) \\ \Delta u_1(t_2) & \Delta u_2(t_2) \\ \vdots & \vdots \\ \Delta u_1(t_{U-1}) & \Delta u_2(t_{U-1}) \end{bmatrix} \quad (6.3)$$

where n is the prediction horizon and U is the control horizon. The step response coefficients matrix is given by :

$$A_1 = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix} \quad (6.4)$$

where A_{11} is the dynamic matrix showing the effect of input u_1 on output y_1 and A_{12} is the dynamic matrix showing the effect of input u_2 on output y_1 , etc. The generalized discrete time step response model of a process is given by :

$$y = y^p + A \Delta u \quad (5.9)$$

Equation (5.9) shows that $y(t_i)$ is made up of two parts: the prediction vector (the effect of past inputs) and the effects of future inputs.

6.3 Objective Function

The objective function, Ψ , is the sum of the square of the errors between setpoint values and predicted values of iron oxide and lime module (n steps into the future) of the raw meal.

$$\Psi = \sum_{i=1}^n [y_{sp} - y(t_i)]^2 \quad (5.10)$$

6.4 Control Law for the Raw Material Blending Process

The control law is based on minimizing the objective function. The purpose of the controller is to choose the control moves $\Delta u(t_i)$ for U moves into the future such that the objective function is minimized. Thus, model predictive controller deals with not only the quality control but quantity control as well. This is accomplished by minimizing quadratic objective function consisting of the energy term (Δu^2). The control law using dynamic matrix control algorithm is obtained as follows :

$$\Delta u = (A^T W_1^2 A + W_2^2)^{-1} A^T W_1^2 E = K_{MPC} E \quad (5.21)$$

where W_1 and W_2 are augmented diagonal weighting matrices, which contains diagonal matrices and K_{MPC} is called the feedback gain matrix while E is the vector of the predicted future errors over the horizon n .

6.5 Simulation Results

Unconstrained multivariable model predictive controller has been designed for three different raw material blending process models. While the figures of simulation works depending on design parameters for model 1 have been given as follows, for two other models have been given in appendix section. The effects of design parameters have been explained on each figure. Iron oxide and lime module reference values and total flow rate are $IO=2.7(\%)$, $ML=94(\%)$ and $\bar{w}_r=500$ kg/min for Nuh Cement Factory respectively. Daily production is about 9000 ton/day in Nuh Cement Factory. In identification section, because the mean value of input/output data has been subtracted from the real data for elimination of the DC component, the new setpoints for iron oxide and lime module are $IO=0.14(\%)$ and $ML=0.25(\%)$ respectively. The control scheme has been programmed using MATLAB. At the end of the control, the mean value ($u_1=60.24(\%)$ and $u_2=3.972(\%)$) of the raw mix proportions were added to simulation values and the optimal raw mix proportions have been found as $u_1(\text{low grade})=59.6(\%)$ and $u_2(\text{iron ore})=3.642(\%)$ for model 1, $u_1(\text{low grade})=60.14(\%)$ and $u_2(\text{iron ore})=6.57(\%)$ for model 2 and $u_1(\text{low grade})=57.54(\%)$ and $u_2(\text{iron ore})=5.97(\%)$ for model 3.

At the beginning of the simulation works, without any adjustment of the design parameters, which are prediction horizon, control horizon and weighting matrices, input-output values of controlled process for model 1 are given in figure (6.2):

$W_1 = \text{unity}$; weight on output

$W_2 = [0.02 \ 0.4]$; weights on both inputs

$n = 3$; for prediction horizon

$U = 3$; for control horizon

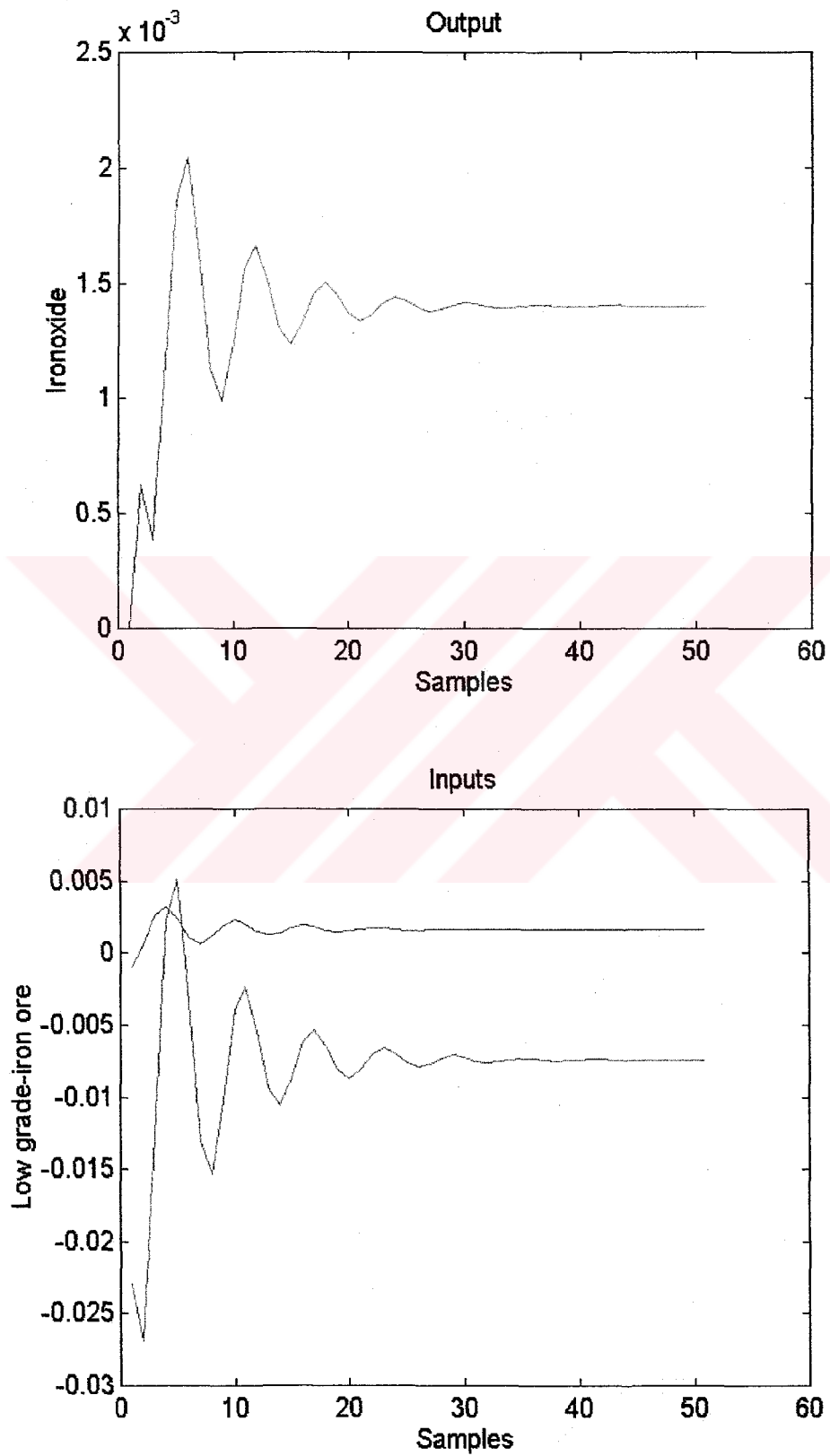


Figure 6.2. Input-output values of controlled process for model 1 at the beginning.

We have reached perfect tracking of the specified setpoint with adjusted design parameters for model 1 (MISO) :

$W_1 = \text{unity}$; weight on output $W_2 = [0.1 \ 0.15]$; weights on both inputs
 $n = 7$; for prediction horizon $U = 7$; for control horizon

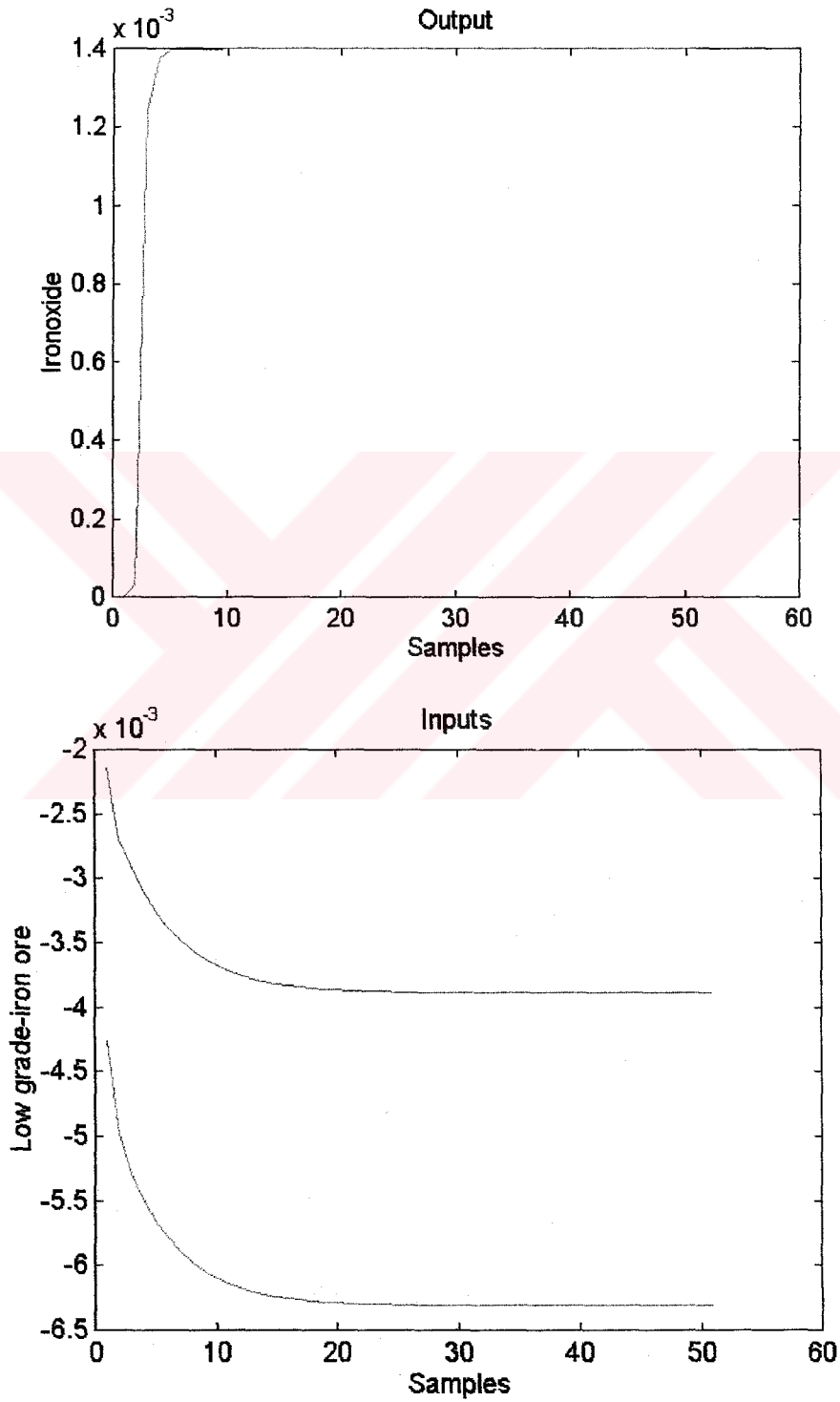


Figure 6.3. Inputs-output values of controlled process for model 1 (adjusted parameters).

