Closed Loop Identification For Model Predictive Control: A Case Study

Shiraz Amjad

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE
In

ELECTRICAL ENGINEERING

October 2003

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SUADI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis written by **SHIRAZ AMJAD** under the direction of his thesis advisor and approved by his thesis committee has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**.

Thesis Committee

Dr. Hussain N. Al-Duwaish (Chairman)

Dr. Jamil M. Bakhashwain (Member)

Dr. Youssef L. Abdel-Magid (Member)

Dr. Samir A. Al-Baiyat (Member)

Dr. Muhammad A. Al-Arfaj (Member)

Dr. Jamil M. Bakhashwain
Department Chairman

Prof. Osama A. Jannadi
Dean of Graduate Studies

Date

Dedicated to

My beloved Parents

ACKNOWLEDGMENTS

In the name of Allah, the Most Beneficent, the Most Merciful.

All praise and glory goes to Almighty Allah (Subhanahu Wa Ta'ala) who gave me the courage and patience to carry out this work. Peace and blessings of Allah be upon his last Prophet Muhammad (Sallulaho-Alaihe-Wassalam) and all his Sahaba (Razi-Allaho-Anhum) who devoted their lives towards the prosperity and spread of Islam.

Acknowledgment is due to King Fahd University of Petroleum & Minerals for providing support and facilities in carrying out this work.

My deep appreciation and profound gratitude goes to my thesis advisor Dr. Hussain N. Al-Duwaish for his constant endeavor, guidance and the numerous moments of attention he devoted throughout the course of this research work. His stimulating suggestions made this work interesting and infromative for me.

I extend my deepest gratitude to my thesis committee members: Dr. Jamil M. Bakhashwain, Dr. Youssef L. Abdel-Magid, Dr. Samir A. Al-Baiyat and Dr. Muhammad A. Al-Arfaj for their help, continuous encouragement and support.

Acknowledgment is due to my fellows Saad Azher and Moinuddin with whom I had informative discussion at various stages of this work.

Special thanks are due to my house mates, colleagues and my friends for their encouragement, motivation and pivotal support. A few of them are Jawwad, Sajid, Ajmal, Majid, Fareed, Munib, Hassan, Naeem, Ahmer and many others, whom I will not be able to name here. They have made my work and stay at KFUPM very pleasant and memorable.

Special appreciation goes to Dr. Anis Siddique, Mr. Habib Babar and Mr. Qayyum Ashraf who provided a great deal of moral support during the course of my studies at KFUPM.

I cannot forget the prayers of my late grandmother Mumtaz Begum and my late grandfather Col (R). Saeed, who always wished to see me attain a successful place in this life and always believed in me. May their souls rest in peace (Ameen).

Finally and humbly, I wish to record my sincere appreciation and thanks to my father and mother who formed part of my vision and taught me the good things that really matter in life. They still provide a persistent inspiration for my journey in this life. I am also grateful to my two sisters, brother and other family members for their invaluable support, prayers and encouragement.

Contents

List of Tables

List of Figures

THESIS ABSTRACT

Name: SHIRAZ AMJAD

Title: CLOSED LOOP IDENTIFICATION FOR MODEL PREDICTIVE CONTROL: A CASE STUDY

Degree: MASTER OF SCIENCE

Major Field: ELECTRICAL ENGINEERING

Date of Degree: OCTOBER 2003

In the past two decades, model predictive control (MPC) technology has gained its industrial position in refinery and petrochemical industries and is beginning to attract interest from other process industries. Model development remains by far the most critical and time-consuming step in implementing industrial MPC. Conventionally, models are identified through a series of single variable open loop step tests. Overtime these models change due to a number of reasons such as change in operating conditions, instability of the process, drift in process and environmental conditions. This causes control performance problems. For this reason, multivariable closed loop identification is proposed for MPC. Identification techniques based on autoregressive moving average (ARMA), state space and neural network models are investigated. These techniques are tested through computer simulations and ultimately on field data from an industrial process. Direct Identification method is employed for this purpose.

Keywords: *Closed loop, MPC, ARX, ARMAX, OE, MFNN.*

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS, DHAHRAN.

OCTOBER 2003

Chapter 1

Introduction

Model Predictive Control (MPC) has developed considerably over the last few years, both within the research control community and in industry. It integrates optimal control, stochastic control and multivariable control. The term MPC does not designate a specific control strategy but a large range of control methods, which make an explicit use of a model of the process to obtain the control signal by minimizing an objective function. These predictive controllers are based on prediction of the future behavior of the process, forecasted using this model of the process. Industrial project experience has shown that the most difficult and time-consuming work in an MPC project is modeling and identification [1].

The dual topics of identification for control and model based control design have attracted extensive discussions for the past two decades, which naturally lead to the currently innovated concept of integrated system identification and control design. In MPC scheme, the controller can be seen as an algorithm operating on a model of the process (subject to disturbances) and optimized in order to reach given control design objectives. The performance of the controller depends on this identified plant model.

Current practice of MPC industry is to use a series of open loop and single variable step tests [1]. The tests are carried out manually. The advantage of this test method is that control engineer can watch many step responses during the tests and can learn about the process behavior in an intuitive way. On the other hand, the biggest problem with open loop identification test is its high cost in time and manpower. Problems arise when these (open loop) identified models start exhibiting degradation in their performance i.e. they can no longer describe correctly the input-output behavior of the process and become inadequate after some span of time. Many reasons can be associated with this phenomenon like change in process operating conditions, drift in process conditions (controlled variables), environment conditions, which are not taken into account during identification, and, instability and inherent feedback mechanisms of the plant. Examples of such processes are refineries, where an increase or decrease in flow of crude oil can change the entire dynamics of the plant model and high purity distillation columns, which are often ill-conditioned where top and bottom compositions have a strong correlation which makes it difficult for the model to represent the process for a long span of time.

The only practical solution existing in the industry today is to shut down the controller and identify the model again, resulting in huge financial and production losses.

Widespread application of MPC technology requires more effective and efficient method of multivariable process identification since modeling of the plant and design of controller cannot be considered always as two separate issues.

This problem of process identification for MPC has started to attract attentions of both academic institutes and industry. The problem of degradation can be dealt with closed loop identifying the model and checking the prediction error between the identified model and original model. This offers a number of practical advantages such as obtaining better models, validation, and controller maintenance and order reduction.

The next section gives an introduction to closed loop identification followed by the literature review.

1.1 Closed Loop Identification

Closed loop identification has often been suggested as a tool for identification of models that are suitable for control, so called identification for control. The main motivation has been that by performing the identification experiments in closed loop it is possible to match the identification and control criteria so that the model is best fit to the data in a control relevant way. In the past, closed loop identification was considered difficult due to lack of modern day fast computing facilities and apprehensions of the industry in general due to financial constraints. However, access to better research and fast computing facilities in the last decade has established the importance of closed loop identification in the process industry.

Many systems work under closed loop control as in Fig. 1.1, where the signal *r(t)* can be a reference value or a set point, *v(t)* is noise disturbance that is modeled as a filtered zero mean white noise, *u(t)* and *y(t)* are the process input and output respectively. The basis for all identification is the available data set

$$Z^N = \{y(1),.....,y(N)\, u(1),......,u(N)\} \tag{1.1}$$

consisting of input-output signals, *u(t)* and *y(t)*, *t=1,…,N*. The process output and the control input are given by [2]:

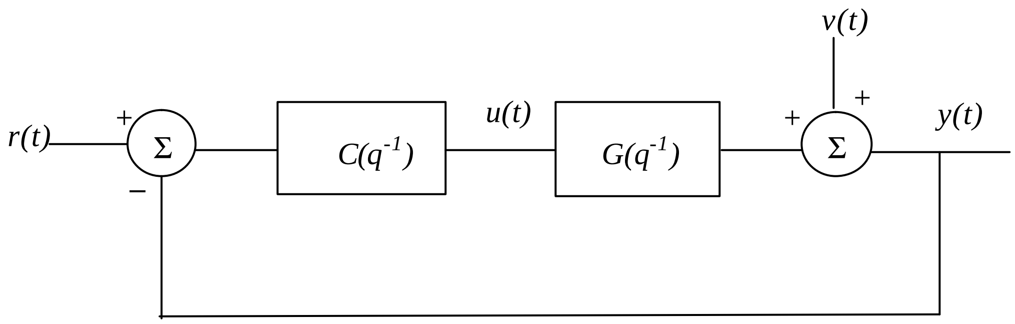

Figure 1.1: Typical feedback system

$$y(t) = G(q^{-1})\,u(t) + v(t) \tag{1.2}$$

$$y(t) = G(q^{-1})\,u(t) + H(q^{-1})e(t) \tag{1.3}$$

$$u(t) = C(q^{-1})(r(t) - y(t)) \tag{1.4}$$

Where, $G(q^{-1})$ is the process transfer function, $H(q^{-1})$ is the noise transfer function and $C(q^{-1})$ is the controller transfer function. $e(t)$ is zero mean white noise sequence and q^{-1} is the unit backward shift operator. The open loop transfer function $G(q^{-1})$ and the controller transfer function $C(q^{-1})$ are given by

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})} = \frac{b_1 q^{-1} + \ldots\ldots\ldots\ldots + b_n q^{-n_b}}{1 - a_1 q^{-1} - \ldots\ldots\ldots\ldots - a_n q^{-n_a}} \tag{1.5}$$

$$C(q^{-1}) = \frac{Q(q^{-1})}{P(q^{-1})} = \frac{q_0 + q_1 q^{-1} + \ldots\ldots\ldots\ldots + q_m q^{-m_q}}{1 + p_1 q^{-1} + \ldots\ldots\ldots\ldots + p_m q^{-m_p}} \tag{1.6}$$

The closed loop system can be represented as

$$y(t) = S(q^{-1})G(q^{-1})C(q^{-1})r(t) + S(q^{-1})v(t) \tag{1.7}$$

where $S(q^{-1})$ is the sensitivity function,

$$S(q^{-1}) = \left(I + G(q^{-1})C(q^{-1})\right)^{-1} \tag{1.8}$$

Eq. 1.7 can also be expressed as

$$y(t) = G_c(q^{-1})r(t) + v_c(t) \tag{1.9}$$

where $G_c(q^{-1})$ and $v_c(q^{-1})$ are defined as in Eqs. 1.10 and 1.11

$$G_c(q^{-1}) = S(q^{-1})G(q^{-1})C(q^{-1}) \tag{1.10}$$

$$v_c(q^{-1}) = S(q^{-1})v(t) \tag{1.11}$$

In closed loop configuration, the input can be expressed as

$$u(t) = C(q^{-1})S(q^{-1})r(t) - C(q^{-1})S(q^{-1})v(t) \tag{1.12}$$

In general, all closed loop identification methods can be classified as *direct, indirect*, or *joint input-output* methods.

1.1.1 Direct Identification Method

In the direct approach, the method is applied directly to measured input-output (*u*, *y)* data and no assumptions whatsoever are made on how the data was generated. In general, model structure of the following form is used.

$$y(t) = G(q^{-1}, q)u(t) + H(q^{-1}, q)e(t) \tag{1.13}$$

where G is the dynamic model and H is the noise model. ? is the parameter vector that has to be estimated. The one step-ahead predictor (Appendix A) for direct identification is

$$\hat{y}(t \mid q) = H^{-1}(q^{-1}, q)G(q^{-1}, q)u(t) + \left(1 - H^{-1}(q^{-1}, q)\right)y(t) \tag{1.14}$$

1.1.2 Indirect Identification Method

The indirect method assumes knowledge of the controller used in the identification experiment and the idea is to identify the closed-loop system and to compute the open loop parameters from this estimate, using the knowledge of the controller. The model structure for indirect method is

$$y(t) = G_C(q^{-1}, q)r(t) + H_*(q^{-1})e(t) \qquad (1.15)$$

where $G_C(q, q)$ is a model of the closed loop system. H_* is a fixed noise model which is standard in indirect method (often equal to 1). The corresponding one-step-ahead predictor is

$$\hat{y}(t\,|\,q) = H_*^{-1}(q^{-1})G_C(q^{-1}, q)r(t) + \left(1 - H_*^{-1}(q^{-1})\right)y(t) \qquad (1.16)$$

From Eq. 1.16, it is clear that estimating q is an open loop problem since the noise and the reference signal are uncorrelated [3]. This implies that any identification method that works in open loop can be used to find the estimate of the closed-loop system. In the first step, G_C is estimated from measured *y* and *r*, giving $\hat{G}_C$. Then the open loop transfer function $\hat{G}$ is found in the second step

$$\hat{G}(q^{-1}) = \hat{G}_C(q^{-1})\left(I - \hat{G}_C(q^{-1})C(q^{-1})\right)^{-1} \qquad (1.17)$$

1.1.3 Joint Input-output Method

The third approach, the joint input-output method amounts to modeling the input *u(t)* and the output *y(t)* jointly as outputs from an augmented system driven by the reference signal *r(t)* and the un-measurable noise *v(t)*. Given an estimate of this augmented system, the open loop model parameters (and the controller) can be estimated. In this method exact knowledge of the controller parameters is not required. However, it must be known or assumed to be of a certain linear structure.

In the first step, measured reference signal (r) and input (u) are used to estimate a model $\hat{S}$ of the sensitivity function S. Next this model is used to construct the signal

$$\hat{u}(t) = \hat{S}(q^{-1})r(t) \tag{1.18}$$

which is then used to identify the open loop system as in Eq. 1.18.

$$\hat{y}(t|q) = G(q^{-1}, q)\hat{u}(t) \tag{1.19}$$

The next subsection gives a detailed literature survey on some of the major works done in the field of closed loop identification.

1.2 Literature Review

Identification is the experimental approach to process modeling. The system identification can be divided into a number of sub problems; experimental design, data collection, model structure selection, model estimation and model validation. These steps are applicable in closed loop identification as well. Mathematical models of dynamic systems are of rapidly increasing importance in engineering and today all designs are more or less based on mathematical models. If the physical laws governing the behavior of the system are known, so called *white-box models* of the system can be constructed. At the other end of the modeling scale, there are so called, *black-box modeling* or *identification*. Black-box models are constructed from data using no physical insight whatsoever and the model parameters are simply knobs that can be turned to optimize the model fit.

In the seventies, there was a very active interest in issues concerning closed loop identification as summarized in the survey paper by Gustavson et al. [4]. Much of the attention was devoted to identification and accuracy aspects. In 1983 Sinha and Kuszta [5] provided the classification of closed loop identification schemes based on signals, plant and regulator architecture for linear systems. The same year Ljung and Soderstrom [6] discussed implementation of recursive identification in closed loop to the adaptive control problem. In 1989, Soderstrom and Stoica [7] presented parametric identification methods that were typically directed towards solving the consistency problem, considering the situation that the plant and disturbance model could be modeled exactly. Polderman and Praagman [8] addressed the problem of classifying the adaptive control problems for which, despite the fact that the system was not identified in closed loop, the system was controlled properly (meaning that due to some miracle a wrong estimate would coincidently give the right controller). As a solution to this paradigm they presented a recursive procedure in which the controller parameters were updated online on the basis of model parameters identified using prediction error method.

The advent of the 90s saw a renewed interest in the closed loop identification of models that were particularly suitable for model-based (robust) control design. Most controller design schemes till now were concentrated on open loop identification. In 1990, Liu and Skelton [9] presented an iterative controller design scheme which consisted of indirect closed loop identification of the plant and then controller design cycles. They utilized q-Markov cover algorithm, which obtained state space realization of the unknown plant using only input-output data. A similar approach was presented by Klauw and Verhaegen [10] in 1991. They used the joint input-output identification scheme in time domain to identify a linear MIMO process controlled by a Proportional Integral (PI) controller. Both the process and the controller were assumed to be linear-time invariant causal systems. The closed loop system was considered as a joint system with white noise inputs, generating the process input and output, which were used to construct a Markov model. From this model minimal realizations of the process and controller were obtained. Stochastic realization was used to solve the spectral factorization problem for this approach. Scharma [11] presented the idea of treating approximate identification and model based control as joint problem when designing a high performance control system. He proposed an iterative scheme based on frequency response identification and robust control design to solve this joint problem. Each identification step used the previously designed model based controller to obtain new data from the plant. The identification was done using coprime factorization of the unknown plant in the frequency domain. Musto and Lauderbaugh [12] presented a heuristic approach capable of generating ARMAX models of linear systems to be used for controller design in an expert-aided adaptive control system, while the system was online (closed loop). The proposed algorithm also contained heuristics for sampling rate selection, delay estimation and model validation.

In 1994, Voda and Landau [13] used an iterative closed loop identification scheme to design a PID controller by using a proper data filter in the model estimation. They used ARMAX model of the plant for real time control of air heater process and feed tank pressure of the heating company of Grenoble. In their proposed iterative scheme, each identification step used the previously designed controller to obtain new data from the plant, which was then filtered to minimize a filtered prediction error to obtain a new model of the plant. The same year Klauw et al. [14] applied both the direct and two-step (joint input-output) identification approaches to identify a suitable model for a two-input-two-output distillation column using closed loop data. They used Output Error (OE) and linear regression model schemes based on orthonormal basis functions. According to them, the two step (joint input-output) method performed better than the direct method and considerable improvement was obtained in the prediction capabilities of the model when compared to the open loop identified model.

By 1995 the problem of unavailability of plants for identification in the open loop due to high costs associated with it, led to researchers like Hof and Schrama [15] to present a survey of many iterative closed loop identification schemes related to the problem of designing high-performance model based controllers for plants with unknown dynamics. They also presented the idea of separating the analysis of both identification and controller design and to have a joint performance criterion of both parts. They proposed that at each iteration data should be collected online, identification performed and the controller should be re-designed. If the controller satisfies some robustness criteria then it should be implemented otherwise a new identification should be performed. The same year Gessing and Lachuta [16] showed that in the case of a constant set point and an ARMAX model resulting from the discretization of a continuous time plant, the parameters of the plant could not be identified in closed loop system controlled with a minimum variance controller. They showed that varying set point has an essential influence on the identifiability of the plant and observed that there does exists a point in which parameter estimates remain accurate but later on the noise causes a slow drift (from the true valuses) of these parameters.

In 1996, Hjalmarsson et al. [17] compared open loop versus closed loop identification when the identified model is used for control redesign. The measure of the model based controller performance was the variance of the error between the outputs of the ideal closed loop system and that of the actual closed loop system. They showed that the optimal experiment setup is to identify the system with some optimal controller operating on the plant (closed loop). Landau and Karami [18] used a RST digital controller to present an iterative closed loop identification scheme. The objective of their scheme was to minimize the error between the true closed loop system (reference) and the designed (model) system by using new data acquired in the sequence of operations carried out. The error was then used to update the parameters of the plant model. They used Closed Loop Output Error (CLOE), Filtered Closed Loop Output Error (FCOE) and Extended Closed loop Output Error (XCLOE) identification algorithms. Gevers et al. [19] derived the asymptotic variance expressions for identified models based on several different closed loop identification methods and compared them to the respective expressions for the open loop situations. They also showed mathematically the consequences for the variance of resulting model based controller designs. These results demonstrated that all identification schemes for closed loop led to the same asymptotic variance expressions and that the controllers designed with closed loop identified model showed better variance results. Hof and Callafon [20] in 1996 compared the classic indirect closed-loop identification and the dual-Youla parameterization technique [15] and provided several relationships between the two approaches. They showed that dual-Youla parameterization technique, which guaranteed the identified model to be stable, was in fact a generalization of the indirect identification scheme. They highlighted the problem of controlling the model order when using these two techniques.

Robust control methodologies aim to design controllers guaranteed to meet the specifications not for a single nominal model, but for all models obtained by given perturbations of the nominal model. Such model set is called uncertainty model. With this in mind, Milanese et al. [21] showed that identifying a model by minimizing the discrepancy between the closed loop performance predicted by the model and the one actually achieved on the plant, is equivalent to finding the best approximated Youla parameterization (indirect method) of the plant in a suitably weighted H_∞ norm. Using this approach, they derived an optimal uncertainty model for the dual Youla parameterized plant and obtained an uncertainty model for the actual plant. This model was then used for designing a robust controller.

Ljung and Forsell [22] compared the statistical properties of a number of closed-loop identification methods and parameterizations. On comparison of asymptotic variances for the parameter vector estimates, they showed that the indirect method failed to give better accuracy than the direct method. They concluded that a directly applied prediction error method would give consistency and optimal accuracy even with closed-loop data, provided the noise model could describe the true noise properties. Sun et al. [23] proposed a new indirect identification algorithm for linear discrete time closed-loop system based on output-over sampling scheme, which did not require knowledge of the reference signal nor of the controller. However, they assumed that the structure of the plant was known in this scheme. Linard et al. [24] extended two linear methods for the identification of approximate models of an open loop plant on the basis of closed loop data to the nonlinear case. The first method was an indirect method based on identification of the sensitivity function [15] and the second method was right co-prime factor identification (joint input-output) method, which identified the sensitivity and complementary sensitivity function of the closed loop system. Yoneya et al. [25] proposed an iterative closed loop identification approach which employed a linear functional model. This model was used to iterate the closed loop system to a solution. This iterative scheme worked on the notion that if the model did not suit closed loop control, the model based controller behavior would display significant different characteristics from the one predicted with the nominal plant. From this difference a new plant model was identified and this procedure continues till a best controller performance was achieved.

Huang and Shah [26] discussed the accuracy aspects of identification and the role of filtering in closed-loop identification. They showed that the key difference between closed loop and open loop identification methods was the existence of the sensitivity function, which inversely affects the variance and bias errors of the estimate under closed loop conditions. They proposed a two-step (joint input-output) closed loop identification algorithm, which through the use of appropriate data filtering could estimate a suitable model from closed loop data.

Geverns et al. [27] presented a detailed paper in 1998, highlighting the role of feedback (closed loop) in the identification and validation of a model, which was to be used for control design. They examined the role of controller in changing the experimental conditions, effects of open loop and closed loop identification in terms of bias and variance errors in the context of identification for control. The same year Hof [28] presented a survey of all the direct and indirect algorithms that were either being used or modified for closed loop identification. According to him closed loop experimental conditions should not be considered as a degenerate or unfavorable situation for identifying dynamic systems. He evaluated characteristic properties of both direct and indirect identification methods on the basis of an explicit assessment criterion, including aspects of bias and variance.

Bruyne et al. [29] presented gradient expressions for closed loop identification scheme based on the minimization of a certain criterion and a parameterization that was tailored to the closed loop configuration. According to them, the main advantage of these gradient expressions was that they could easily be extended to non-standard identification criteria in which the plant, the parametric model and the controller could be nonlinear. Kulikov et al. [30] presented a modified version of least squares algorithm for online identification. The purpose of this algorithm was to estimate the delay of linear part of a digital model in continuous time and parameters of the numerator and denominator of the transfer function in discrete time. This scheme allowed for the correction of the delay in discrete time and was shown to be efficient for adaptive control systems design.

The advent of 1999 saw a lot of research being conducted in closed loop identification field. Sun et al. [31] in continuation of their earlier work [23], presented a new direct closed loop identification algorithm for an unstable discrete time linear system that was run by a feedback controller using only input-output data. This technique was based on the output inter-sampling scheme and did not require the reference signal to keep the Persistently Exciting (PE) property. Chou and Verhaegen [32] used indirect approach to identify a Wiener model of a high purity distillation column in closed-loop. Psadyn et al. [33] used subspace, partial least squares, ordinary least squares and output error approaches both for closed loop and open loop identification of a waste water reactor. They showed that in their case, open loop identification proved very effective and closed loop operational data could not be used for model identification regardless of the method (parametric or non-parametric). Forsell [34] in his thesis report provided a detailed description of methods for closed loop black-box identification of linear, time invariant dynamical systems given discrete time data. He focused mainly on prediction error methods and suggested modified versions of output error and Box-Jenkins model structures in the case of unstable systems. Ljung and Forsell [3] discussed closed loop identification approaches and showed that most of the common methods could be viewed as special parameterizations of the general prediction error methods. They also proposed a projection method for closed loop identification which allowed approximation of the open loop dynamics in a given and user chosen frequency domain norm. Ljung [36] in his book provided a lot of mathematical details related to closed loop identification techniques. Landau et al. [37] focused their research on recursive identification of nonlinear plants operating in closed loop with a nonlinear controller using closed loop output error (prediction error) identification schemes. An interesting aspect of their research was that they tried to show that a number of closed loop output error identification schemes could be used in nonlinear systems, which by itself was a notable effort. Gaspar et al. [38] motivated by different engineering problems caused by large uncertainties in the modeling of processes, presented a closed loop identification method based on the construction of Generalized Orthonormal Basis Functions (GOBF). This method utilized appropriately chosen basis functions generated by all-pass functions having poles close to the poles of the actual system.

Jin et. al. [39] in 2000, presented yet another approach by which, system transfer function was identified from closed loop data by using state space identification technique and correspondingly a LQG controller was redesigned. The same year Zheng [40] used Bias-Eliminated Least-Squares (BELS) method for direct identification of closed loop systems with colored noise. Sun et al. [41] extended the output inter-sampling based closed loop identification approach to the case where an unstable plant was disturbed by stochastic colored noise. They studied the time domain and frequency domain properties of the inter-sampled plant model and proposed several identification algorithms. Forsell and Ljung [42] used flexible, parameterized noise model in the prediction error method to eliminate the bias when applying an output error model with a fixed noise model/prefilter to a closed loop data. They termed this approach as projection method. However they showed that regardless of the fact that this method gave consistent estimates, the accuracy of it was sub-optimal. Schwarm et al. [43] presented a Model Predictive Control and Identification scheme (MPCI) that employed online optimization to perform closed loop identification and controller adaptation. This scheme was shown to identify single input single output (SISO) system efficiently while satisfying standard MPC constraints and keeping the process output robustly within specified bounds. Hof et al. [44] presented a GUI (Graphical User Interface) based CLOSID toolbox for matlab, to be used for the identification of linear systems on the basis of experimental data.

Recently in 2001, Ooi and Weyer [45] used direct closed loop identification approach for irrigation channels using a lead lag controller. In this case the water level was the controlled variable and gate position was the manipulated variable. Zheng [46] presented a new algorithm based on the combination of least squares and bias correction principle, for direct identification of closed loop plants. Zheng [47] applied conventional least squares estimation technique to obtain closed loop parameter estimates of the process model and then adjusted these parameters in order to remove the bias caused by colored noise. This proposed method was called Bias Eliminated Least Squares Method with No Prefiltering (BELSNP). Continuing with this work Zheng [2] in 2002, applied a more improved version of his proposed BELSNP method that could be used for indirect identification of transfer function models for unstable plants in closed loop. However, this method required prior knowledge of low order regulators. Klerk and Craig [48] provided an overview of different closed loop identification techniques with a simulation example. The purpose of which was to motivate new researchers to enter into this field. Keviczky and Banyasz [49] compared different variations of the Youla parameterization schemes (indirect method) including the K-B parameterization. They introduced R and S parameterization as a new form of Youla parameterization schemes. Wang and Yin [50] presented their findings on timing complexity problem that occur in the identification of unstable, non-minimum phase and time varying plants operating in closed loop. Eker and Nikalaou [51] in continuation of their earlier work [43] presented a rigorous study of their MPCI technique. Issues related to stability and convergence properties were discussed. Leskens et al. [52] presented an application of a specific system identification procedure to a municipal solid waste (MSW) incinerator. The proposed procedure was a combination of two-stage (joint input-output) closed loop identification method as well as the approach of high-order multiple-input-multiple-output (MIMO) ARX model estimation followed by model reduction. Katayama et al. [53] performed closed loop identification of the deterministic part of the process in the framework of joint input-output approach. They used orthogonal decomposition technique to decouple the deterministic part from the stochastic part of the process. They obtained the state space models of the plant and the controller by applying a standard subspace method to the deterministic component of the joint process.

1.3 Motivation for Present Work

Dynamic models play a central role in MPC technology. The assumption during MPC design is that a reliable model of the plant under consideration is available. Normally identification methods deliver a nominal model of the plant with unknown dynamics. The performance achieved by this controller when applied to the plant is highly dependent on the accuracy of this model. However, in real life situations, models that are identified from open loop data are generally contaminated with errors and are inaccurate for many reasons such as equipment degradation (e.g. catalyst change, heat exchanger fouling etc.), low quality measurement data etc. Problem also arises when these models start exhibiting degradation in their performance after some span of time. A number of reasons can be attributed to this phenomenon like change in process operating conditions, drift in process conditions, environmental conditions, instability and inherent feedback mechanisms of the plant. Examples of such processes are refineries, where a change in the crude oil flow can change the entire dynamics of the plant model, and high purity distillation columns where the correlation between different controlled variables makes it difficult for the model to represent the process for a long span of time. The only practical solution existing in the industry today is to shut down the process and identify the model again, resulting in huge financial and production losses.

This problem of process identification can be solved by using closed loop techniques to identify the model. For this reason much research is being done on closed loop identification. According to Veres [54] identification for the purpose of controller design is best achieved when the process is operating in closed loop under an optimal controller. The phenomenon that the operating controller helps the identification of a model that is good for the controller itself is called synergy between control and identification. This means that not only a model can help control design but vice versa. A significant amount of research has been done on closed loop identification but not for MPC. The reasons associated with this fact are that many of the closed loop identification techniques proposed in the literature are based on the assumption that the existing controller is linear and the process is single variable whereas MPC is inherently nonlinear due to constraints imposed on it and the process under control is mostly multivariable. Secondly MPC is not a structure like other classical controllers but is an algorithm that is programmed to run a certain task under desired constraints. For this reason closed loop identification methods based on joint input-output and indirect techniques cannot be used for MPC as complete knowledge of the controller is not possible.

These issues motivated this work in which the objective is to show that the plant model can indeed be identified in closed loop with the MPC running (online) using direct identification techniques. Identification techniques based on least squares method, prediction error method, subspace method and neural network method are examined in this regard.

1.4 Thesis Contributions

In this thesis, direct closed loop identification techniques are studied for MPC. An effort to bring fresh perspective to this area is made. Different identification schemes for use in closed loop identification for MPC have been quantified. ARX, ARMAX, state space and OE models are used in these schemes. A multivariable Demethanizer column process from a gas plant in Saudi Arabia has been used as a generic case study. The contributions can be enumerated as follows:

- Model for Demethanizer column process is identified from open loop data collected for a month by using single variable step testing at a sampling time of 1 minute.
- For the purpose of collecting simulated closed loop data, a high performance MPC comprising of 4 Controlled Variables (CV's) and 4 manipulated variables (MV's) is designed for this complex Demethanizer column process, using the open loop identified model.
- Different modeling techniques based on least squares, prediction error and, subspace identification methods have been tested for their accuracy and consistency for use in closed loop identification for MPC process.
- Closed loop field data is collected for the months of November, December and January 2003 at a sampling time of 1 minute.
- Best modeling techniques are highlighted and are tested on actual closed loop field data.
- Recommendations are made on the basis of these simulations.
- Neural networks based NNARX model is used to estimate the Demethanizer column process from open loop data. It is shown through simulation that this model retains its accuracy in a global sense, that is, it does not change and loose its accuracy when tested with closed loop field data.

1.5 Thesis Organization

This thesis is organized as follows.

To make the reader familiar with the MPC concepts and terminology, Chapter 2 begins with an overview of MPC and key concepts related to it. This is followed by a brief description of the Demethanizer column process under study and simulation results for the open loop model identification are presented. MPC is then designed for the Demethanizer column process using this open loop identified model. Closed loop (simulated) data is then collected from this designed MPC process.

Chapter 3 deals with closed loop identification from simulated data. Simulation results are presented and the best identification/modeling schemes are selected.

In chapter 4, the selected schemes are tested for their performance on real field data from the Demethanizer column process running with MPC.

In Chapter 5, neural networks are introduced. MFNN are used to model the Demethanizer column process from the open loop data. Simulation results on the performance of this model are presented in this chapter.

Chapter 6 presents the conclusions and avenues for future work.

Chapter 2

Design of MPC – Demethanizer Column

2.1 Introduction

The term MPC represents a family of model based controllers. The MPC family of algorithms are designed on the basis of a multi-step optimization objective. In general, several controllers' moves in the future are computed but only the first control action is implemented, hence these controllers are also referred to as receding horizon controllers. The earlier versions of MPC are – the identification and command algorithm (IDCOM) proposed by Richalet et al. [55] in 1978 and the dynamic matrix control (DMC) algorithm due to Cutler and Ramaker [56] in 1980. Other well known variations of MPC include: model algorithmic control (MAC) by Rouhani and Mehra [57], multivariable optimal constrained control algorithm (MOCCA) from Sripada and Fisher [58] and generalized predictive control (GPC) by Clarke [59].

The underlying philosophy of MPC type control algorithms differs from conventional PID controllers in several aspects.

- An explicit model of the process is used within the control algorithm to determine the control actions at every step based on the minimization of a cost function.

- It is not restricted to single-input, single output (SISO) processes and can be derived for and applied to multi-input, multi-output (MIMO) processes.
- The family of controllers has the ability to deal with hard constraints on the inputs and outputs in an optimal way. This represents a significant step in terms of practical implementation. The computational complexity of the optimization step is restricted to a linear or quadratic program in the worst case. Thus these algorithms can be easily implemented on-line.
- At each sampling instant several control actions are calculated, only the first control move is implemented. These controllers are thus known as receding horizon controllers.
- In contrast to PID controllers, predictive controllers can also be derived for non-linear and multivariable processes. The concept of predictive control 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 system with large time delay, processes that are non-minimum phase and processes that are unstable.

Simplicity of design combined with its ability to tackle realities such as constraints and interactions has helped MPC achieve its current popularity with the process industry. The industrial success of these algorithms spurred the growth of MPC as a research area in academia.

In this chapter, an overview of the key ideas involved in the classical model predictive control with a tutorial flavor is presented. Section 2.2 illustrates the basic concepts related to MPC technology as depicted in Fig. 2.1. Control objective in its commonly used forms and constraints are also high lighted with respect to their type.