If we make the prediction horizon significantly larger than the control horizon for $W_2=[0.1 \ 0.15]$, more sluggish servo response can be obtained, but it makes the control action less aggressive and tends to stabilize the system.

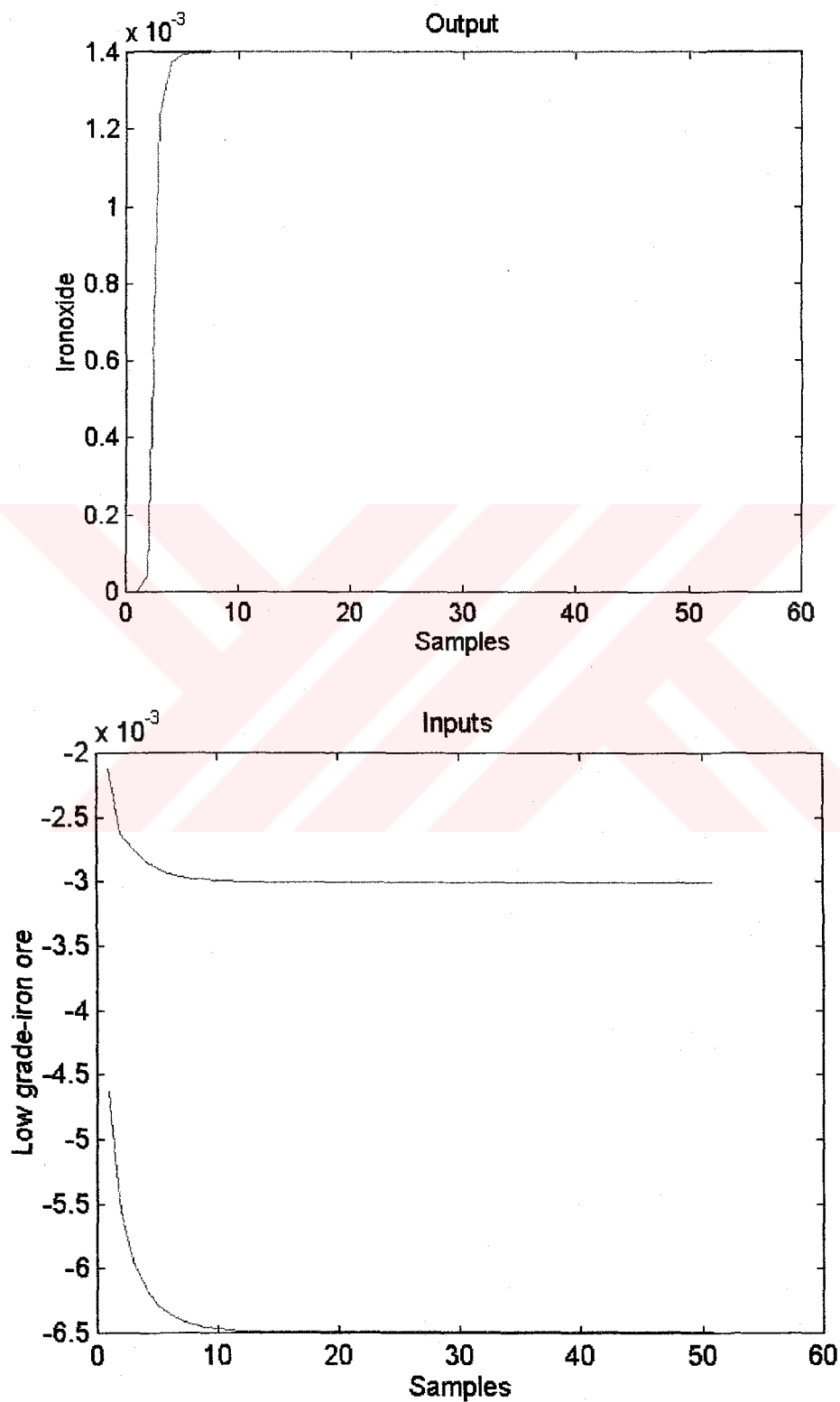


Figure 6.4. Input-output values of controlled process for model 1 ($n=10$, $U=3$).

If we increase the weights on the inputs for $n=10$ and $U=3$, more sluggish servo response can be obtained, but it makes the control action less aggressive and tends to stabilize the system.

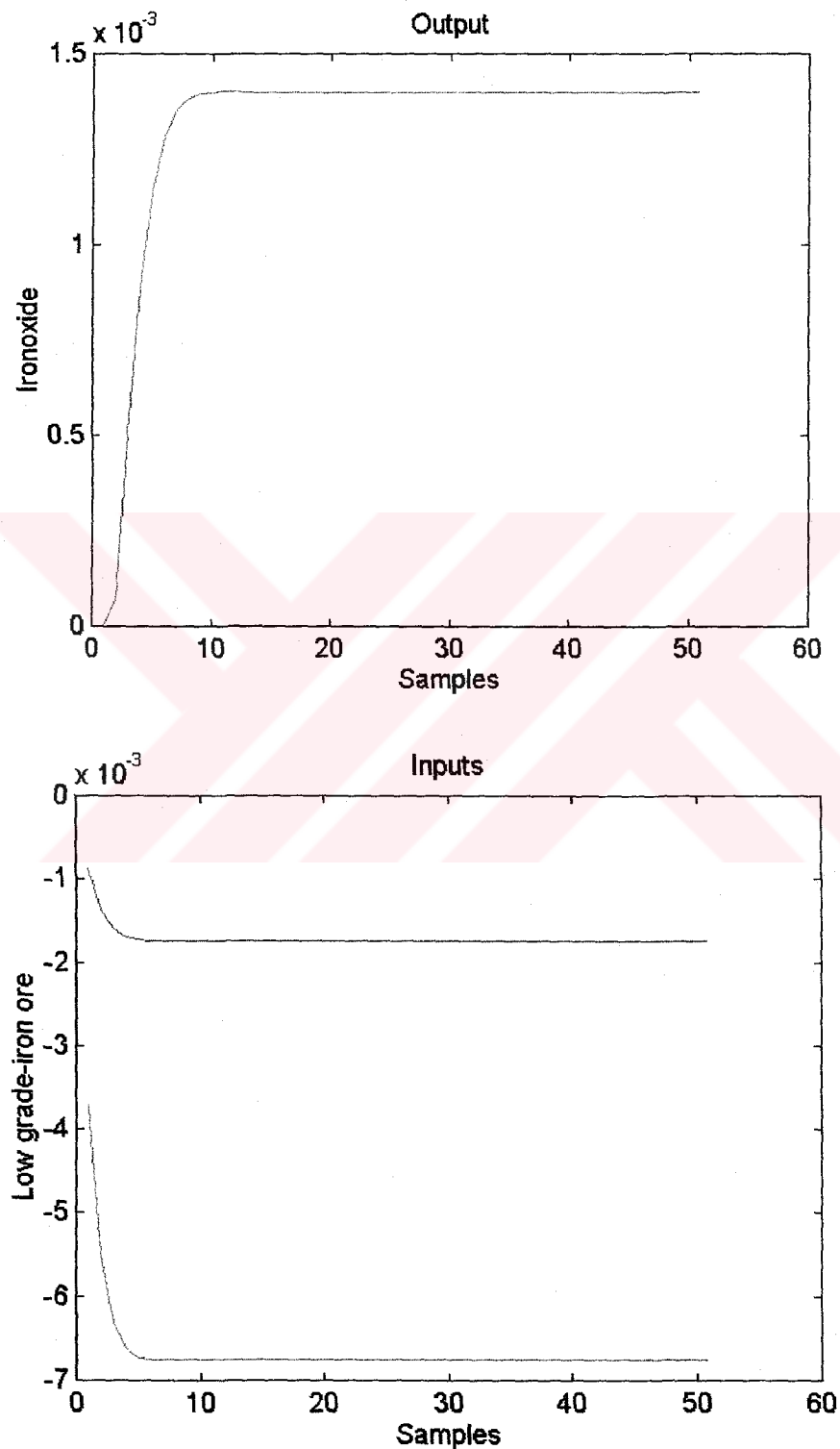


Figure 6.5. Input-output values of controlled process for model 1 ($W_2=[0.2 \ 0.9]$).

If we put the noise (white noise) to the system ($n=10$, $U=3$ and $W_2=[0.2 \ 0.9]$), the results show that the controller compensates the disturbances rapidly.

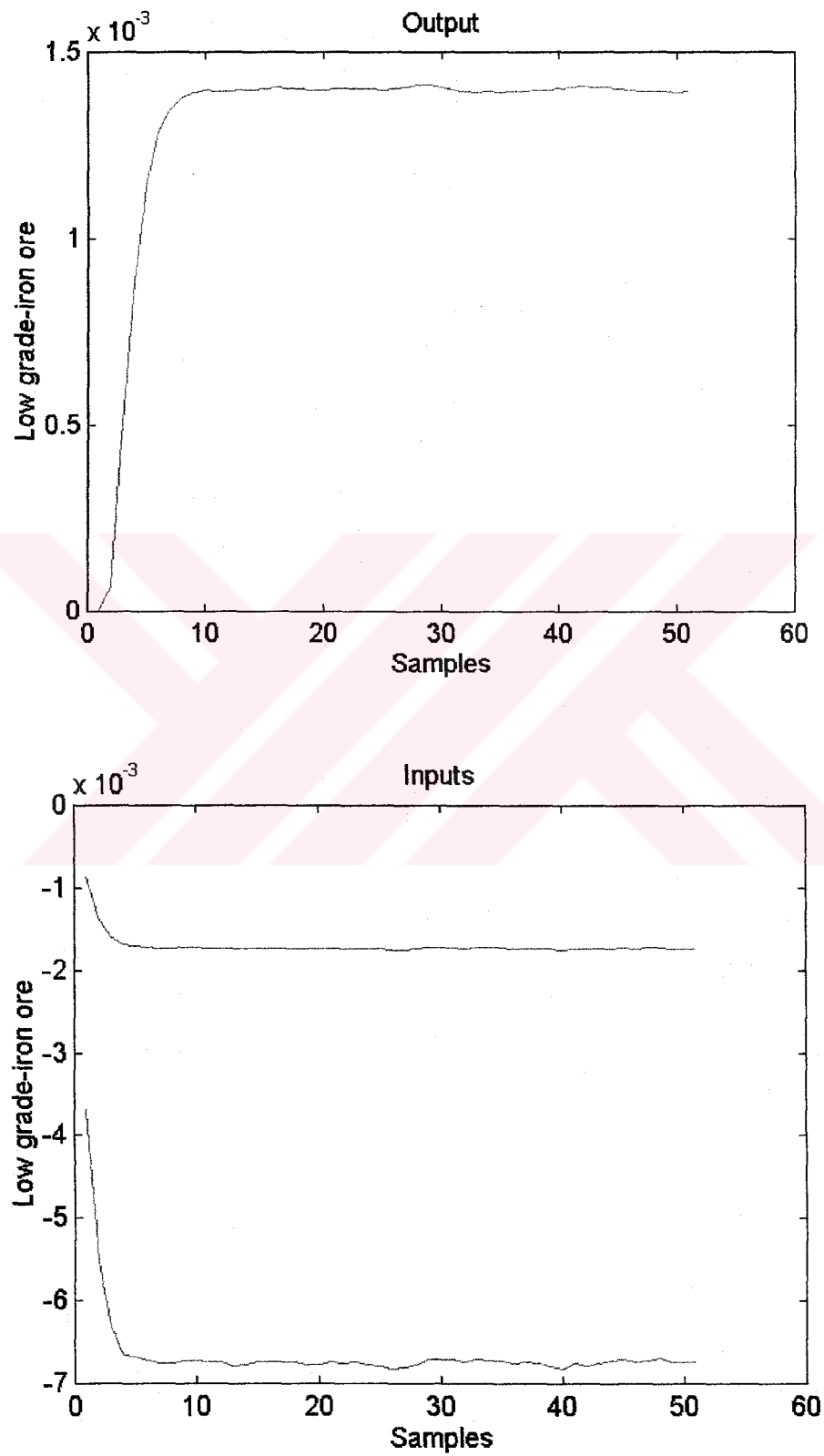


Figure 6.6. Input-output values of controlled process for model 1 with noise.

At the end of the control part, iron oxide and lime module values have been reached the specified setpoint values quickly. This is an important result pointing out the system performance. Furthermore, the sum of the raw material proportions, which are low grade, high grade and iron ore, must be equal to 100 as percentage in Nuh Cement Factory. In this study, the sum of the obtained optimal raw mix proportions for low grade and iron ore, which are two of three raw materials, is lower than this constrained value (%100). This shows the validity of developed models. Finally, applying this controller to process, important decrease has been obtained at the value of the iron ore in contrast to nominal proportion of the process in Nuh Cement Factory. In which, the value of iron ore determines the cost of the cement significantly.



7. CONCLUSION AND COMMENTS

Identification and model predictive control of the cement raw material blending process in cement factory have been examined in the thesis. The disturbances coming from the variations in chemical compositions of the raw material from long-term average compositions cause the changes of the system parameters. The process is multivariable and coupled one, because the feeder tanks do not contain chemically homogeneous raw materials. The time delays in the system are also considerable, despite the new PGNAA (prompt gamma neutron activation analyzer) technique applied in recent years for analyzing the chemical composition of the raw meal. Some commercial control packages are available today. They usually deal with only the quality control. In the mean time, finding the optimal setting of the parameters of the controllers, which are usually designed in cascade loops, is a very difficult and time-consuming task. As the best solution for the problem, stochastic multivariable dynamic models using a priori experimental knowledge of the system and characteristic of the disturbance have been constructed and then multivariable model based predictive controllers have been designed to achieve desired control performance. Proposed multivariable model based predictive controller in this study deals with not only the quality control but the quantity control as well.

In the identification section, experimental data has been collected from the controlled process under varying operating conditions in Nuh Cement Factory in Turkey. Good parametric multivariable models having minimum order and minimum number of parameters have been obtained. In which, the inputs are the feed ratios of the raw materials (low grade and iron ore) and the outputs are iron oxide or/and lime module values of the raw meal.

Developed multi input-multi output model takes into consideration the interactions of the outputs each other. This is an important advantage to increase control performance and also, this model have made possible the control of iron oxide and

lime module together instead of using parallel working single loop controllers. As another advantage with these developed models, the control strategy could have been designed without putting any constraints on input-output variables, which causes solving constrained optimization problem. Thus, it has been saved from computational effort and time. Furthermore, stable operation conditions have been obtained for the process with these models.

The control algorithm needs a limited and very intuitive knowledge of the control and tuning is relatively easy. It can be used to control of difficult processes such as processes with a large time delay and processes that are open loop unstable and non-linear. It deals with not only the quality control but the quantity control as well. This is accomplished by minimizing quadratic objective function consisting of the energy term (Δu^2).

In the control section, by tuning controller design parameters easily, in which are prediction horizon, control horizon and weighting matrices, we have obtained perfect controls giving fast response and perfect tracking compensating the disturbances rapidly for each model. In other words, the efficient raw feed proportion and significant decrease in the variance of controlled outputs has been provided at the end of this study.

Applying this controller to process, important decrease has been obtained at the value of the iron ore (%6 for MIMO model) in contrast to nominal proportion (%11) of the process in Nuh Cement Factory. The value of the iron ore determines the cost of the cement significantly. Also, the value of the high grade proportion, which is third feedstream and outside of the control, has been slowed down slightly from nominal proportion, which also means less cost.

One remark in this study, because the only lime module or iron oxide could have been controlled depending on choice in Nuh Cement Factory with available control package, two different multivariable model based predictive controllers have been designed for these two cases to see the differences of the control performance and then these outputs have been controlled together by using two input-two output process model.

Presented control strategy in this study has also the following main features:

- estimates the raw meal compositions using a priori stochastic dynamic model of the process
- take cares of the process dead time delays
- works in the condition of model uncertainty and unstable operation conditions
- makes effective noise reduction means robust control
- can be applied for any number of feeders
- is easy to implement in any new plant or any change of the raw materials.
- can be applied for all types of blending operations
- provides direct module control
- eliminates the problems caused by local controllers
- has optimal satisfaction of the technological requirements

Finally, we can say that, the simulation results and the main features of the presented control strategy are very encouraging for the application of the proposed study in the cement industry. One comment in this study, it has been seen from the simulation studies that the stable operation conditions have been provided after approximately ten minutes for the constant raw feed ratios. It means that, increasing the sampling period will improve the control performance of the process in Nuh Cement Factory. Another comment, oxide values of the raw meal has been measured from the input of the raw mill grinder since conveyor type system used in Nuh Cement Factory. The system performance of the process will be increased by measuring data from the output of the raw mill grinder because oxide composition of the raw meal in this site has a more homogenous structure. Some modifications are also possible in future works by providing more precise information for noise characteristics. Also as a next step of this study in the future, self-tuning model based predictive controller which adjusts the changes of parameters optimally will be designed and applied for on-line operation conditions in Nuh Cement Factory.

REFERENCES

- [1] Bay, T., Ross, C. W., Andrews J. C. and Gilliland, J. L., 1968. Dynamic control of the cement process with a digital computer system, *IEEE Transactions on Ind. & Gen. Appl.*, **4**, 294-298.
- [2] Hammer, H., 1972. Computer controlled raw meal production in cement industry, *Regelungstechnik*, **20**, 190-198.
- [3] Hawkins, J. C., 1974. Selecting control techniques for cement plants, *Proceedings of the IEEE Cement Industry Technical Conference*, Mexico City.
- [4] Lundan, A. and Mattila, O., 1974. System for the control of the homogenizations of the cement raw meal," *Proceedings of the 4th IFAC Symposium on Digital Computer Applications to Process Control*, Zurich, 424-435.
- [5] Westerlund, K. T., Toivenon, H. and Nyman, K. E., 1979. Stochastic modelling and self-tuning control of a continuous cement raw material mixing system, *Proceedings of the 18th IEEE Conference on Decision and Control*, Fort Lauderdale, Florida, 610-615.
- [6] Vajk, I., Keviczky, L., Hetthessy, J., Adrian F. J., Hilger, M., and Kolostori, J., 1991. Adaptive composition control, *IFAC Identification and System Parameter Estimation*, Budapest, Hungary, 511-515.
- [7] Hubbard, M. and Dasilva, T., 1982. Estimation of feedstream concentrations in cement raw material blending, *Automatica*, **18**, 595-606.
- [8] Westerlund, K. T., 1983. Real time optimization of raw materials in cement industry, *IFAC Automation in Mining, Mineral and Metal Processing*, Helsinki, Finland, 111-115.
- [9] Lin, P., Yun, Y. S., Barbier, J. P., Babey, P. H. and Prevot, P., 1991. Intelligent tuning and adaptive control for cement raw meal blending process, *Proceedings of the IFAC Intelligent Tuning and Adaptive Control*, Singapore, 301-306.
- [10] Banyasz, C. S., Keviczky, L. and Vajk, I., 1995. Application of a new generic optimal controller structure for raw material blending," *IFAC 8th Symposium on Automation in Mining, Mineral and Metal Processing, MMM'95*, Sun City, Republic of South Africa. August 29-31.