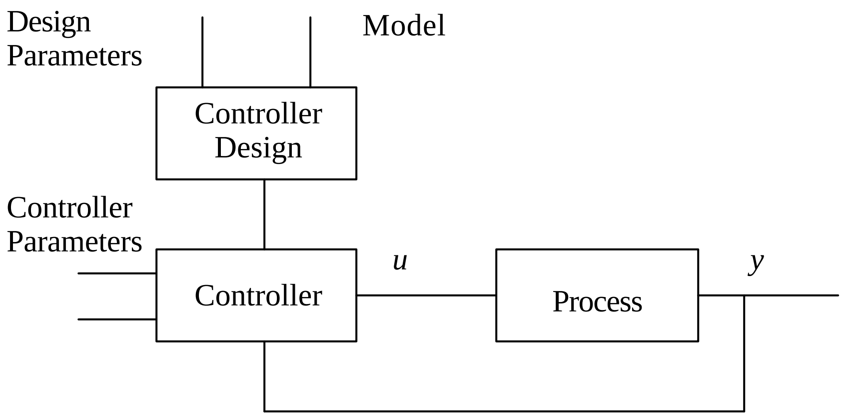

Figure 2.1: Model predictive control scheme

2.2 The Predictive Controller Concept

In model predictive control technique, the dynamic optimization problem is solved on-line at each control execution. Process inputs are computed so as to optimize future plant behavior over a time interval known as the *prediction horizon*. In the general case any desired objective function can be used. Plant dynamics are described by an explicit process *model* which can take, in principle, any required mathematical form. Process input and output constraints are included directly in the problem formulation so that future constraint violations are anticipated and prevented. The first input of the optimal input sequence is injected into the plant and the problem is solved again at the next time interval using updated process measurements.

The various implementations of MPC preferred by the different vendors and users are identical in their main structure, but differ in details. These details are largely proprietary and are often critical for the success of the algorithm in an application [60].

The general structure of a process running with MPC is shown in Fig. 2.2. A model is used to predict the future plant outputs, based on the past and current values and on the optimal future control actions. These action are caluclated based on an optimization algorithm that minimizes the performance index subject to the given constraints. Further if there is disturbance and noise present, a disturbance model can be added, thus allowing the effect of the disturbance to be taken into account.

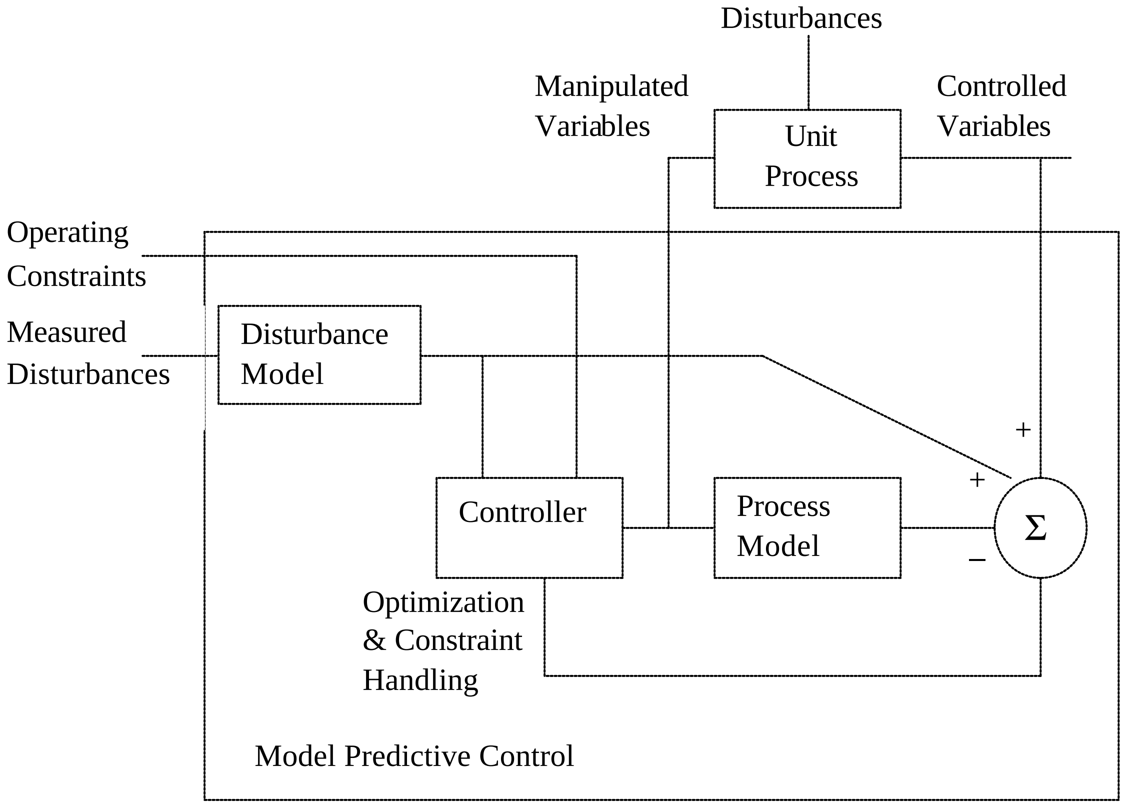

Figure 2.2: Model predictive control system

The methodology of all controllers belonging to MPC family is charcterized by the the receding horizon strategy as follows:

The future outputs for a determined horizon H_p called the prediction horizon, are predicted at each sampling instant k which denotes the time scale. These predicted outputs denoted by $\hat{y} = [\hat{y}(k+1), \hat{y}(k+2), \dots, \hat{y}(k+H_p)]^T$ are dependent on the future control moves given by $u = [u(k), u(k+1), \dots, u(k+H_p-1)]^T$ which are to be calculated and sent to the system. A sequence of these future control moves is calculated by optimizing a criterion in order to keep the process as closed as possible to the reference trajectory, $w = [w(k+1), w(k+2), \dots, w(k+H_p)]^T$, which can be a set point itself or a close approximation of it. This criterion usually takes a form of a quadratic function of the errors between the predicted output signal and the reference trajectory. Such a simple criterion function is described as follows [61].

$$J = \sum_{i=1}^{H_P} (\hat{y}(k+i) - w(k+i))^2 \tag{2.1}$$

In some controllers the criterion function is augmented with some weighting factor terms penalizing particular components of *y* or *u* at certain future time intervals [62].

$$J = \sum_{i=1}^{H_P} \left\| \hat{y}(k+i) - w(k+i) \right\|^2_{Q(i)} + \sum_{i=1}^{H_C} \left\| \Delta u(k+i-1) \right\|^2_{R(i)} \tag{2.2}$$

where,

$$\Delta u(k+i-1) = u(k+i) - u(k+i-1) \tag{2.3}$$

Q and R are weighting matrices to penalize particular components of y and u at a certain future time interval, $w(k+i)$ is the vector of future reference values (set points), H_p is the prediction horizon and H_c is the control horizon (time after which the input is held constant). Here the weighting matrix R is often called suppression factor since increasing it penalizes changes in the input vector more heavily (increasing in the value of R results in smaller changes in the ?u's). This cost function was first used in DMC by

Cutler et al. [56] in 1980, which went on to become one of the most well known of the commercial predictive control products. DMC was conceived to tackle multivariable constrained control problems typical for the oil and chemical industries and today there is probably not a single major oil company in the world, where DMC is not employed in most new installations or revamps [63].

After the future controller moves sequence is predicted, the first element of the sequence *u(k)* is sent to the process while the other control moves are rejected. At the next sample, $k+1$, the whole cycle of output measurement and input sequence prediction is repeated using the latest measured information. This is called receding horizon principle as shown in Fig. 2.3. Assuming that there are no disturbances and no modeling error, the predicted process output $\hat{y}(k+1)$ is exactly equal to the process output. In general, this controller output sequence is different from the one obtained at the previous sample.

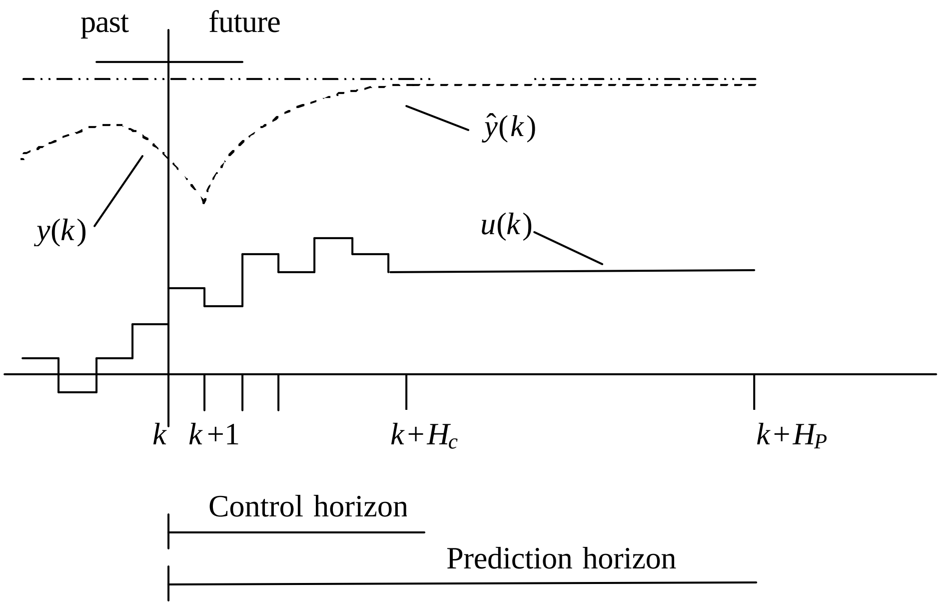

Figure 2.3: Definition of optimization problem for MPC

The advantage of using the receding horizon strategy is that future constraint violations are anticipated and prevented. In other words, the predicted output of the process is close to the desired process output.

2.2.1 Constraints

A real plant has to work with certain physical limitations e.g. a value which can handle only a particular range of flow rates, and market forces which result in rigid quality requirements on the process outputs. Usually a real process involves rate and amplitude constraints on the input, and may also require outputs constrains to be considered,

$$
\begin{aligned}
\Delta u_{\min} &\leq \Delta u(k) \leq \Delta u_{\max} &&: \text{Rate Constraints} \\
u_{\min} &\leq u(k) \leq u_{\max} &&: \text{Amplitude Constraints} \\
y_{\min} &\leq \hat{y}(k) \leq y_{\max} &&: \text{Quality Constraints}
\end{aligned}
$$

In practice, most of the processes are nonlinear. The most common nonlinearites are constraints on the input of the process, or equivalently, constraints on the output of the controller. Qin et al. [64] categorized these into two types for MPC technology: hard and soft. Hard constraints are those which should never be violated i.e. no-violation of the bounds are allowed at any time. Soft constraints are those for which some violation is allowed i.e. violations of the bounds can be allowed temperorily for the satisfaction of other criteria. Most often soft constraints can be taken care of by the minimization of the criterion function. For instance, the requirement is that the controller output must be held between an upper and a lower limit. This constraint may be considered as a soft constraint i.e. a temperory violation of this constraint is allowed if it is required, for example, to drive the process output after a load change has occurred. Then the weighting factor in Eq. 2.2 can be selected such that the soft constraint is satisfied most of the time.

In constrast to soft constraints, hard constraints cannot be handled by minimization of Eq. 2.2 only. They must be taken into account explicitly when minimizing the criterion function. This results in constrained minimization problem.

2.2.2 Controller Tuning

The prediction and control horizons (H_p and H_c) and weights (Q and R) in the criterion of Eq. 2.3 effect the bahaviour of the closed-loop combination of the plant and the predictive controller. Usually these are referred to as the adjusting parameters and are in effect 'tuning parameters' which are adjusted to give satisfactory dynamic performance. Increasing the weights R on the control moves relative to the weights Q on the tracking errors has the effect of reducing the control activity. Increasing theses weights indefinitely will reduce the control activity to zero, which 'switches off' the feedback action. If the plant is stable, this will result in a stable closed loop system, but not otherwise [61]. The penalty of increasing the control weigthing will be slow response to disturbances, since only small control actions will result. On the otherhand, some processes may require no penalty on the control moves i.e. $R = 0$.

In a similar manner, decreasing the weights Q mean that the tracking error in the near future is less impotant than in the far future, yielding less active control moves. This choice is usually motivated by the fact that a real process cannot track a set point change in, for example, one sample. As a result the tracking error can be quite large in the first few samples (near future). By not including these large tracking errors in the optimization criteria, the controller will not attempt to make them smaller. Hence smooth control is achieved. When $Q = 1$ the tracking error is equally weighted over the prediction horizon.

The selection of of H_p and H_c is a compromise between the robustness and performance of the process. Increasing H_p improves the robustness of the process at the cost of slower process response to set point changes. Similarly incresing H_c results in reduced process robustness but better performance.

2.3 Demethanizer Column

This Demethanizer column is a part of a NGL (natural gas liquids) gas plant in Saudi Arabia and consists of 19 Koch valve trays. Its primary function is to remove light hydrocarbons out of the feed gas, condensed in the three chills down trains. There are two pumps at the bottom of the column. Both these pumps have three stages associated with them. In the first stage they take the suction from the Demethanizer sump at a temperature between 25 and 30 ^{o}F. In the second stage they increase the liquid pressure from 160 to 300 psig and in the last stage they send it to the NGL surge sphere in the product surge unit. Additionally there are two Demethanizer reboiler pumps. Their primary function is to take the suction from tray 1 at a temperature between –15 and –10 ^{o}F, pump this liquid around the Demethanizer reboiler system and return it back to the column below tray 1.

The Demethanizer column has four controlled variables (outputs) and four manipulated variables (inputs). In addition to these, there are two measured disturbances associated with this column. They are a follows:

Output variables (CV's):

- Bottom C1 over C2

- LP residue gas pressure Setpoint

- Demethanizer pressure differential

- Demethanizer Tray 6 temperature

Input variables (MV's):

- LP residue gas valve opening

- Jump-Over valve opening

- Trim re-boiler valve opening

- Tray 6 bypass valve opening

Disturbance variables:

- Ambient temperature

- Feed compressor discharge pressure

This process can be explained by Fig. 2.4. Here the controlled variables are represented by y_1 (Output-1), y_2 (Output-2), y_3 (Output-3), and y_4 (Output-4). The manipulated variables are represented by u_1 (Input-1), u_2 (Input-2), u_3 (Input-3),

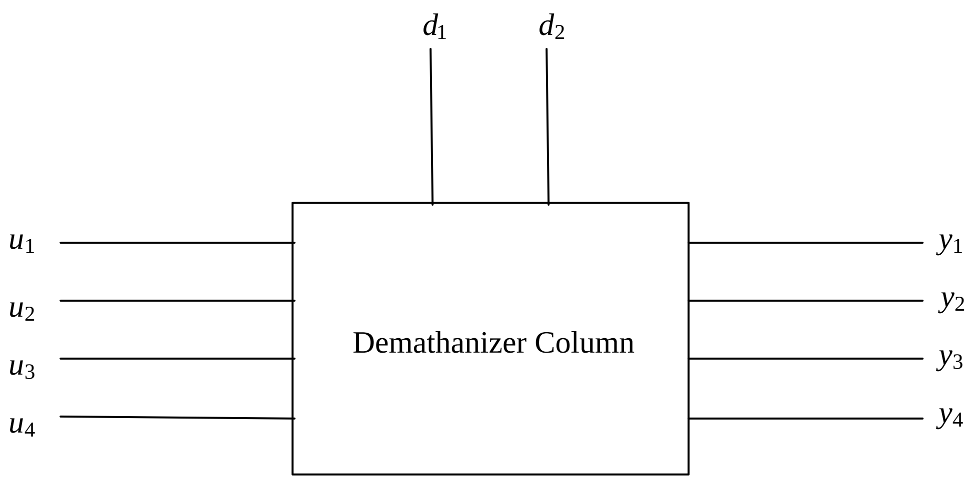

Figure 2.4: Block diagram representation of Demethanizer Column

u_4 (Input-4) and measure disturbance variables are represented by d_1 (Disturbance-1) and d_2 (Disturbance-2).

2.4 Open Loop System Identification

System identification is the task of constructing mathematical models of dynamical systems from measured data. It involves four basic steps namely experiment design, selection of a suitable model structure, parameter estimation and model validation. Experiment design involves issues like choice of which signals to measure, choice of sampling time, choice of excitation signals. Once these issues have been settled, the actual identification experiment can be performed and process data be collected. The next step is to decide on a suitable model structure. This is a crucial step in the identification process and to obtain a good and useful model, this step must be done with care. Once a suitable model structure and measured data is obtained, the actual estimation of the model parameters is performed. Before the model is finalized, it has to pass some validation test. Model validation can loosely be said to deal with the issue of whether the identified model is also "good enough" for its intended use. Common validation procedure is so called, cross-validation, where the model is simulated using "fresh" data and the output is compared to the measured output. If the first model fails to pass the validation tests, some, or all, of the above steps have to be iterated until a model that passes the validations tests is found.

For the Demethanizer column, one month open loop data is collected at a sampling time of 1 minute using step testing. The step tests are typical in MPC projects. They generate good data with good enough signal-to-noise ratio but do not disturb the actual process operation. A total of 9434 data samples are gathered for the fours inputs, four outputs and two measured disturbances. Different identification methods such as least squares, subspace, prediction error etc. are employed to identify a 6-input-4-output model (including the disturbance model) for this column. Among these the subspace method is found to give the best possible identified model and is therefore selected. The simulation results for this step are presented in the following subsection.