- [11] **Swain, A. K.**, 1995. Material mix control in cement plant automation," *IEEE Transactions on Automatic Control*, **95**, 23-27.
- [12] **Özsoy, C., Kural, A., Çetinkaya, M., and Ertuğ, S.**, 1999. Constrained MIMO self-tuning composition control in cement industry, *7th International Conference on Emerging Technologies and Factory Automation*, Barcelona, Spain, October 20-22, 1021-1027.
- [13] **Bogue, R.H.**, 1955. Chemistry of Portland Cement, Reinhold, New York.
- [14] **Lea, F.M.**, 1970. The Chemistry of Cement and Concrete, Arnold, London.
- [15] **Özsoy, C., Kazan, R., Tutum, G., and Çetinkaya, M.**, 1997. Çimento endüstrisinde hammadde harmanlama prosesinin optimal kontrolü, *III. Uluslararası Mekatronik Tasarım ve Modelleme Çalışma Toplantısı*, ODTÜ, Ankara, Türkiye, 15-18 Eylül.
- [16] **Lundan, A. and Mattila, O.**, 1974. A system for the control of the homogenization of the cement raw meal, *4th IFAC Symposium on Digital Computer Applications to Process Control*, Zurich, Switzerland.
- [17] **Pedersen, H.B.**, 1977. Automatic raw materials proportioning principles and experiences, *19th IEEE Cement Industry Technical Conference*, Omaha, NB.
- [18] **Lauren, K.G.**, 1978. Raw meal preparation a new technological system, *13th International Cement Seminar, Rock Products*, Chicago, USA.
- [19] **Weiss, S.J., and Gartner E.M.**, 1986. New control strategies for raw mix preparation, *IEEE Transactions on Industry Applications*, **22**, 324-329.
- [20] **Aström, K.J. and Bohlin, T.**, 1965. Numerical identification of linear dynamic systems from normal operating records. *IFAC Symposium on Self-Adaptive Systems*, Teddington, England.
- [21] **Ljung, L.**, 1986. The System Identification Toolbox: The Manual, Natick, MA.
- [22] **Ljung, L. and Glad, T.**, 1994. Modeling of Dynamic Systems. Prentice Hall, Englewood Cliffs, N. J.
- [23] **Eykhoff, P.**, 1974. System Identification. Wiley, New York.
- [24] **Goodwin, G.C. and Payne R.L.**, 1977. Dynamic System Identification: Experiment Design and Data Analysis. Academic Press, New York.
- [25] **Ljung, L.**, 1987. System Identification - Theory for the User. Prentice-Hall, Englewood Cliffs, N.J.
- [26] **Söderström, T. and Stoica, P.**, 1989. System Identification. Prentice-Hall, London.

- [27] **Box, G.E.P. and Jenkins, D.R.**, 1970. Time Series Analysis, Forecasting and Control. Holden-Day, San Francisco.
- [28] **Schoukens, J. and Pintelon, R.**, 1991. Identification of Linear Systems: A Practical Guideline to Accurate Modeling. Pergamon Press, London.
- [29] **Ljung, L. and Söderström, T.**, 1983. Theory and Practice of Recursive Identification. MIT press, Cambridge.
- [30] **Brillinger, D.**, 1981. Time Series: Data Analysis and Theory. Holden-Day, San Francisco.
- [31] **Guidorzi, R.P.**, 1975. Canonical structures in the identification of multivariable systems, *Automatica*, **11**, 361-374.
- [32] **Gautier, A. and Landau I.D.**, 1978. On recursive identification of multi-input multi-output system, *Automatica*, **14**, 609-614.
- [33] **Guidorzi, R.P.**, 1981. Invariants and canonical forms for systems structural and parametric identification, *Automatica*, **17**, 117-133.
- [34] **Correa, G.O. and Glover, K.**, 1984. Pseudo-canonical forms, identifiable parameterizations and simple parameter estimation for linear multivariable systems input output models, *Automatica*, **20**, 429-442.
- [35] **Gevers, M. and Wertz V.**, 1984. Uniquely identifiable state space and ARMA parameterizations for multivariable linear system, *Automatica*, **20**, 333-347.
- [36] **Gevers M.**, 1986. ARMA models, their Kronecker indices and their McMillan degree, *Int. J. Control*. **43**, 1745-1761
- [37] **Gevers, M. and Wertz V.**, 1987a. Parameterization issues in system identification, *Proc. IFA 10th World Congress*, Munich.
- [38] **Gevers, M. and Wertz, V.**, 1987b. Techniques for the selection of identifiable parameterizations for multivariable linear systems. *Control and Dynamic Systems: System Identification and Adaptive Control - Advances in Theory and Applications*, **26**, 35-86, Academic Press, New York.
- [39] **Duong, H.N. and Landau, I.D.**, 1994. On statistical properties of a test for model structure selection using the extended instrumental variable approach, *IEEE Transactions on Automatic Control*, **39**, 211-215.
- [40] **Duong, H.N. and Landau, I.D.**, 1996. An IV based criterion for model order selection, *Automatica*, **32**, 909-914.
- [41] **Rubinstein, R.Y.**, 1981. Simulation and the Monte Carlo Method. Wiley-Interscience, New York.
- [42] **Morgan, B.J.T.**, 1984. Elements of Simulation. Chapman and Hall, London.

- [43] **Davies, W.D.T.**, 1970. System Identification for Self-Adaptive Control. Wiley-Interscience, New York.
- [44] **Landau, I. D., Lozano R. and M'Saad M.**, 1998. Adaptive Control. Spriger - Verlag, London.
- [45] **Box, G.E.P. and Pierce D.A.**, 1970. Distribution of residual autocorrelations in autoregressive integrated moving average time series models, *J. Amer. Statist. Assoc.*, **65**, 1509-1526.
- [46] **Söderström, T.**, 1977. On model structure testing in system identification, *International Journal of Control*, **26**, 1-18.
- [47] **Freeman, T.G.**, 1985. Selecting the best linear transfer function model, *Automatica*, **21**, 361-370.
- [48] **Billings, S.A. and Woon W.S.**, 1986. Correlation based model validity tests for non-linear models, *International Journal of Control*, **44**, 235-244.
- [49] **Lehmann, E.L.**, 1986. Testing Statistical Hypotheses, Wiley, New York.
- [50] **Stoica, P., Eykhoff, P., Janssen, P. and Söderström, T.**, 1986. Model structure selection by cross-validation, *International Journal of Control*, **43**, 1841-1878.
- [51] **Leontaritis I.J. and Billings, S.A.**, 1987. Model selection and validation methods for non-linear systems, *International Journal of Control*, **45**, 311-341.
- [52] **Duong, H.N. and Landau, I.D.**, 1994. On test horizon for model validation by output error, *IEEE Transactions on Automatic Control*, **39**, 102-106.
- [53] **Richalet, J., Rault, A., Testud, J.L. and Papon, J.**, 1978. Model predictive heuristic control: application to industrial processes. *Automatica*, **14**, 413-428.
- [54] **Cutler, C.R. and Ramaker, B.C.**, 1980. Dynamic matrix control - a computer control algorithm, *In Automatic Control Conference*, San Francisco.
- [55] **Clarke, D.W., Mohtadi, C. and Tuffs, P.S.**, 1987. Generalized predictive control : Part I. The basic algorithm. *Automatica*, **23**, 137-148.
- [56] **Lelic, M.A. and Welstead, P.E.**, 1987 Generalized pole placement self tuning controller, Part 1. basic algorithm, *International Journal of Control*, **46**, 547-568.
- [57] **Linkers, D.A. and Mahfonf, M.**, 1994. Advances in Model-Based Predictive Control: Generalized Predictive Control in Clinical Anaesthesia. Oxford University Press.
- [58] **Clarke, D.W.**, 1988. Application of generalized predictive control to industrial processes, *IEEE Control Systems Magazine*, **122**, 49-55.

- [59] **Richalet, J.**, 1993. Industrial applications of model based predictive control, *Automatica*, **29**, 1251-1274.
- [60] **Kinnaert, M.**, 1989., Adaptive generalized predictive controller for MIMO systems, *International Journal of Control*, **50**, 161-172.
- [61] **Clarke, D.W., Mohtadi, C. and Tuffs, P.S.**, 1987. Generalized predictive control : Part II. Extensions and interpretations. *Automatica*, **23**, 149-160.
- [62] **De Keyser, R.M.C. and Van Cuawenberghe, A.R.**, 1985. Extended prediction self-adaptive control. In *IFAC Symposium on Identification and System Parameter Estimation*, York, UK, 1317-1322.
- [63] **Richalet, J. Abu el Ata-Doss, S., Arber, C., Kuntze, H.B., Jacobash, A. and Schill, W.**, 1987. Predictive functional control : Application to fast and accurate robots. In *Proceedings 10th IFAC World Congress*, Munich.
- [64] **Ydstie, B.E.**, 1984. Extended horizon adaptive control, In *Proceedings 9th IFAC World Congress*, Budapest, Hungary.
- [65] **Söeterboek, A.R.M.**, 1992. Predictive Control: A unified approach. Prentice-Hall.
- [66] **Richalet, J., Rault, A., Testud, J.L. and Papon, J.**, 1976. Algorithmic control of industrial processes. In *4th IFAC Symposium on Identification and System Parameter Estimation*. Tbilisi, URSS.
- [67] **Propoi, A.I.**, 1963. Use of LP methods for synthesizing sampled-data automatic systems. *Automn Remote Control*, **24**.
- [68] **Peterka, V.**, 1984. Predictor based self-tunning control, *Automatica*, **20**, 39-50.
- [69] **Greco, C., Menga, C., Mosca, E. and Zappa, G.**, 1984. Performance improvement of self tuning controllers by multistep horizons: the MUSMAR approach. *Automatica*, **20**, 681-700.
- [70] **Lemos, J.M. and Mosca, E.**, 1985. A Multipredictor-based LQ self-tuning controller, In *IFAC Symposium on Identification and System Parameter Estimation*, York, UK, 137-141.
- [71] **Moran, M.**, 1994. Advances in Model-Based Predictive Control : Model Predictive Control, Oxford University Press.
- [72] **Lee, J.H., Moran, M. and Garcia, J.E.**, 1994. State-space interpretation of model predictive control. *Automatica*, **30**, 707-717.
- [73] **Mohtadi, C.**, 1986. Advanced self-tuning algorithms. *PhD thesis*, Oxford University, U.K.