2.4.1 Subspace Identification Method

In the category of state space model identification schemes, the most commonly used algorithm is N4SID short for Numerical algorithm for Subspace State Space System Identification. It was proposed by Peter Van Overschee and De Moor [65] in 1994. It is can be summarized as follows:

Let $u_k \in \Re^m$, $y_k \in \Re^l$ be the observed input and output generated by the unknown system described by Eqs. 2.4 and 2.5.

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{2.4}$$

$$y_k = Cx_k + Du_k + v_k \tag{2.5}$$

or in innovation form

$$x_{k+1} = Ax_k + Bu_k + Ke_k \tag{2.6}$$

$$y_k = Cx_k + Du_k + e_k \tag{2.7}$$

where the vectors $u_k \in \Re^m$ and $y_k \in \Re^l$ are the measurements at time instant k of m inputs and l outputs of the process respectively. The vector x_k is the state vector of the process at discrete time instant k, $v_k \in \Re^{l\,x1}$ is called the measurement noise and $w_k \in \Re^{n\,x1}$ is called the process noise. K is the Kalman gain.

Given a large number of measurements of the inputs $u_k \in \Re^{m}$ and outputs $y_k \in \Re^{l}$ generated by the unknown system of Eqs. 2.4 and 2.5, the problem can then be defined as of simply determining the order n of the system, the system matrices $A \in \Re^{n\,x\,n}$, $B \in \Re^{n\,x\,m}$, $C \in \Re^{n\,x\,l}$, $D \in \Re^{l\,x\,m}$ up to within a similarity transformation and if required K, without any prior knowledge of the structure of the system. Fig. 2.5 gives a block representation of this algorithm. The left hand side shows the subspace method approach where the (Kalman filter) states are first estimated directly from input and output data, then the system matrices are obtained. The right hand side shows the classical approach in which the system matrices are found first and then the estimate of the states. The usual steps involved in N4SID algorithm are described in Appendix B of this thesis.

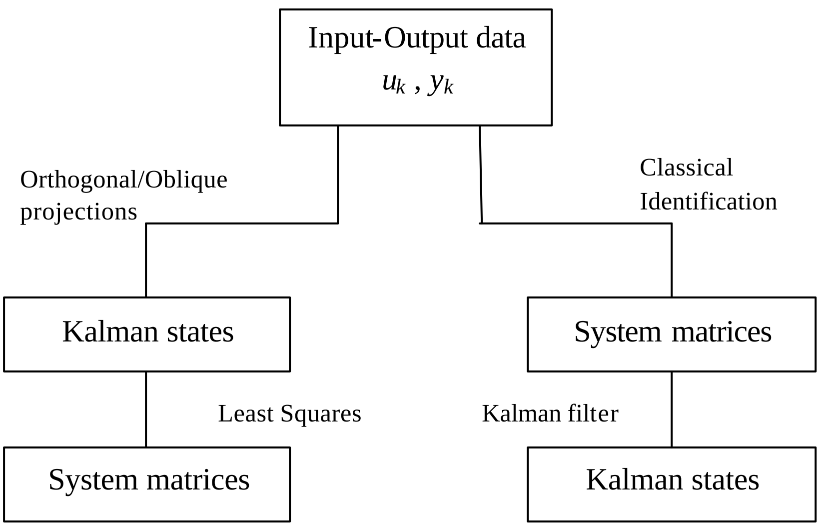

Figure 2.5: Numerical algorithm for subspace state space system identification

2.4.2 Simulation Results

The state space model identified using the N4SID (subspace) algorithm is given in Eqs. 2.8 - 2.11 for the four outputs respectively. The identified model shall be used as benchmark through out this thesis. For the purpose of identification the data is first pre-filtered to remove bad data and outliers. 70% of the data samples are used for identification and the remaining 30% are used for validation. The goal of model validation is to ascertain whether the identified model is good enough to represent the process. The simulation results are shown in Figs. 2.6 - 2.9 in which the dotted line is the model response and the solid line is the actual response. The results show a very good performance of the identified open loop model (dashed), considering the presence of high noise and nonlinearity in the actual data (solid). The model delivers almost accurate results for the first two outputs. The third and fourth identified outputs exhibit a small error due to the presence of high disturbance and noise in the original data. However, the models obtained for the four outputs are by far the best ones that can be achieved by state space modeling technique. This can be seen from the corresponding distribution of prediction errors are shown in Figs. 2.10 - 2.13. These show that the errors are mostly between ± 0.4 for all outputs. The step responses for the four outputs corresponding to the four inputs respectively are shown in Figs. 2.11 - 2.26, for 700 sampling instants. It is noticed that the settling time for the process is quite large.

As said before, the quality of the state space model identified by using subspace methods is exceptionally good and is the best amongst all the identification methods, the results of which are irrelevant to this work. This model is from here on referred to as open loop process model and will be used as a benchmark in the analysis of closed loop identification techniques in the next chapters.

$$\boldsymbol{x}(k+1)=\begin{bmatrix} 0.9723 & -0.0838 & -0.1546 & 0.3243 \\ -0.0099 & 0.9737 & -0.2652 & 0.0564 \\ 0.1891 & 0.1703 & 0.4192 & 0.1052 \\ -0.2330 & -0.1456 & -0.7164 & 0.5624 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix}$$

$$+\begin{bmatrix} -0.1237 & -0.1218 & -0.0431 & 0.0764 & 0.0214 & 0.0104 \\ -0.1146 & -0.0117 & -0.0117 & -0.0590 & 0.0062 & -0.0289 \\ -0.2898 & 0.0570 & -0.0495 & -0.1599 & 0.0069 & -0.0899 \\ -0.1819 & 0.0783 & 0.0622 & -0.04175 & -0.0117 & -0.1020 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \\ d_1(k) \\ d_2(k) \end{bmatrix}$$

$$y_1(k)=\begin{bmatrix} 0.3061 & 0.0199 & -0.5983 & -0.8061 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \tag{2.8}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix} 0.9090 & 0.1917 & -0.0143 \\ 0.0973 & 0.7970 & -0.3054 \\ 0.0371 & -0.0580 & 0.7801 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}$$

$$+\begin{bmatrix} -0.2959 & -0.0482 & -0.0142 & 0.0150 & -0.0147 & 0.0214 \\ 0.4235 & 0.0078 & 0.0694 & -0.0574 & -0.0664 & -0.1580 \\ 0.1707 & -0.0117 & 0.0423 & -0.0310 & -0.0486 & -0.0944 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \\ d_1(k) \\ d_2(k) \end{bmatrix}$$

$$y_2(k)=\begin{bmatrix} -0.3907 & 0.2538 & -0.2969 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \tag{2.9}$$

$$
\boldsymbol{x}(k+1)=\begin{bmatrix} 0.9656 & 0.2192 \\ -0.0016 & 0.9824 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}
$$

$$
+\begin{bmatrix} 0.0228 & 0.0414 & -0.0091 & 0.0032 & 0.0466 & 0.0438 \\ 0.0033 & 0.0048 & 0.0010 & -0.0006 & -0.0068 & 0.0017 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \\ d_1(k) \\ d_2(k) \end{bmatrix}
$$

$$
y_3(k)=\begin{bmatrix} 0.3209 & -0.4040 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \tag{2.10}
$$

$$
\boldsymbol{x}(k+1)=\begin{bmatrix} 0.1966 & -0.2198 & 0.2530 & 0.2750 \\ -0.0403 & 0.9721 & 0.2439 & 0.0553 \\ -0.1079 & -0.0707 & 0.8406 & -0.3715 \\ 0.2511 & -0.0431 & -0.2477 & -0.1234 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix}
$$

$$
+\begin{bmatrix} -0.13985 & 0.9153 & 0.0976 & -0.1497 & 0.1644 & 0.6754 \\ -0.1694 & 0.1218 & 0.0131 & -0.2689 & -0.0045 & -0.0291 \\ 0.6498 & -0.3888 & 0.0107 & 0.6870 & -0.0616 & -0.2496 \\ 2.5584 & -1.4989 & -0.0277 & 2.5726 & -0.3230 & -1.2044 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \\ d_1(k) \\ d_2(k) \end{bmatrix}
$$

$$
y_4(k)=\begin{bmatrix} 0.5983 & -0.0927 & 0.3952 & 0.2395 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ x_4(k) \end{bmatrix} \tag{2.11}
$$