- [74] **Clarke, D.W. and Scattolini, R., 1991.** Constrained receding-horizon predictive control. *Proceedings IEE*, **138**, 347-354.
- [75] **Mosca, E., Lemos, J.M. and Zhang, J., 1990.** Stabilizing I/O receding horizon control. *In IEEE Conference on Decision and Control*.
- [76] **Kouvaritakis, B., Rossiter, J.A. and Chang, A.O.T., 1992.** Stable generalized predictive control: an algorithm with guaranteed stability. *Proceedings IEEE*, **139**, 349-362.
- [77] **Campo, P.J. and Morari, M., 1987.** Robust model predictive control. *In American Control Conference*, Minneapolis, Minnesota.
- [78] **Allwright, J.C., 1994.** Advances in Model-Based Predictive Control: On min-max model-based predictive control. Oxford University Press.
- [79] **Camacho, E.F. and Bordons, C., 1995.** Model Predictive Control in the Process Industry, Springer Verlag, London.
- [80] **Riggs, J.B., 1999.** Chemical Process Control, Ferret, USA.
- [81] **Seborg, D.E., Edgar, T.F. and Mellichamp, D.A., 1989.** Process Dynamics and Control, John Wiley&Sons., USA.

APPENDIX A

Model 2 (MISO): u_1 = low grade, u_2 = iron ore, y_1 = lime module

W_1 =unity; weight on output, $W_2=[0.01 \ 0.001]$; weights on inputs

$n = 4$; for prediction horizon, $U = 3$; for control horizon

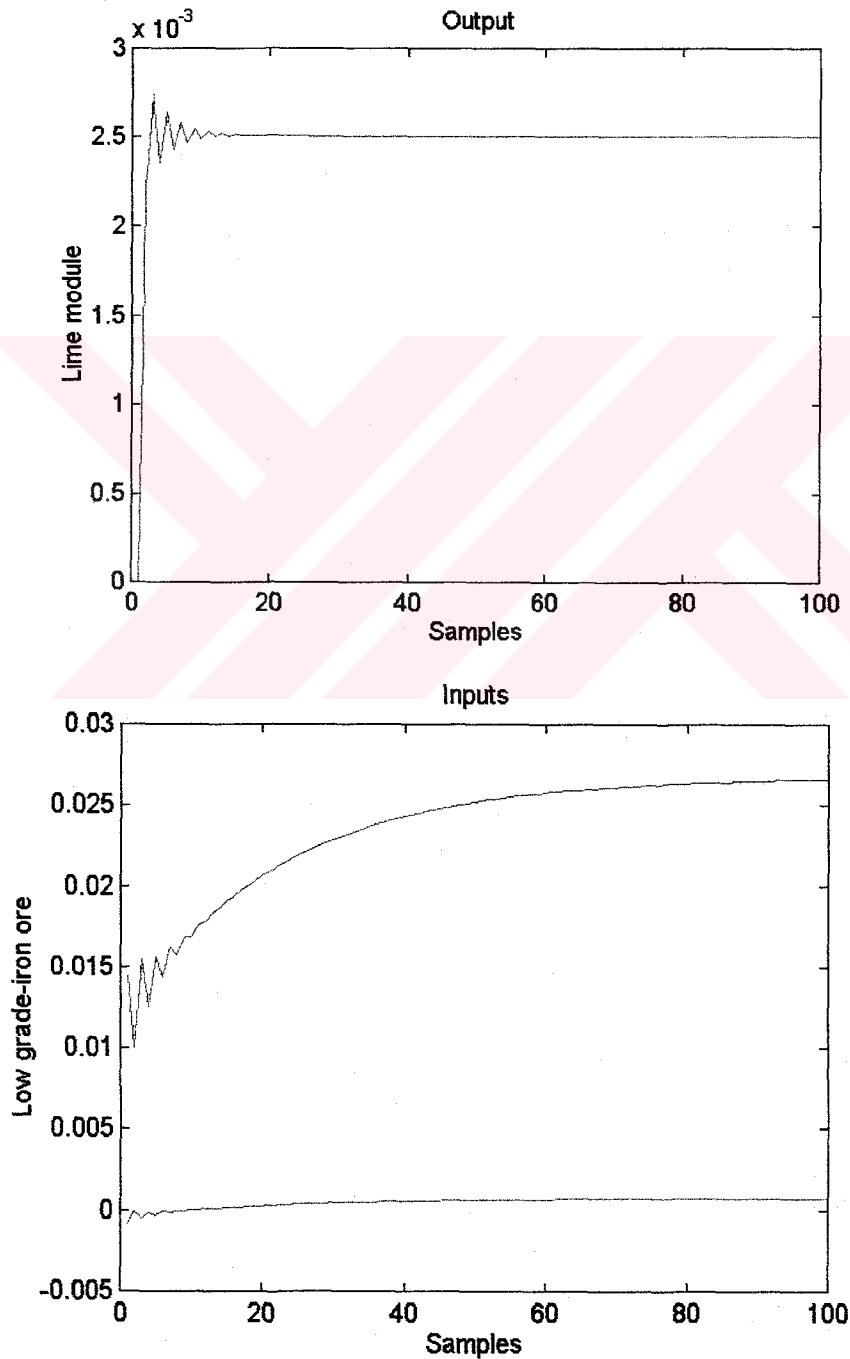


Figure A.1. Input-output values of controlled process for model 1 at the beginning.

Adjusted design parameters :

W_1 = unity ; weight on output

$W_2 = [0.4 \ 0.3]$; weights on both inputs

$n = 6$; for prediction horizon,

$U = 6$; for control horizon

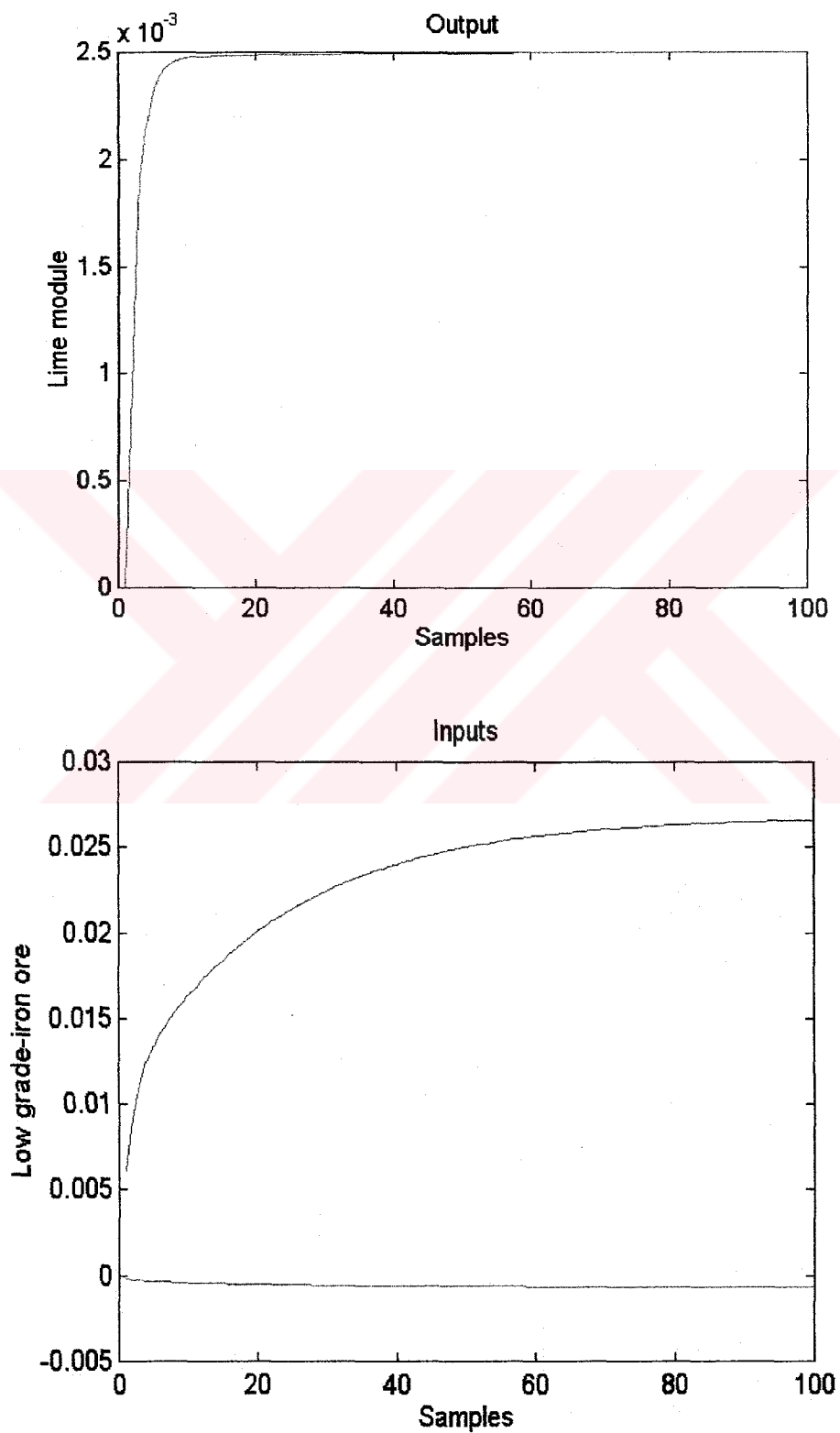


Figure A.2. Input-output values of controlled process for model 2 with adjusted parameters.