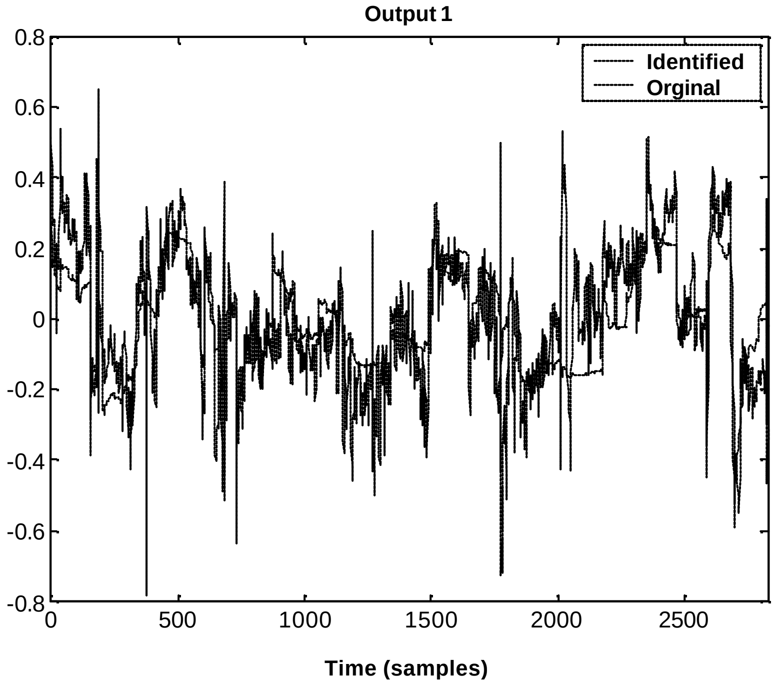

Figure 2.6: Open loop identified (dashed) and actual (solid) Output-1

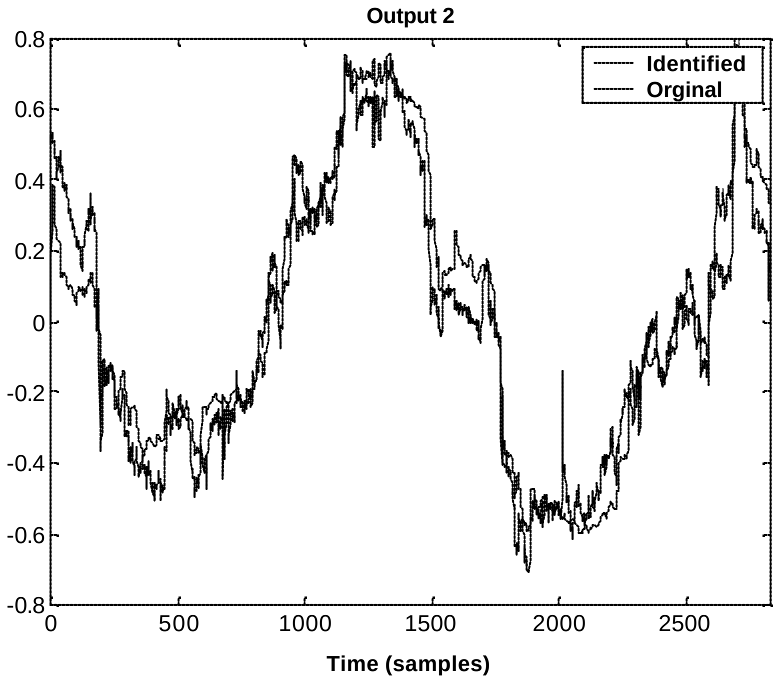

Figure 2.7: Open loop identified (dashed) and actual (solid) Output-2

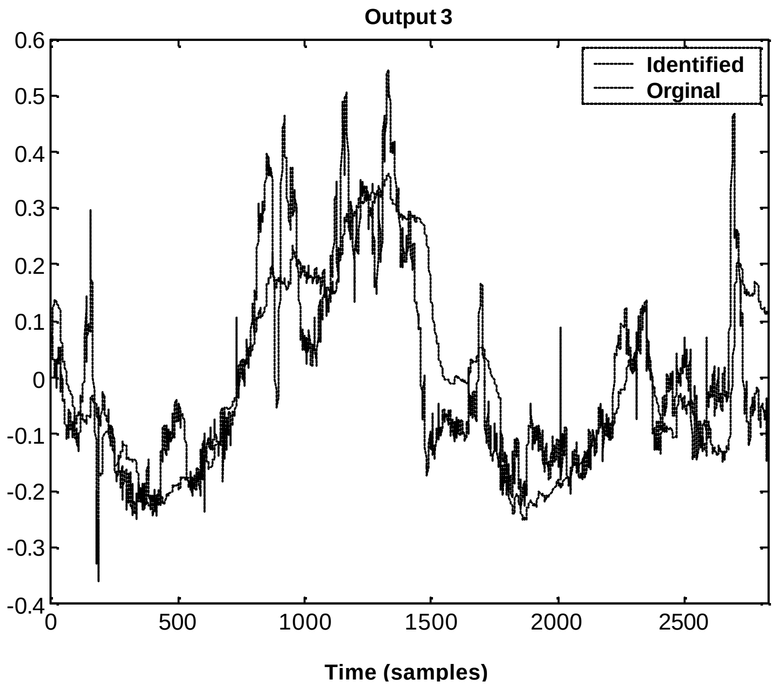

Figure 2.8: Open loop identified (dashed) and actual (solid) Output-3

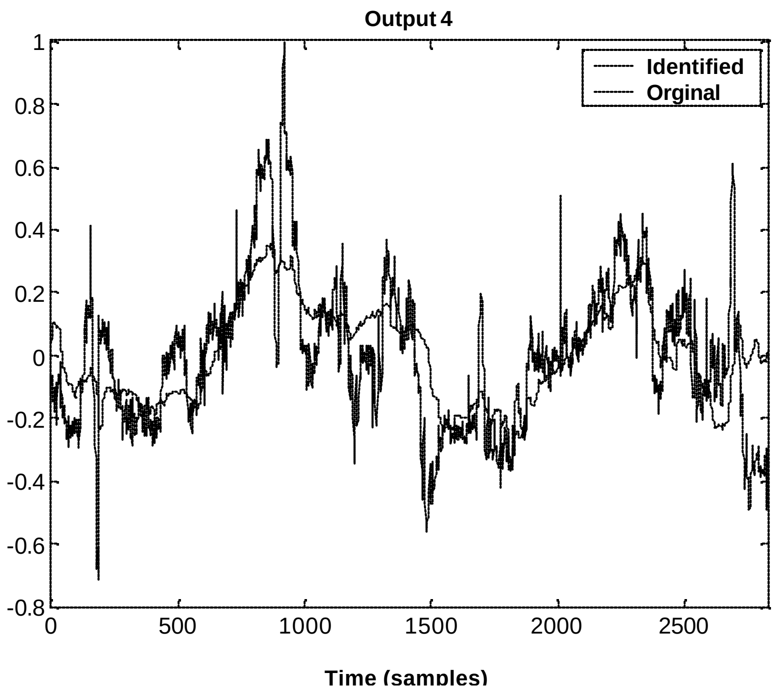

Figure 2.9: Open loop identified (dashed) and actual (solid) Output-4

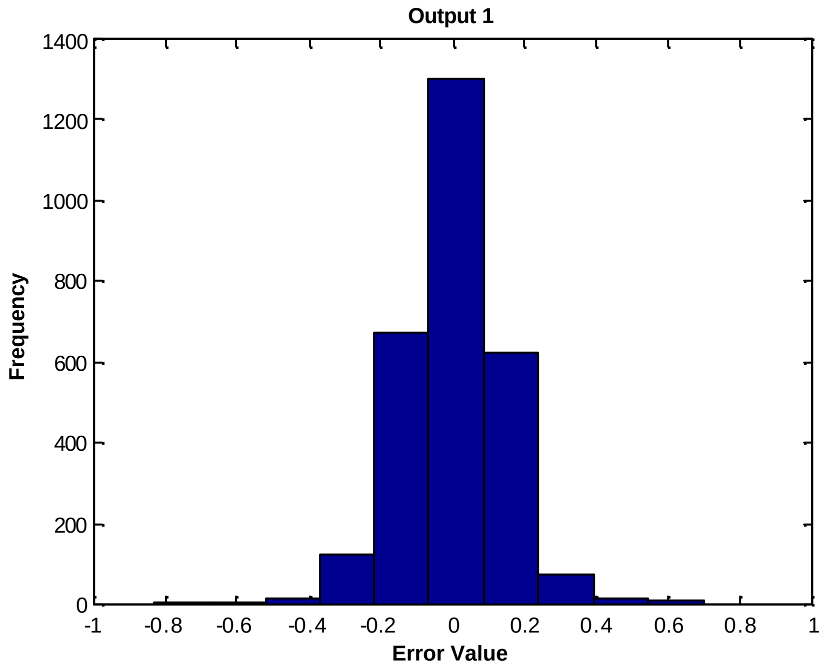

Figure 2.10: Prediction Error Distribution for Output-1

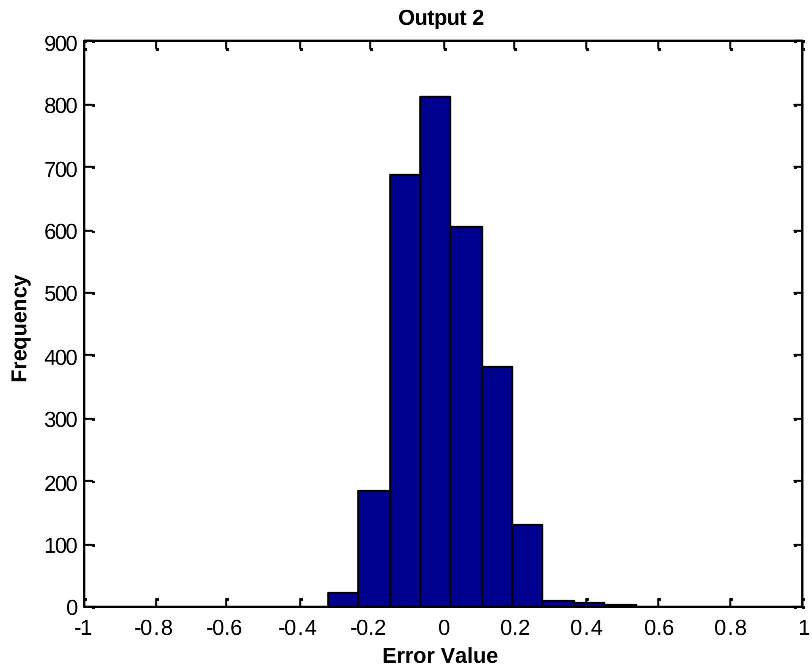

Figure 2.11: Prediction Error Distribution for Output-2

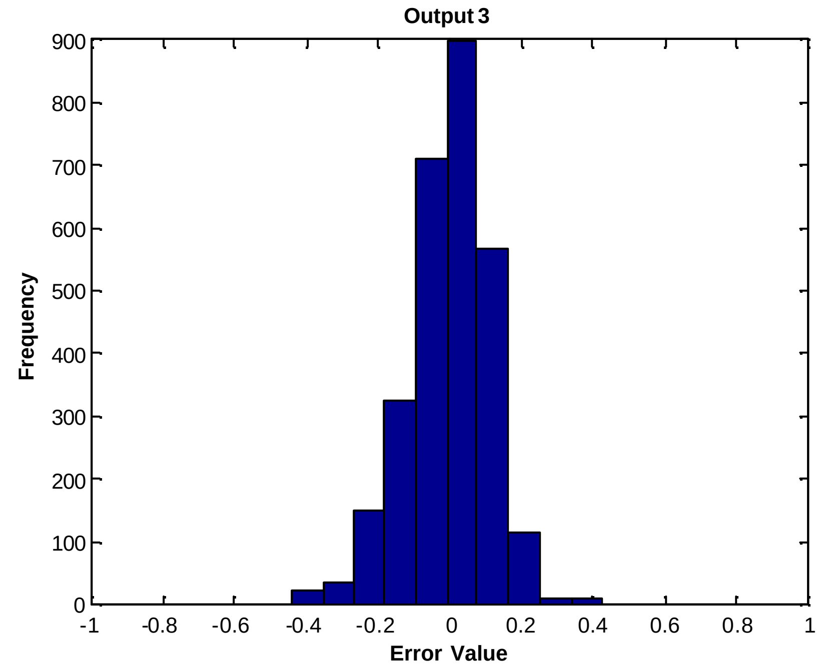

Figure 2.12: Prediction Error Distribution for Output-3

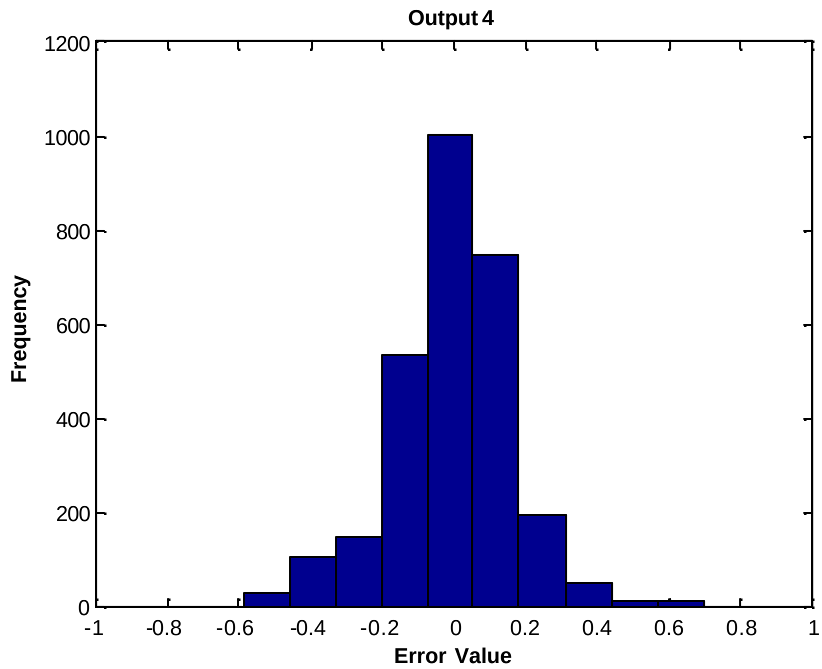

Figure 2.13: Prediction Error Distribution for Output-4

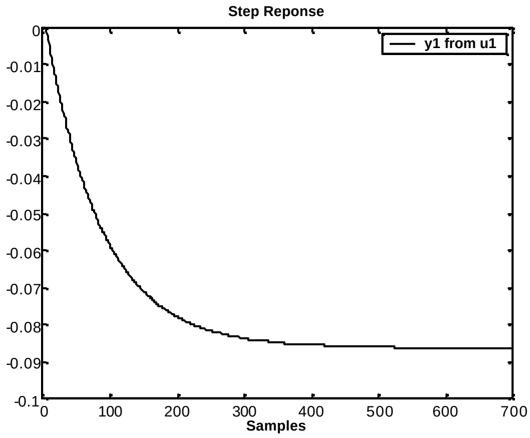

Figure 2.14: Unit step response of Output-1 from Input-1

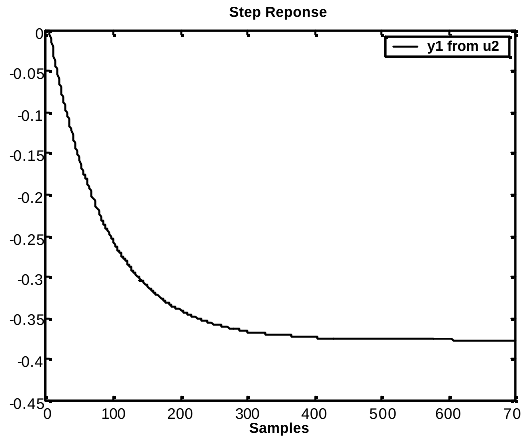

Figure 2.15: Unit step response of Output-1 from Input-2

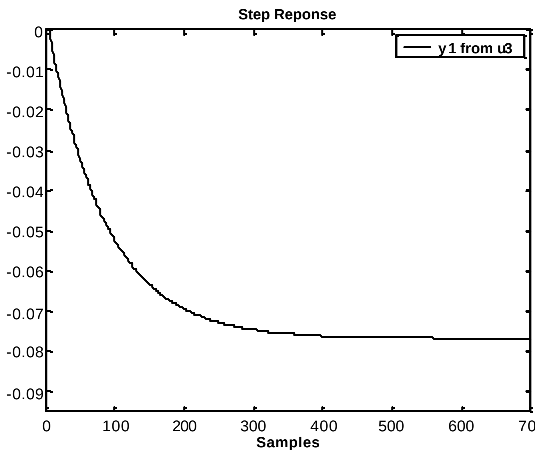

Figure 2.16: Unit step response of Output-1 from Input-3

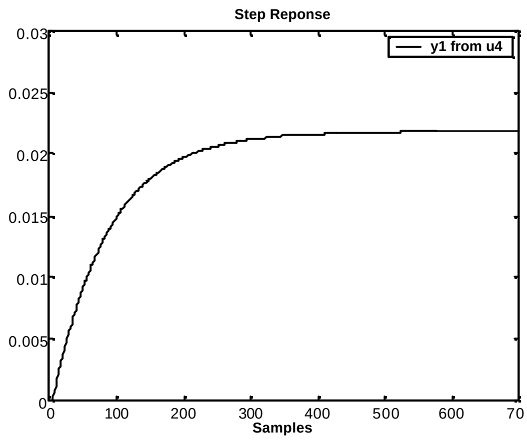

Figure 2.17: Unit step response of Output-1 from Input-4

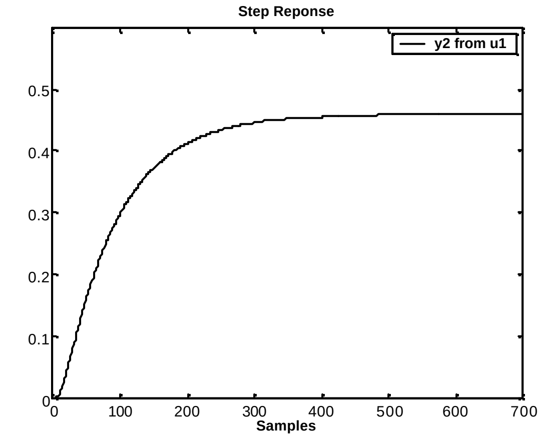

Figure 2.18: Unit step response of Output-2 from Input-1

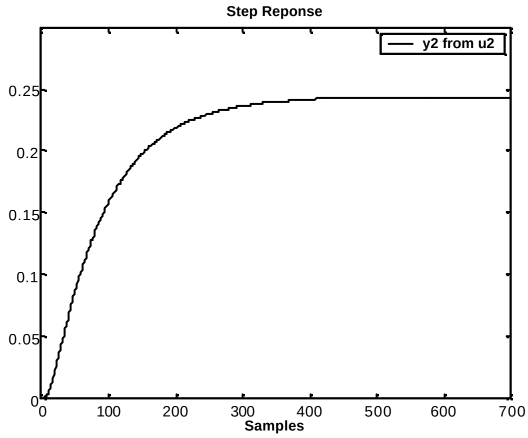

Figure 2.19: Unit step response of Output-2 from Input-2

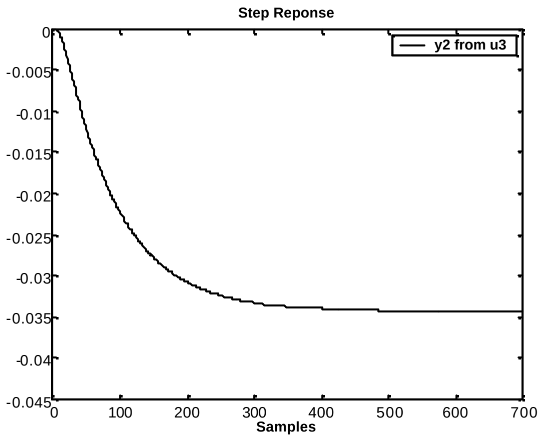

Figure 2.20: Unit step response of Output-2 from Input-3

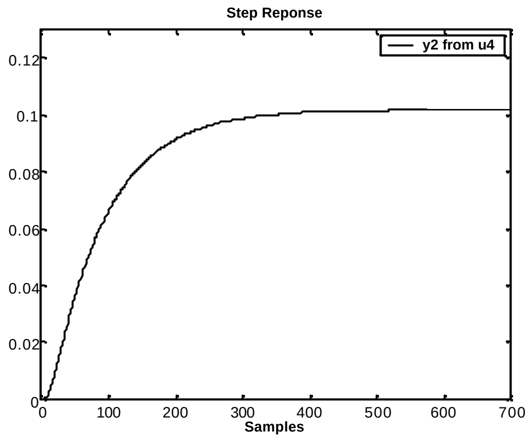

Figure 2.21: Unit step response of Output-2 from Input-4

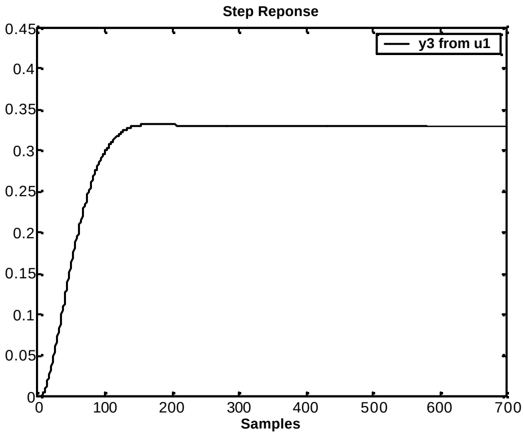

Figure 2.22: Unit step response of Output-3 from Input-1

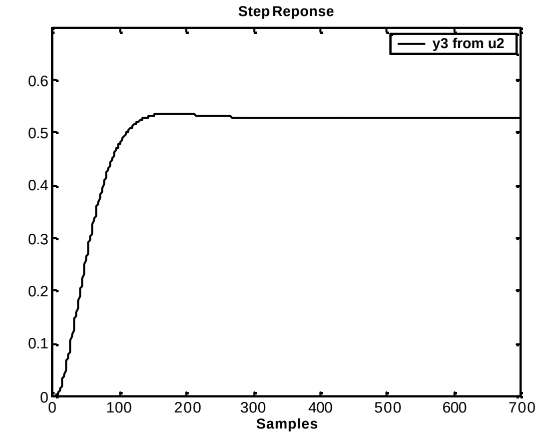

Figure 2.23: Unit step response of Output-3 from Input-2

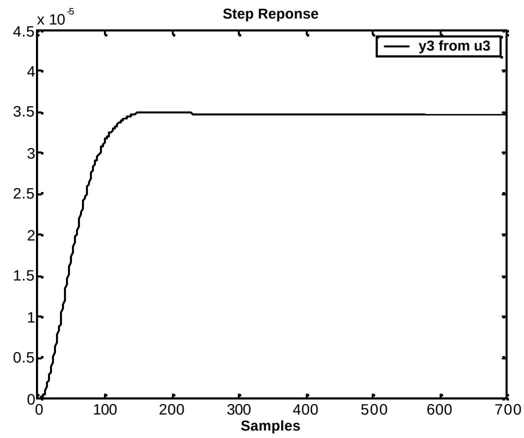

Figure 2.24: Unit step response of Output-3 from Input-3

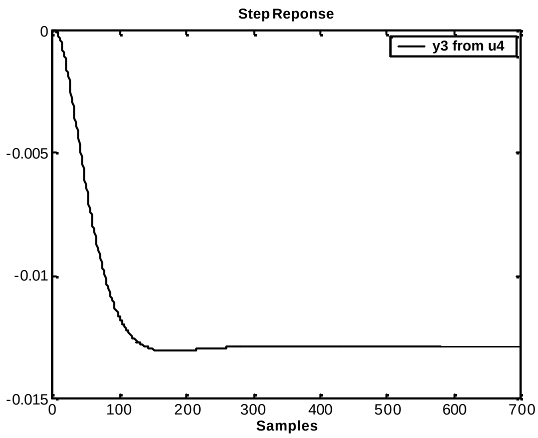

Figure 2.25: Unit step response of Output-3 from Input-4

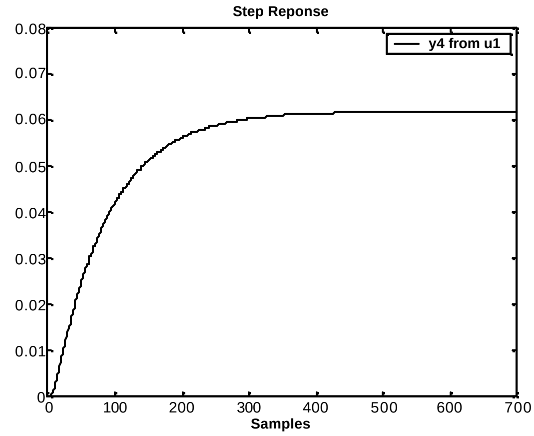

Figure 2.26: Unit step response of Output-4 from Input-1

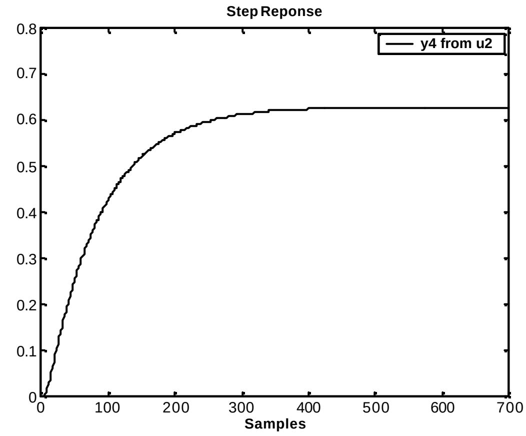

Figure 2.27: Unit step response of Output-4 from Input-2

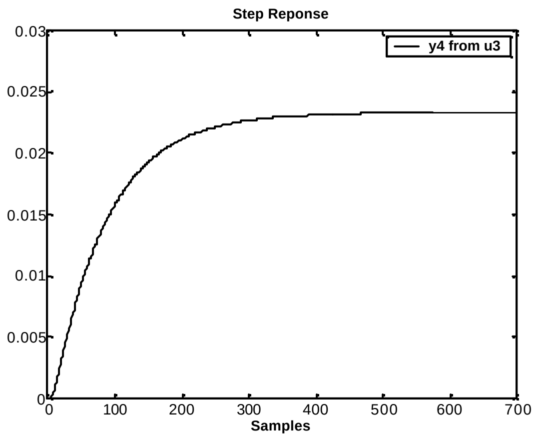

Figure 2.28: Unit step response of Output-4 from Input-3

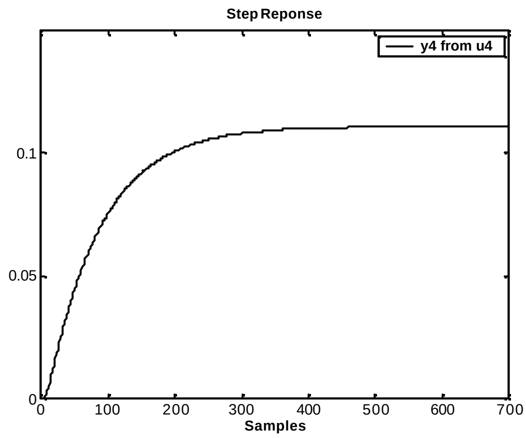

Figure 2.29: Unit step response of Output-4 from Input-4

2.5 Design of MPC

In this section, MIMO MPC is designed for the Demethanizer column. The following criterion function is minimized

$$J = \sum_{i=1}^{H_P} \left\| \hat{y}(k+i) - w(k+i) \right\|_{Q(i)}^2 + \sum_{i=1}^{H_C} \left\| \Delta u(k+i-1) \right\|_{R(i)}^2 \qquad (2.12)$$

In chapter 2 this cost function has already been discussed. In the first step, unconstrained MPC is designed. This is done to ensure that the designed scheme is stable and robust. The following subsections detail the steps followed in this regard. The most important thing in MPC design is the selection of the tuning parameters as discussed in section 2.2.2.

2.5.1 Set Points

The set points are chosen as $-0.2 \le y_1 \le 0.2$ for bottoms C_1/C_2, $-0.09 \le y_2 \le 0.09$ for LP residue gas valve opening, $-0.09 \le y_3 \le 0.09$ for Demethanizer pressure differential and $-0.13 \le y_3 \le 0.13$ for tray 6 temperature.

2.5.2 Prediction and Control Horizons

The prediction and control horizons are tuning parameters for the Demethanizer column process running with MPC. The selected prediction and control horizons are $H_p = 5$ and $H_c = 4$ respectively. Selection of these parameters for the horizons generates a very good and robust performing MPC scheme in which the controller meets all the requirements (set points) as far as the tracking performance is concerned. It is found that varying the values of H_c results in poor performance of the controller. Similarly increasing H_p results in reduce tracking performance of the controller.

2.5.3 Selection of weighting matrices Q and R

The weighting matrices Q and R in Eq. 2.2 are chosen as

$$Q = I_4 \qquad \text{and} \qquad R = O_4$$

where

$$I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad O_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

This means that the weight on the tracking error is 1 resulting in equally weighted tracking error over the prediction horizon. Selection of R to be zero means that there is no weight on the control moves and the process does not rquire certain components of the controller output to be enhanced or attenuated.

Figs. 2.30 - 2.33 show the unconstrained tracking performance of the process. MPC performs very well and the process is stable. Optimal inputs generated by the designed MPC for the Demethanizer column process are shown in Figs. 2.34 - 2.37. Large peaks are observed at these inputs which show that the controller reacts strongly to the set point changes. Here no (amplitude) constraints have been added on the control moves. The resulting controller is stable and meets the requirement of the process (set points). However, in the next section this process, running with MPC, will be subject to different process constraints and the effect of measured disturbances will be taken into account.

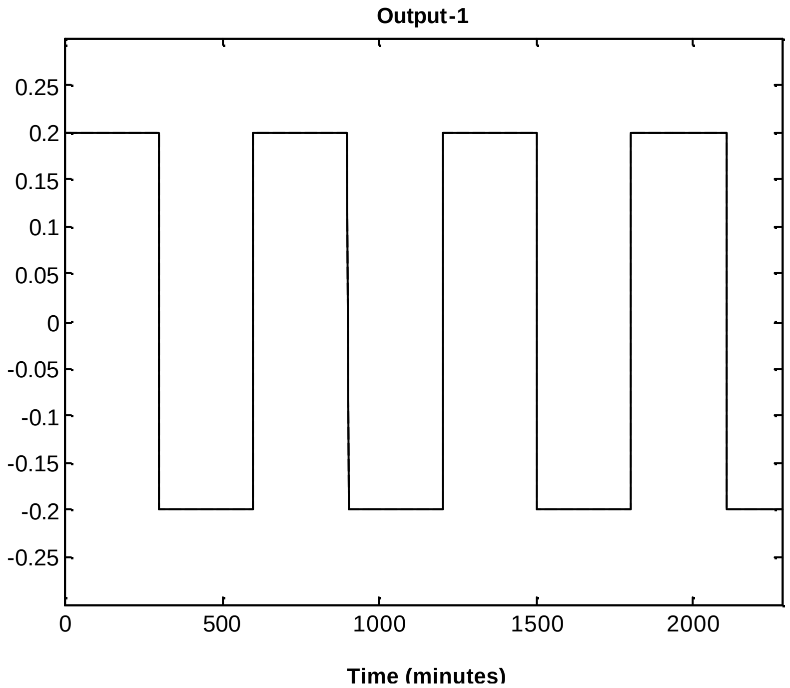

Figure 2.30: Tracking response of MPC to set point change (Output-1)

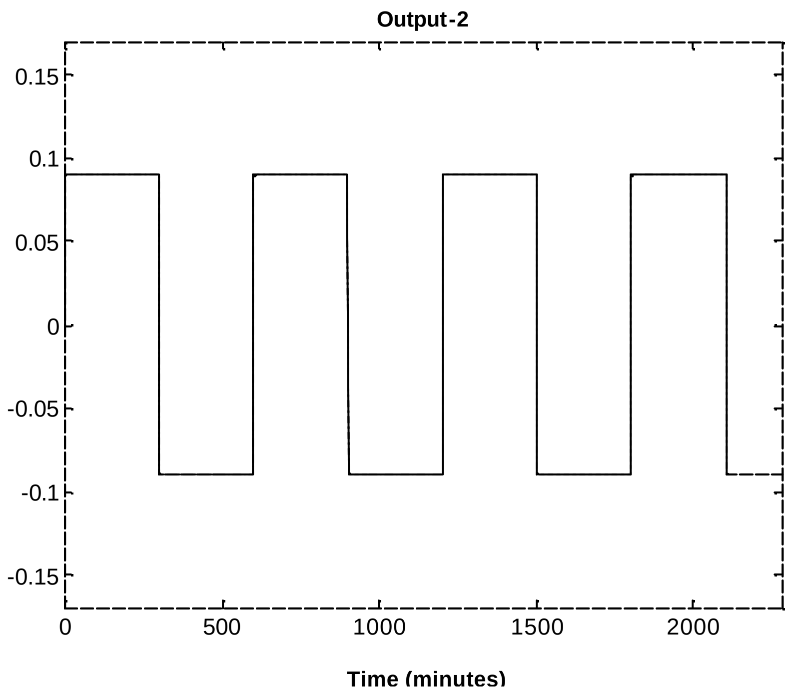

Figure 2.31: Tracking response of MPC to set point change (Output-2)

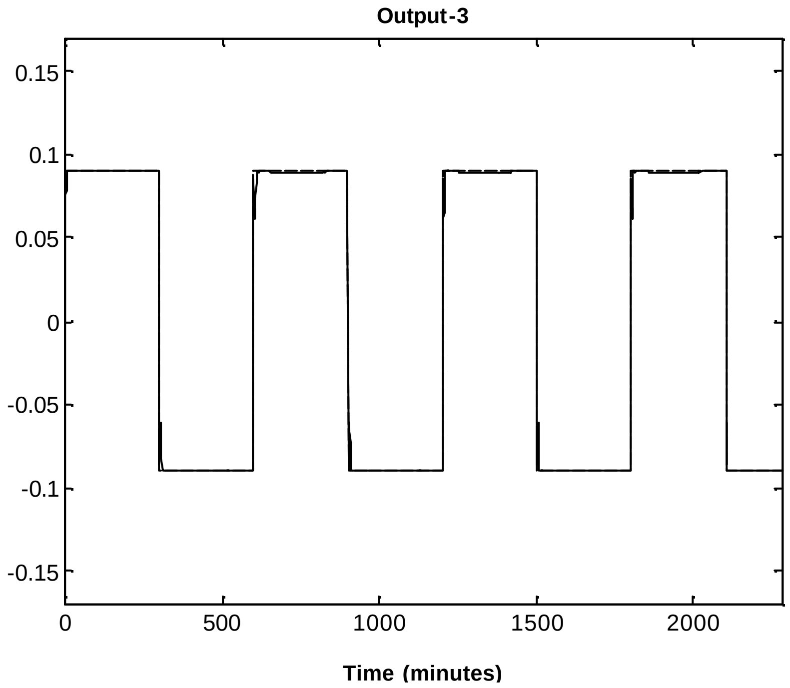

Figure 2.32: Tracking response of MPC to set point change (Output-3)

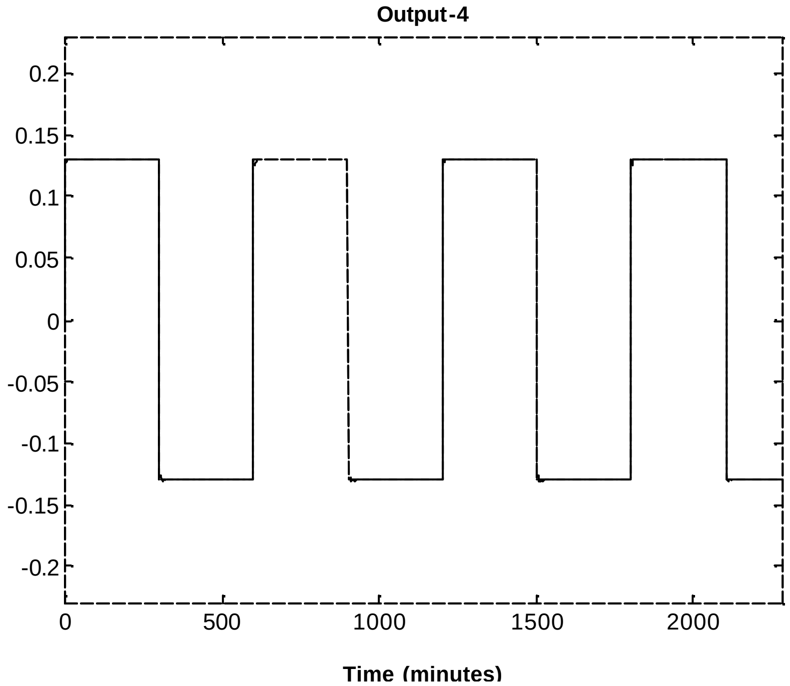

Figure 2.33: Tracking response of MPC to set point change (Output-4)

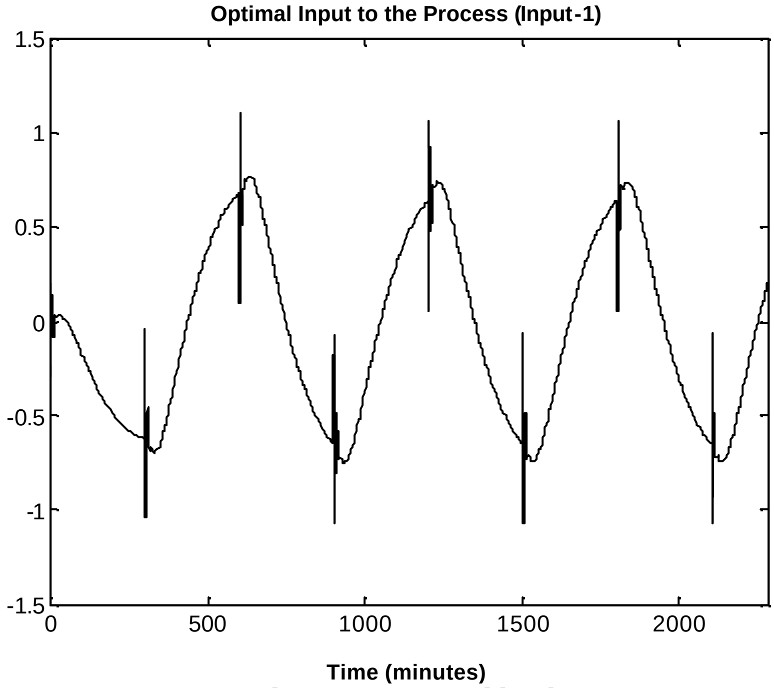

Figure 2.34: Optimal inputs generated by the MPC (Input-1)

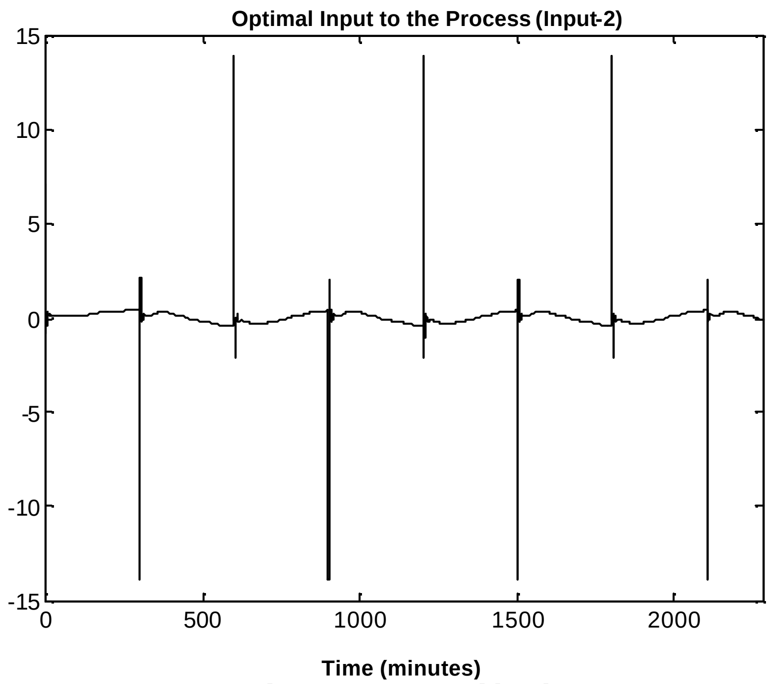

Figure 2.35: Optimal inputs generated by the MPC (Input-2)

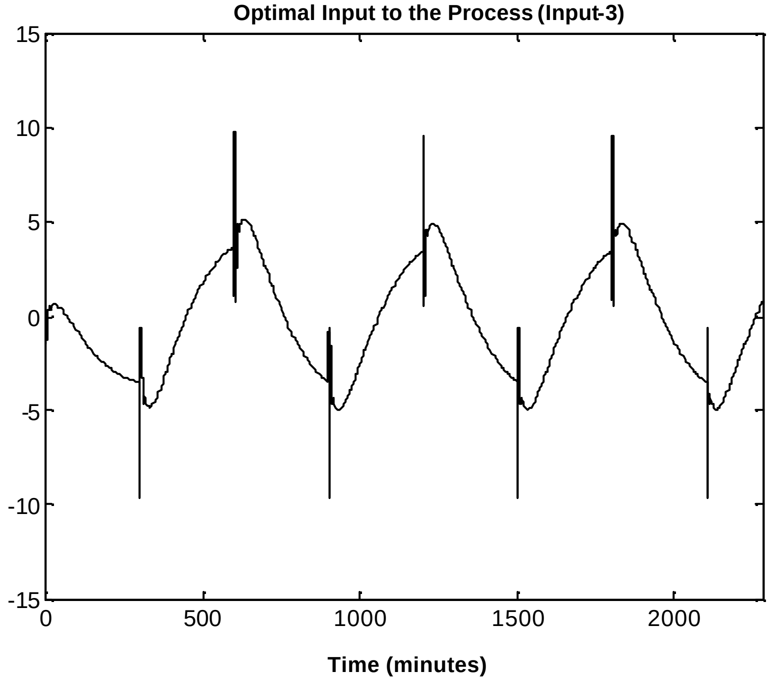

Figure 2.36: Optimal inputs generated by the MPC (Input-3)

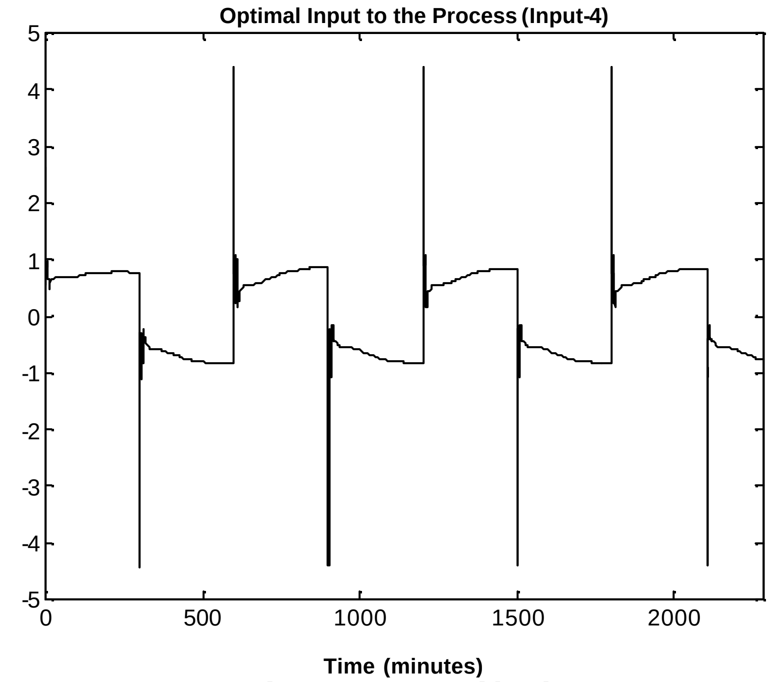

Figure 2.37: Optimal inputs generated by the MPC (Input-4)

2.5.4 Case 1 – With Amplitude Constraints, Noise and Measured Disturbances

The designed MPC is now subjected to input amplitude constraints, noise and measured disturbances. Disturbances cannot be ignored in practice. In order to take the disturbances into account, they have to be modeled first. This is already done in Eqs. 2.8 - 2.11. These disturbances are often referred to as deterministic disturbances. In addition to deterministic disturbances there are also stochastic disturbances present in a process. These are discrete white noise sequence with zero mean and a certain standard deviation$s$.