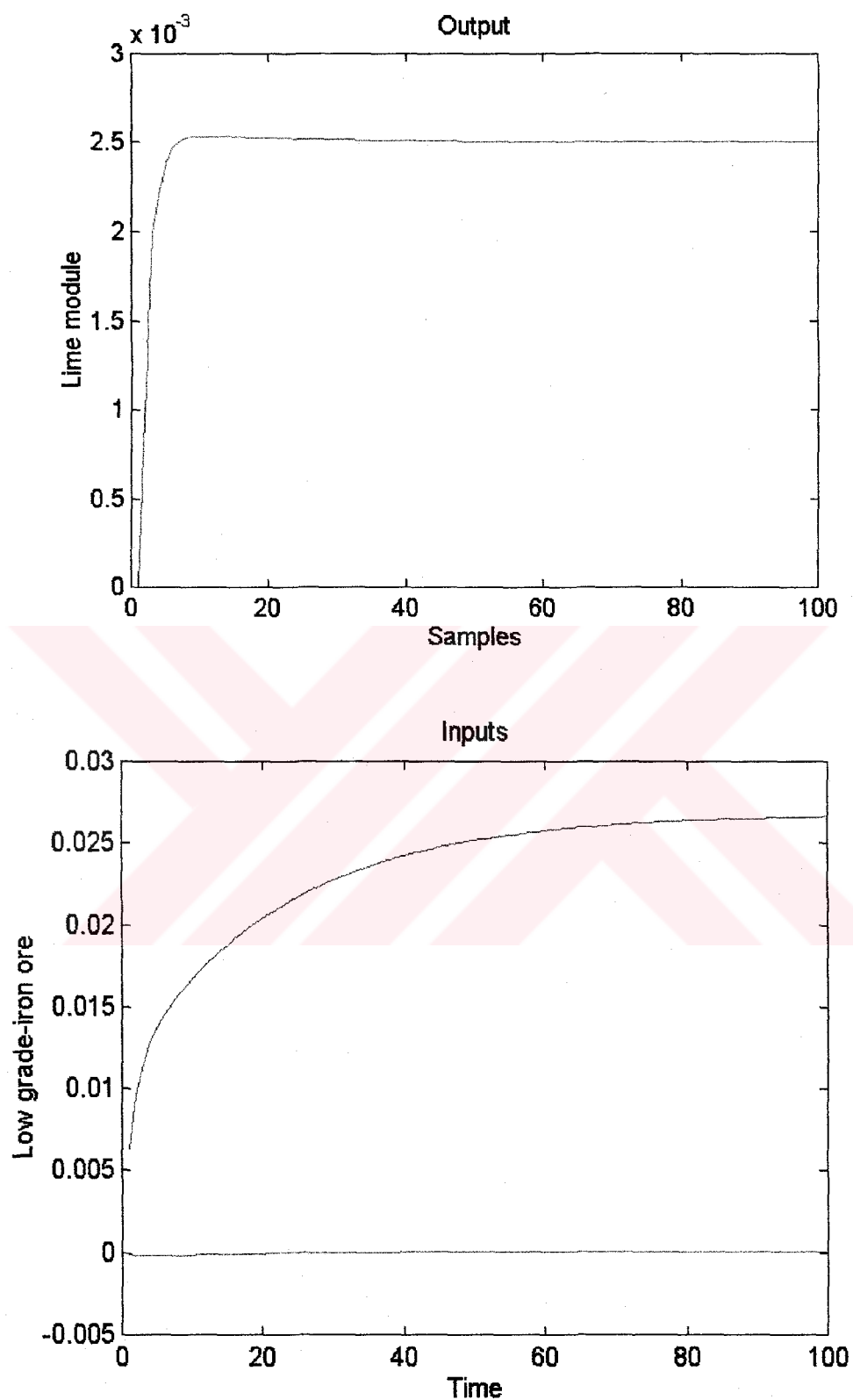


Figure A.3. Input-output values of controlled process for model 2 ($n=12$, $U=4$).

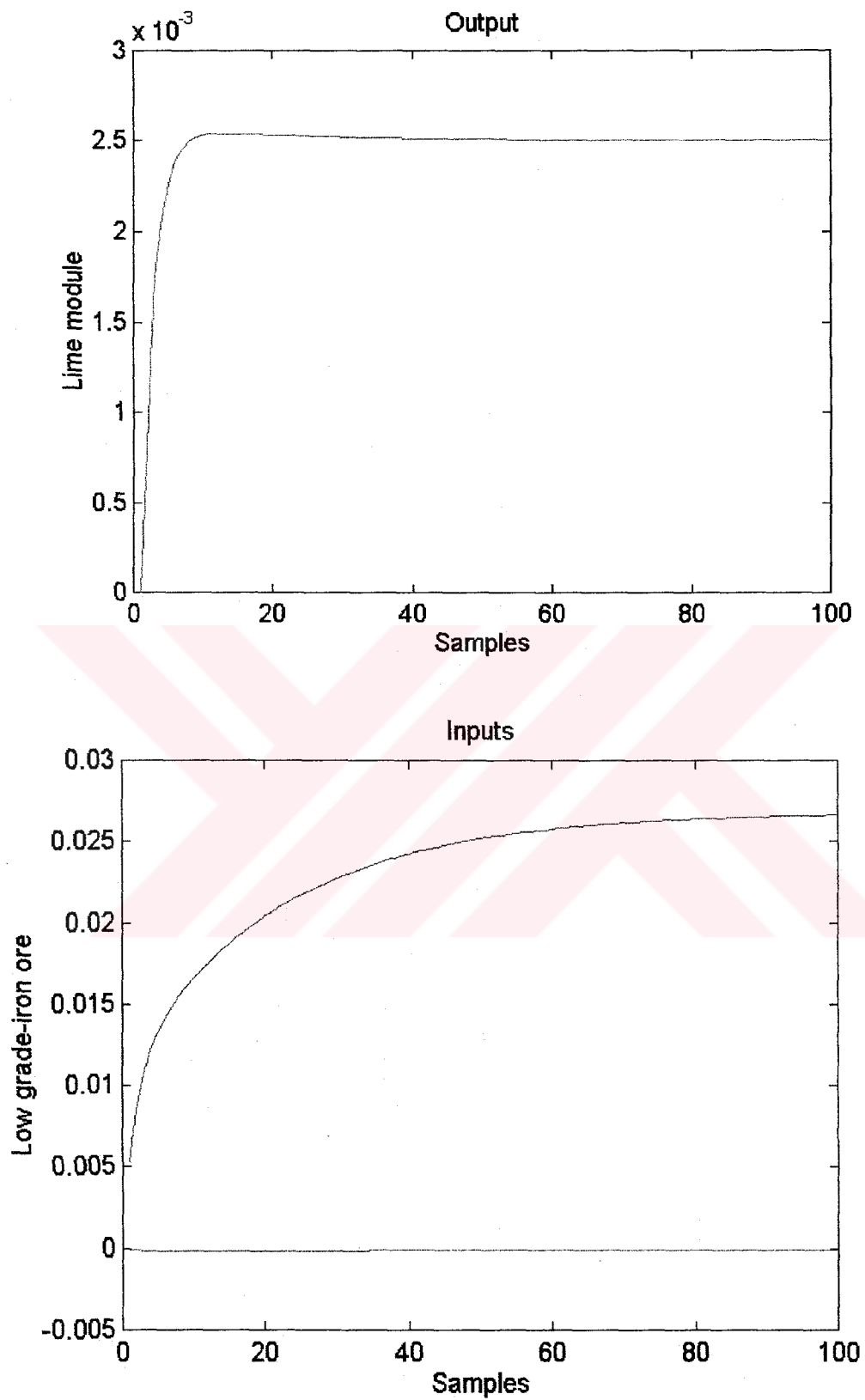


Figure A.4. Input-output values of controlled process for model 2 ($W_2=[0.6 \ 0.4]$).

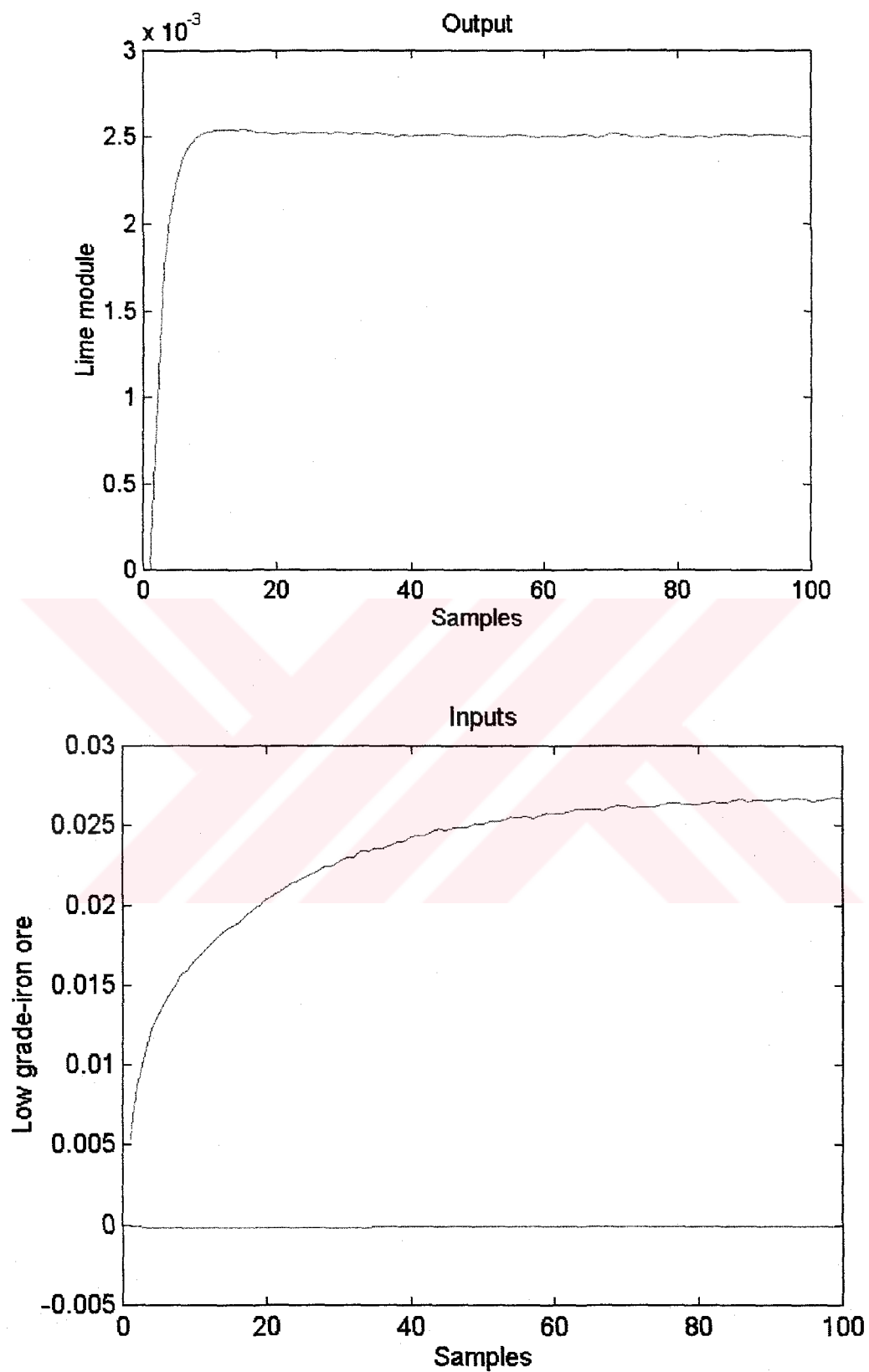


Figure A.5. Input-output values of controlled process for model 2 with noise.

Model 3 (MIMO): u_1 = low grade, u_2 = iron ore, y_1 = iron oxide, y_2 = lime module
 W_1 =unity; weight on output, $W_2=[0.1 \ 0.08]$; weights on inputs
 $n = 2$; for prediction horizon, $U = 2$; for control horizon

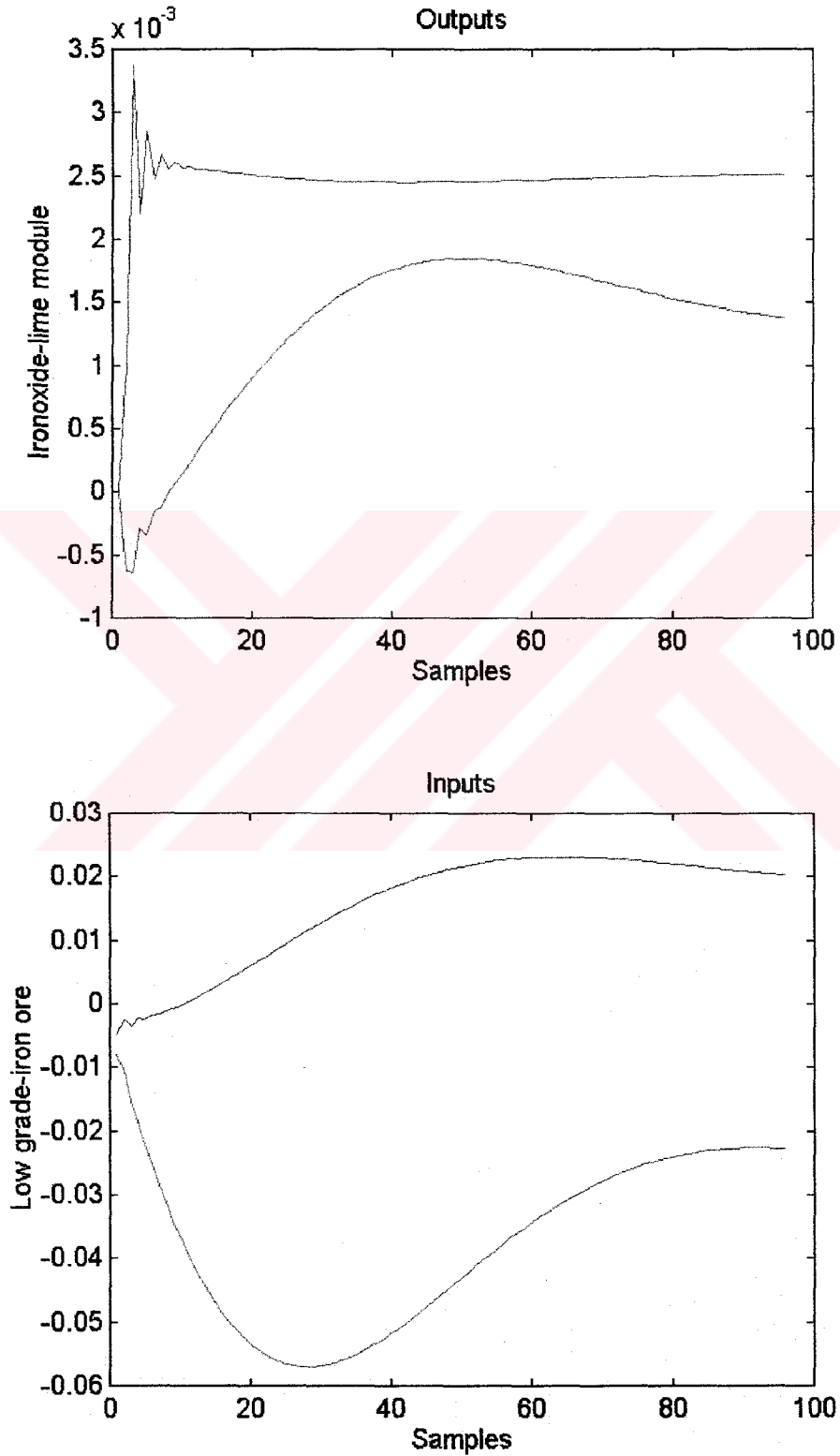


Figure A.6. Input-output values of controlled process for model 3 at the beginning.

Adjusted design parameters :

W_1 = unity ; weight on output $W_2 = [0.015 \ 0.015]$; weights on both inputs
 $n = 8$; for prediction horizon, $U = 8$; for control horizon

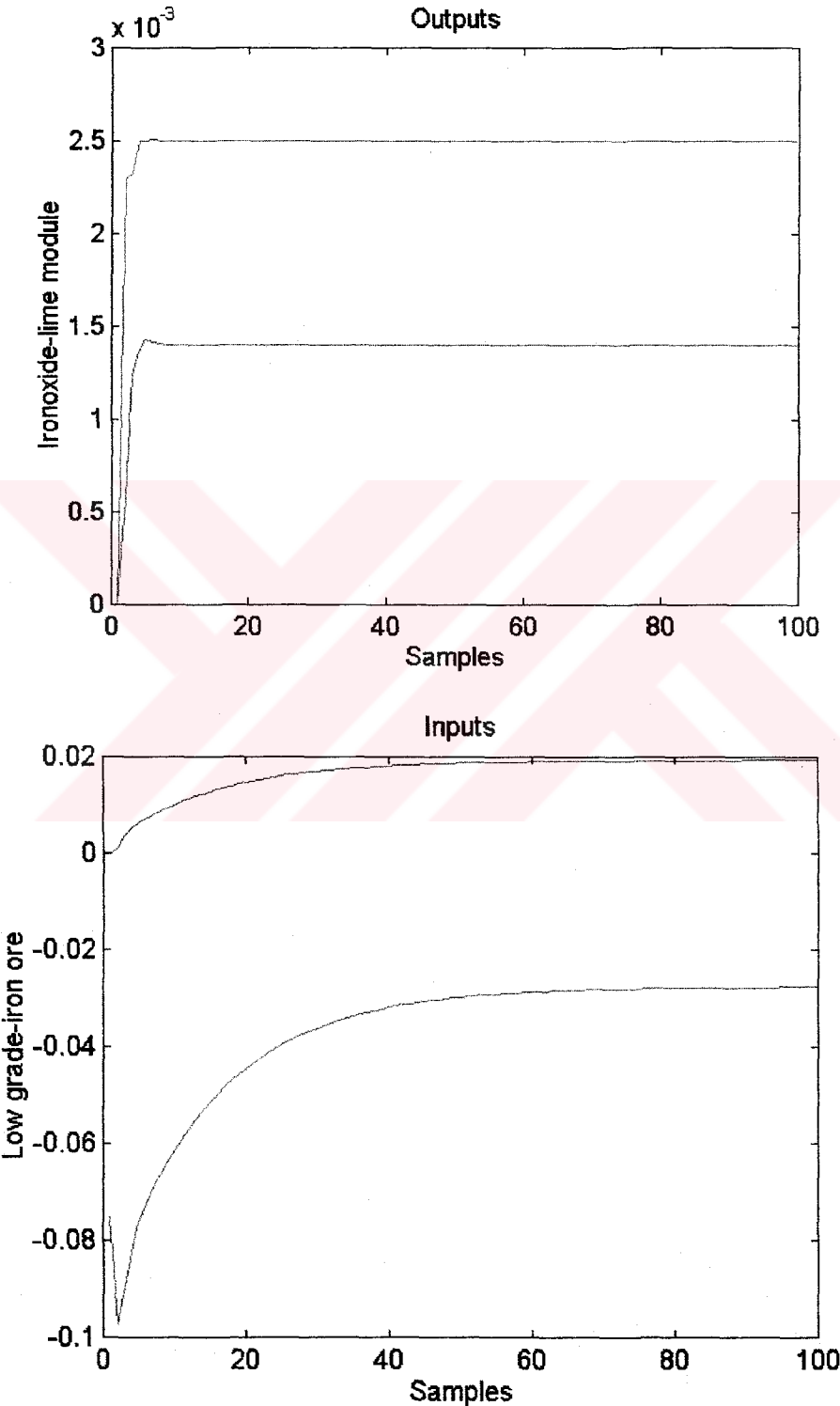


Figure A.7.Input-output values of controlled process for model 3 with adjusted parameters