MPC is penalized with the following constraints on the inputs.

$$-0.7617 \leq u_1 \leq 1 \tag{2.13}$$

$$-0.5 \leq u_2 \leq 0.5411 \tag{2.14}$$

$$-1.625 \leq u_3 \leq 1.238 \tag{2.15}$$

$$-0.9 \leq u_4 \leq 0.8 \tag{2.16}$$

These constraints are a requirement of the process itself. The controller inputs are not required to exceed these values. With the addition of these constraints the predictive controller must anticipate violations and correct for them in a systematic way such that no violations are allowed while keeping the operation closed to these constraints.

The standard deviations of the two measured disturbances (d_1 and d_2) are selected as 0.14. This is a high value considering the process dynamics, the signal to noise ratio. The standard deviation of the white noise sequence is selected as 0.01 beyond which the noise is too high for the process. The responses of the MPC process are shown in Figs. 2.38 – 2.41. The optimal inputs are shown in Figs. 2.42 – 2.45 where the constraints are shown by dotted lines. It can be seen that no violation of the bounds is allowed at any time. The controller takes the constraints into account and allows the process to operate closed to these physical limitations. In addition to this, it is observed that the high value of disturbance causes perturbations in all the outputs, especially in the case of outputs 2 and 4. The effect of noise is also visible as it is forcing the responses to deviate slightly from the set points. However, the performance of MPC under constraints is acceptable for the purpose of this thesis. This closed loop data for the four inputs and four outputs process is collected as 'case 1' and will be used later in Chapter 3 where closed loop identification is performed.

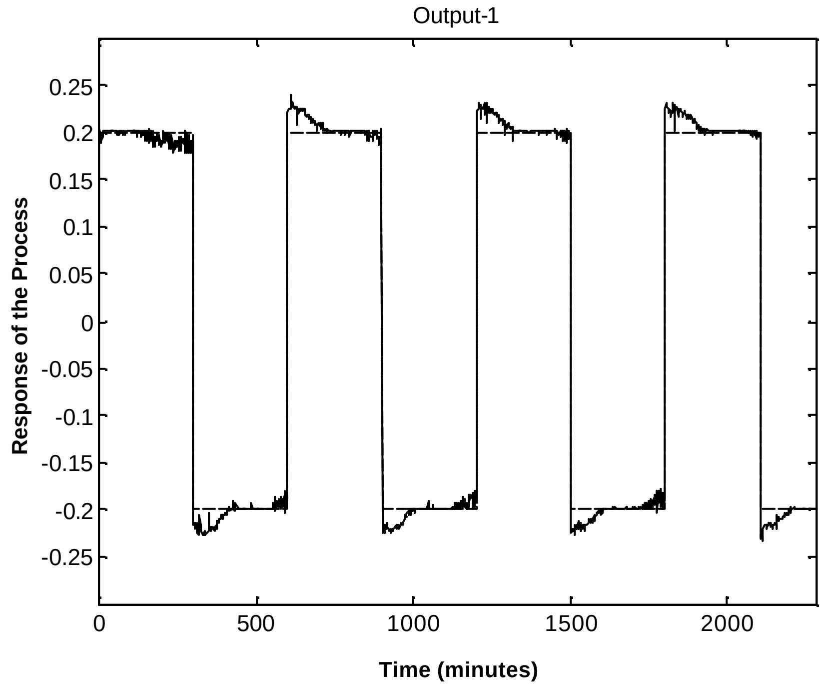

Figure 2.38: Case 1 - Response of MPC (Output-1)

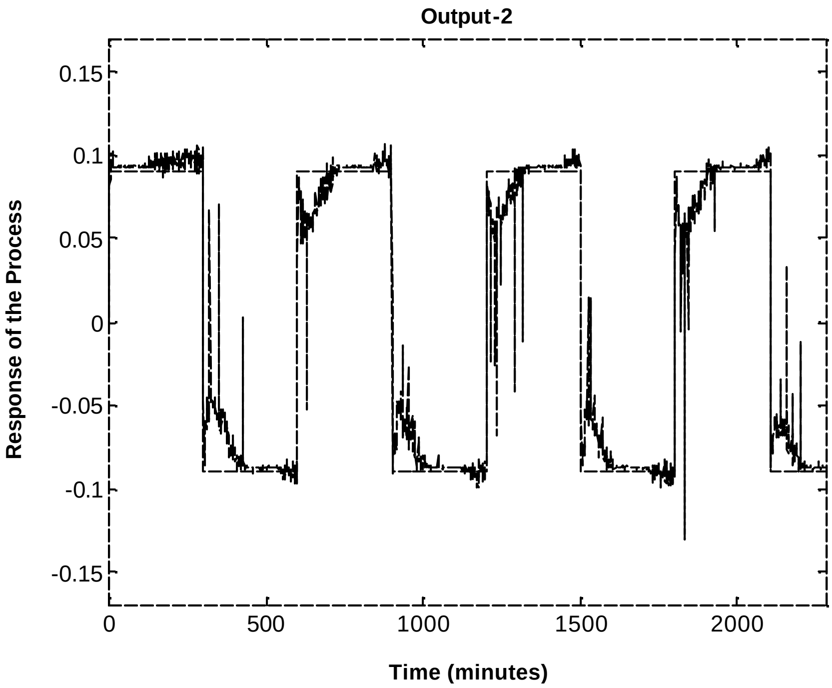

Figure 2.39: Case 1 - Response of MPC (Output-2)

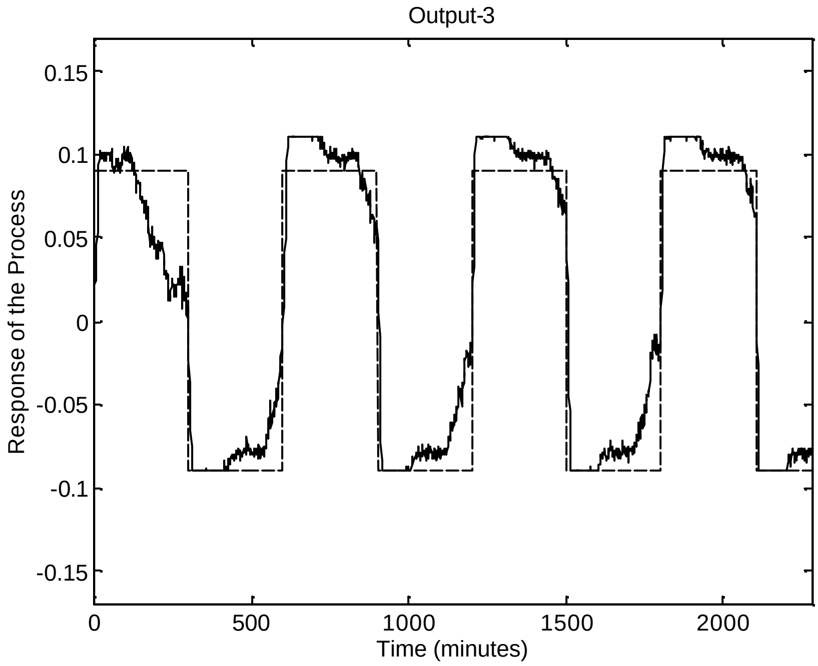

Figure 2.40: Case 1 - Response of MPC (Output-3)

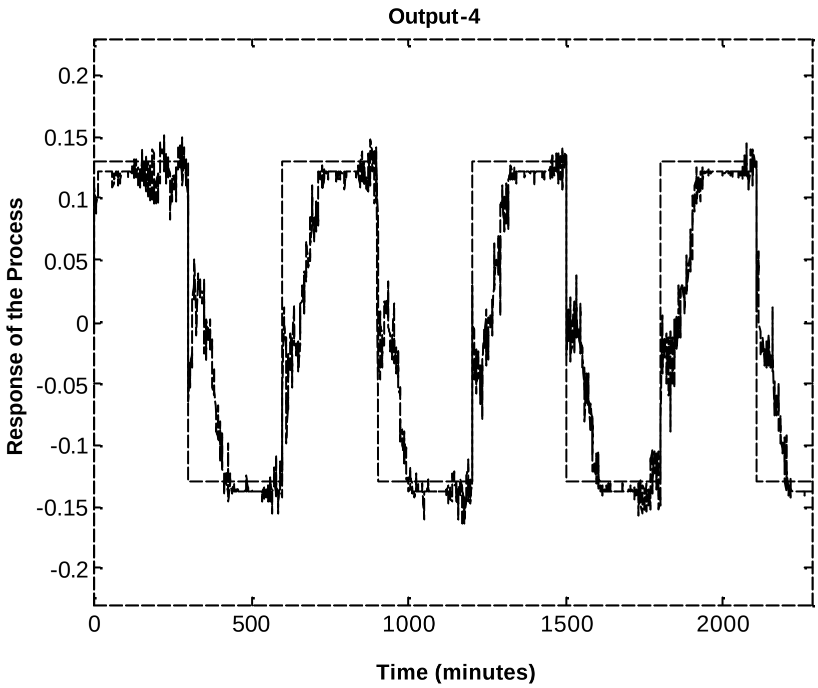

Figure 2.41: Case 1 - Response of MPC (Output-4)

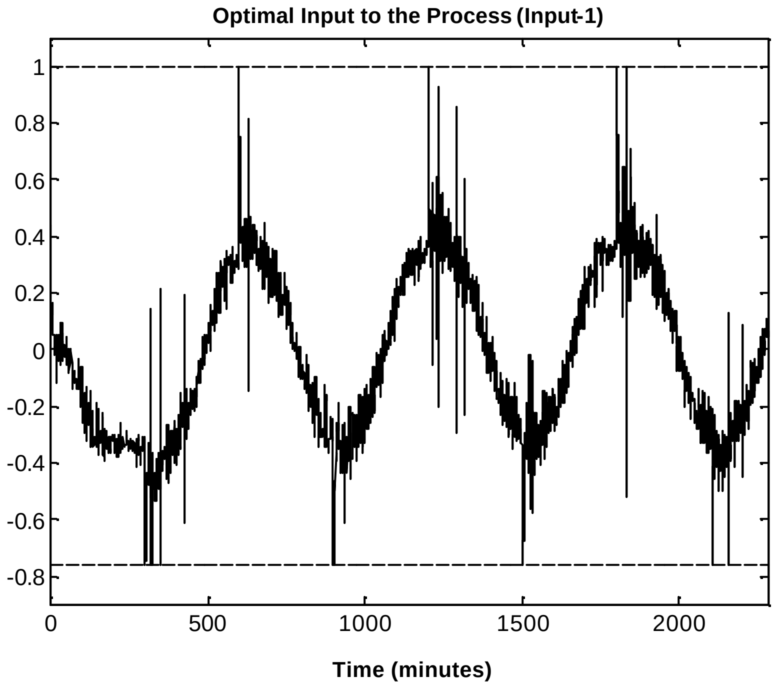

Figure 2.42: Case 1 - Optimal inputs generated by the MPC (Input-1)

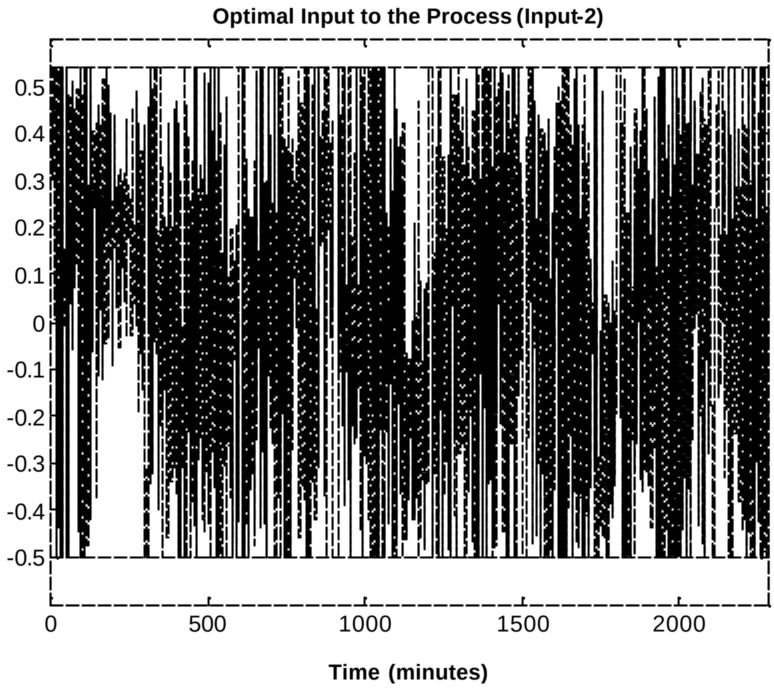

Figure 2.43: Case 1 - Optimal inputs generated by the MPC (Input-2)

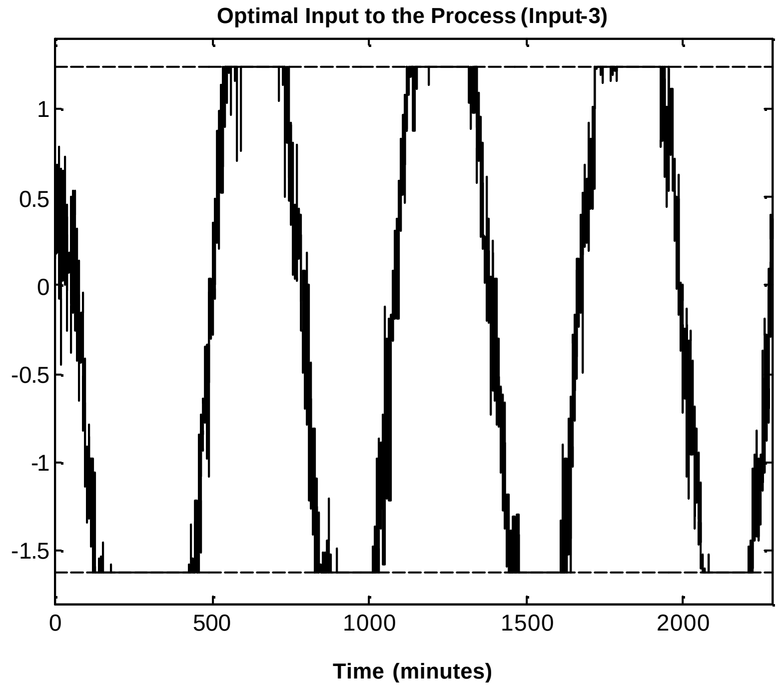

Figure 2.44: Case 1 - Optimal inputs generated by the MPC (Input-3)

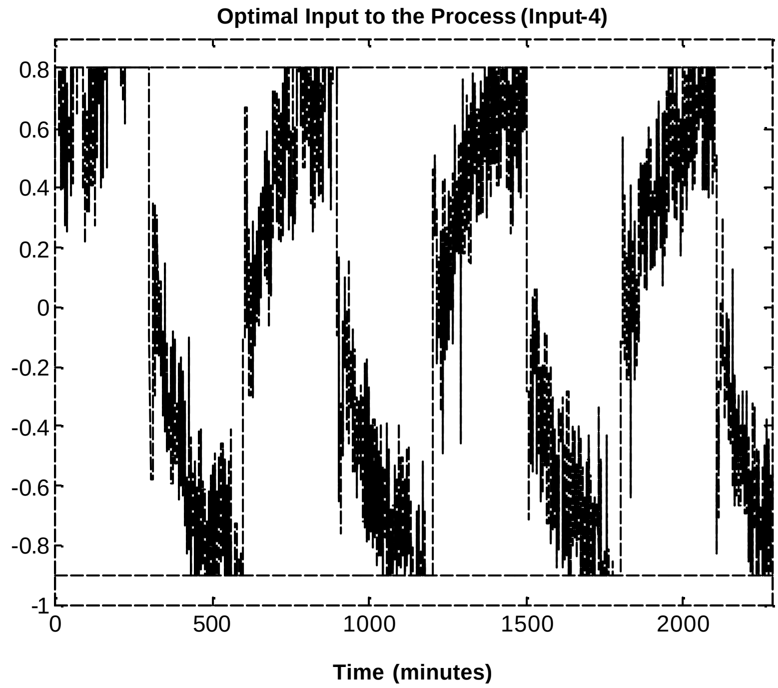

Figure 2.45: Case 1 - Optimal inputs generated by the MPC (Input-4)

2.5.5 Case 2 – With Amplitude and Rate Constraints, Noise and Measured Disturbances

In this subsection, rate constraints are added to the designed MPC. Recall, that rate constraints refer to the limitation of MPC output per sample between two values. In other words, large changes in input moves are avoided to limit large changes at the output of the process. The following rate constraints are added to the designed process.

$$-0.6 \leq \Delta u_1 \leq +0.6 \tag{2.17}$$

$$-0.6 \leq \Delta u_2 \leq +0.6 \tag{2.18}$$

$$-0.6 \leq \Delta u_2 \leq +0.6 \tag{2.19}$$

$$-0.6 \leq \Delta u_2 \leq +0.6 \tag{2.20}$$

It has already been discussed that in the presence of amplitude constraints, any violation may lead to a performance degradation of the system. In a similar manner, the presence of rate constraints may lead to an unstable system [61]. It has been observed that the MPC works very well under amplitude constraints. Now, the effect of rate constraint is ascertained. In these simulations, the high value of the two disturbances is reduced a little.