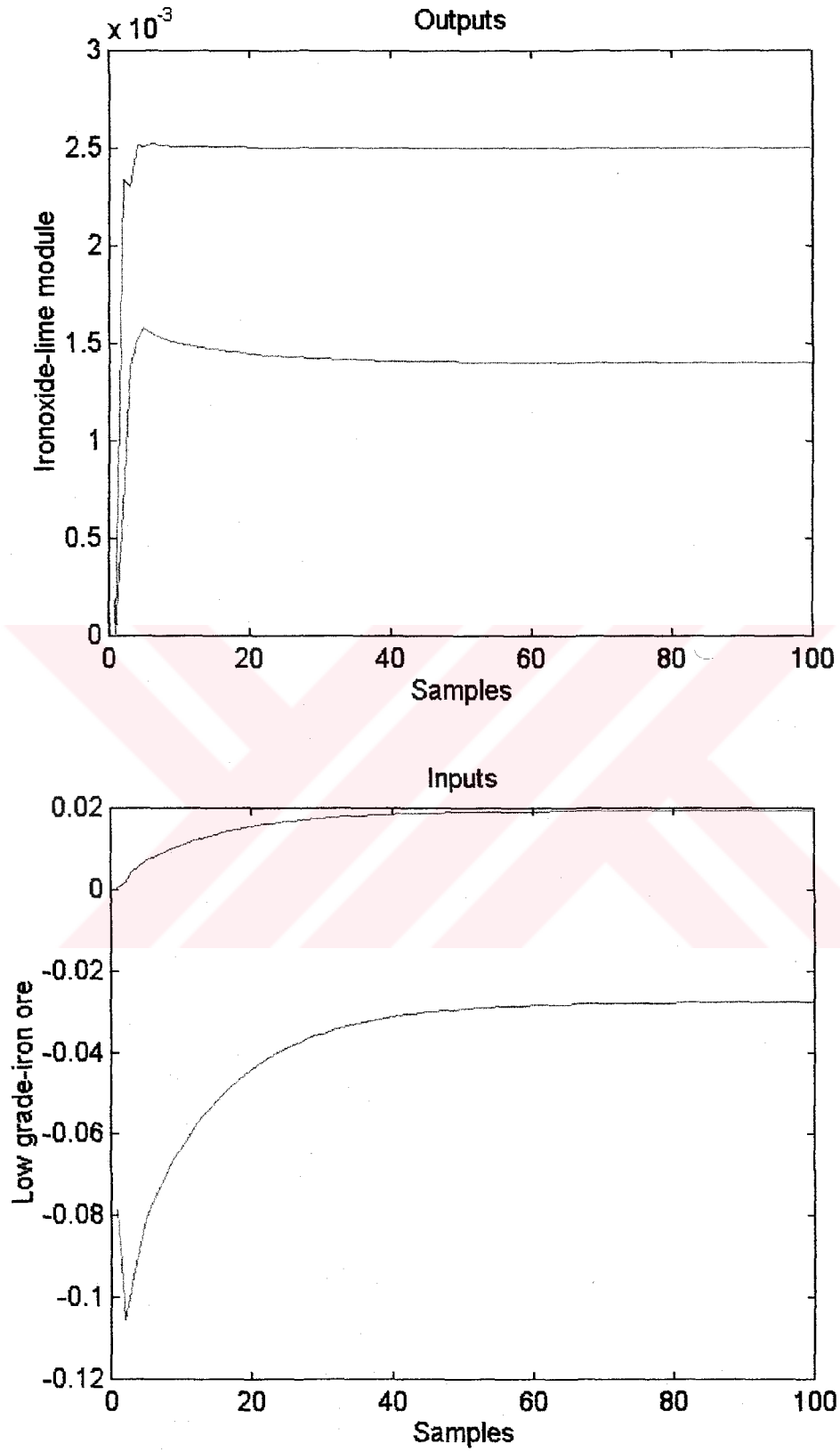


Figure A.8. Input-output values of controlled process for model 3 ($n=13$, $U=6$).

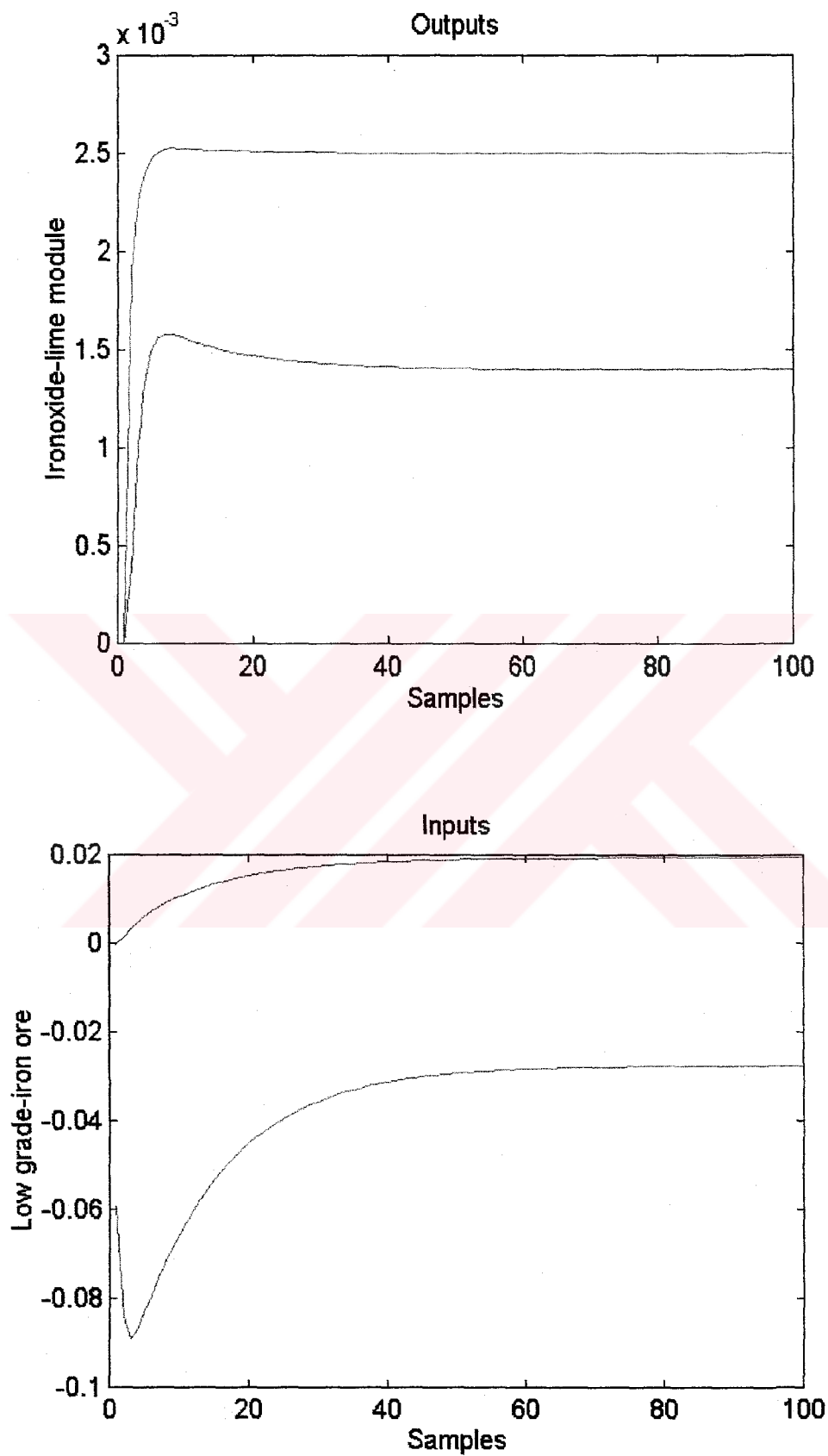


Figure A.9. Input-output values of controlled process for model 3 ($W_2=[0.03, 0.1]$).

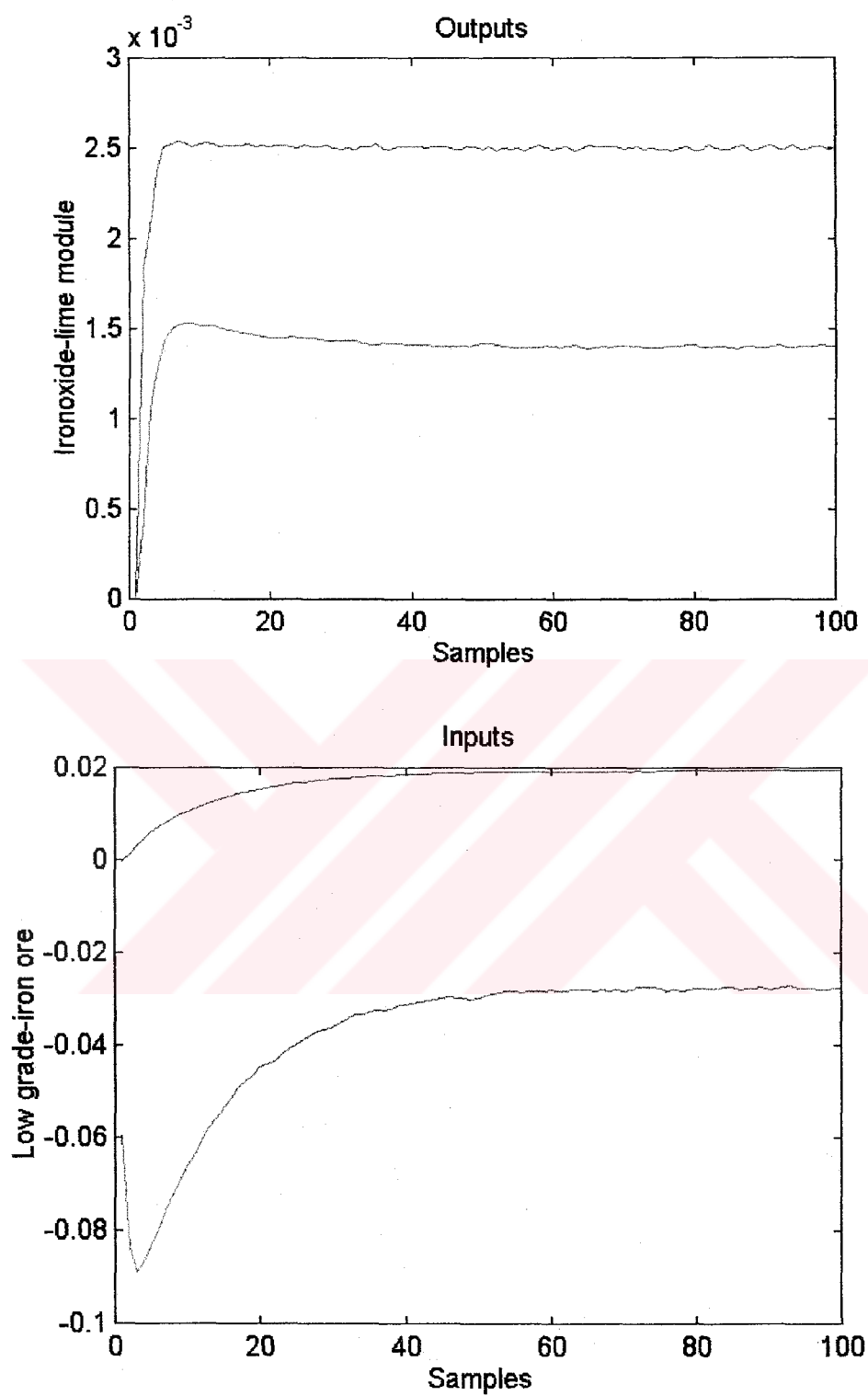


Figure A.10. Input-output values of controlled process for model 3 with noise.

AUTOBIOGRAPHY

Ayhan Kural was born in İstanbul in 1968. He graduated from Kabataş High School in 1985. He obtained his B.Sc. degree in Mechanical Engineering from İstanbul Technical University (İTÜ) in 1989. After his graduation, he became research assistant at Automatic Control Department in İstanbul Technical University. He continued his graduate studies at the same university and received his M.Sc. degree at the end of 1992 and then attended Ph.D. program offered in İ.T.Ü. He is still working a research assistant in the same department.