In the previous subsection, the disturbance values were kept very high ($s = 0.14$) depending on the signal to noise ratio. In this design, the standard deviations of the two disturbances d_1 and d_2 are reduced to 0.08. The four outputs of the process are shown in Figs. 2.46 – 2.49. The outputs show improved tracking behavior with reduced disturbances. The deviation of output 2 from the set point is reduced. The affect of rate constraints is evident from the optimal inputs shown in Figs. 2.50 – 2.53. The noise level has not been changed in this design.

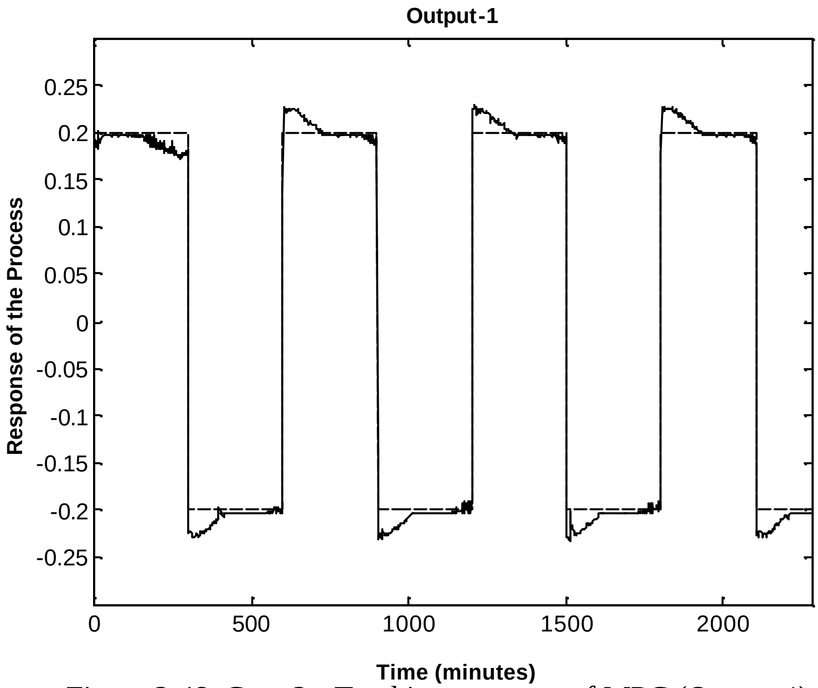

Figure 2.46: Case 2 - Tracking response of MPC (Output-1)

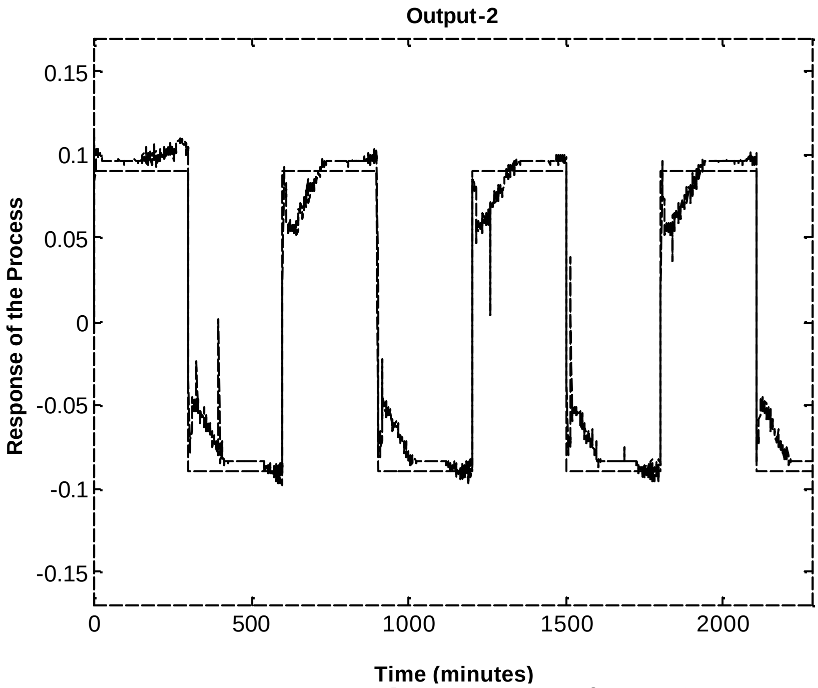

Figure 2.47: Case 2 - Tracking response of MPC (Output-2)

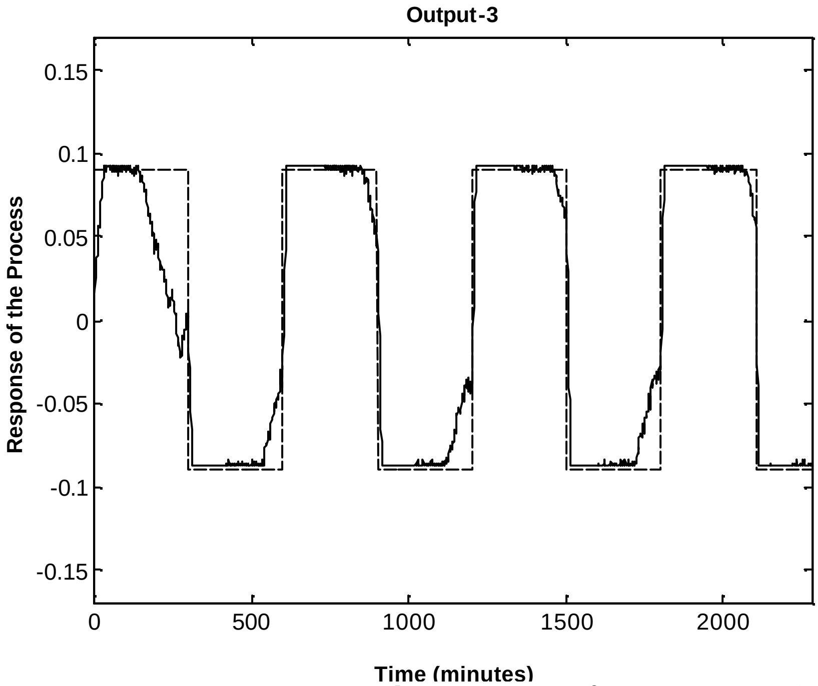

Figure 2.48: Case 2 - Tracking response of MPC (Output-3)

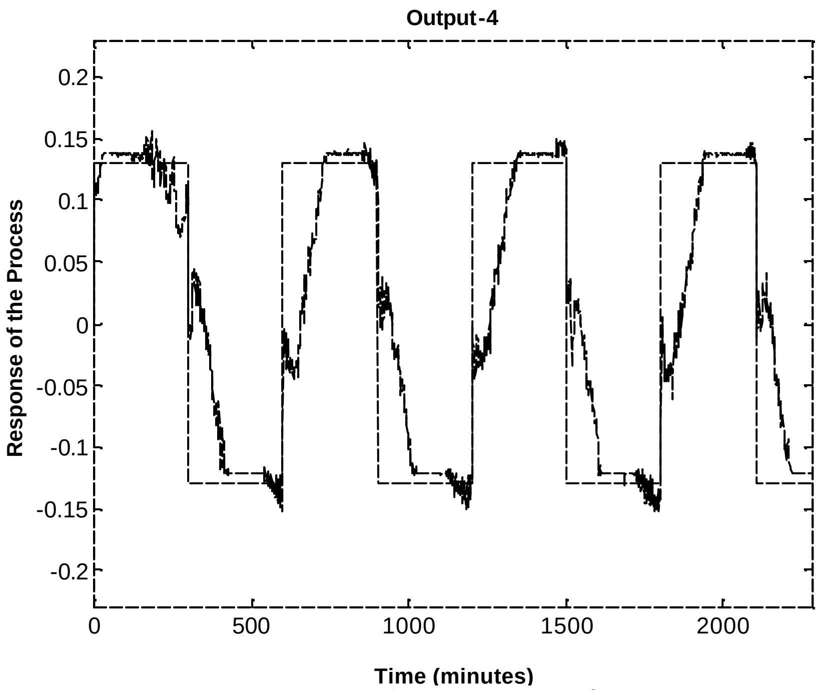

Figure 2.49: Case 2 - Tracking response of MPC (Output-4)

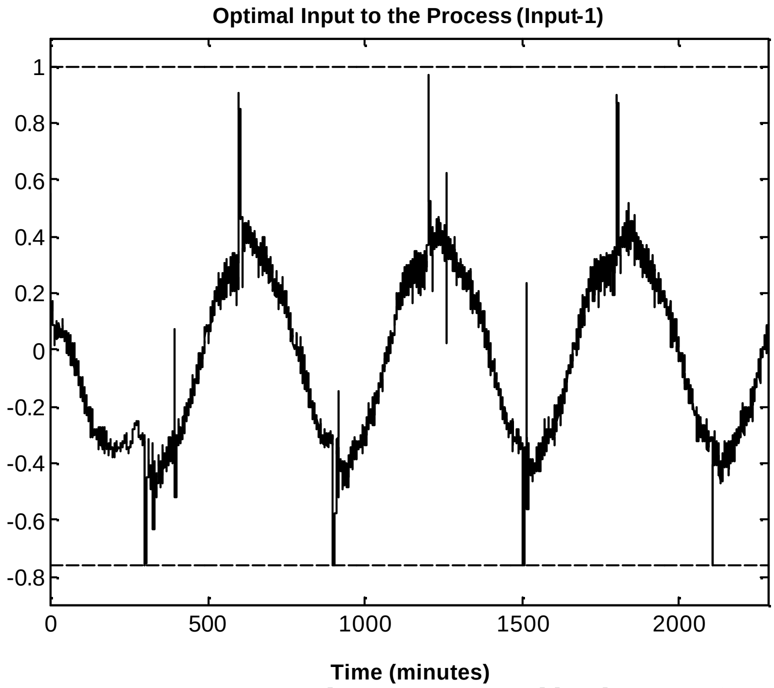

Figure 2.50: Case 2 - Optimal inputs generated by the MPC (input-1)

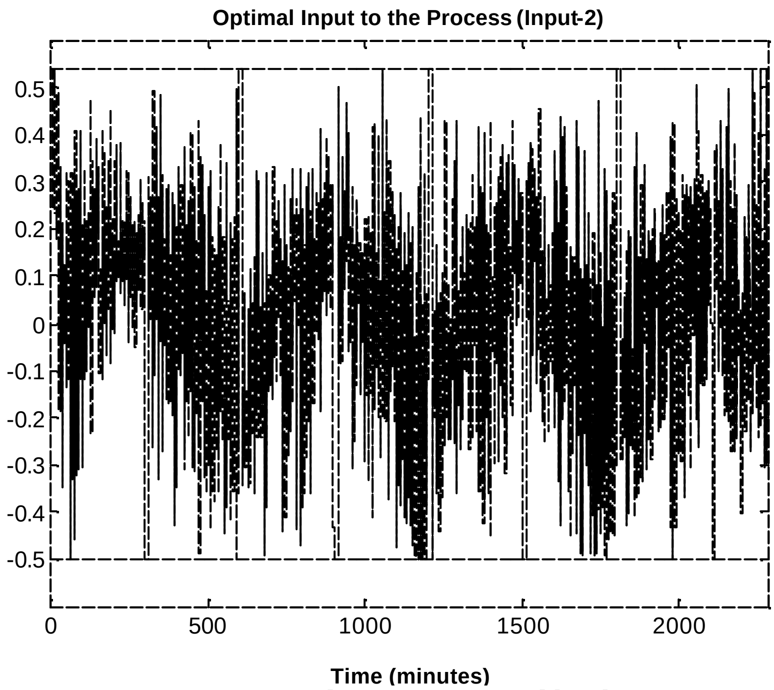

Figure 2.51: Case 2 - Optimal inputs generated by the MPC (input-2)

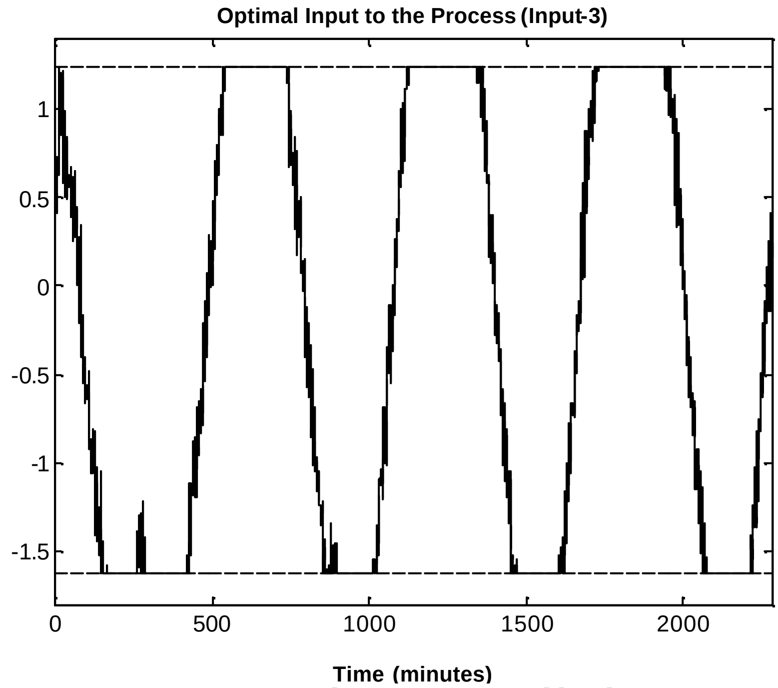

Figure 2.52: Case 2 - Optimal inputs generated by the MPC (input-3)

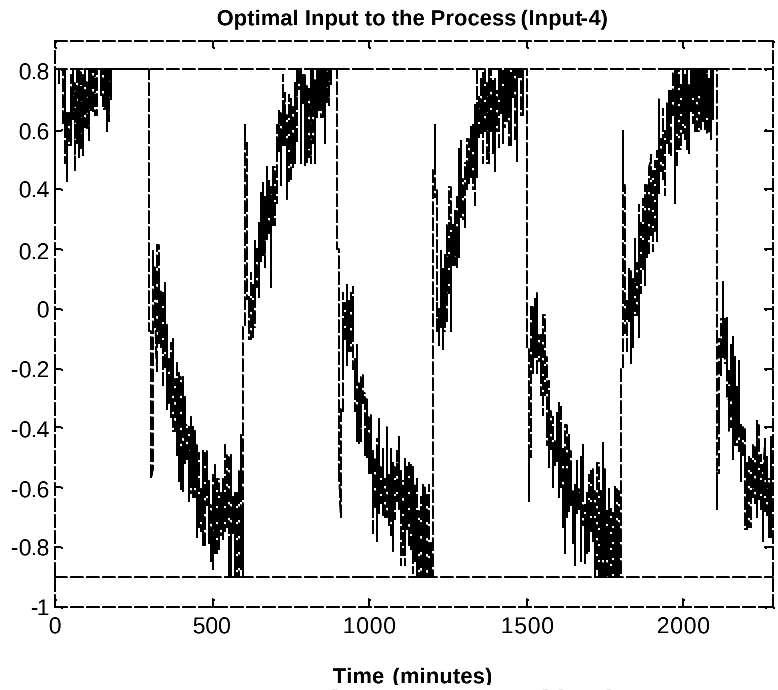

Figure 2.53: Case 2 - Optimal inputs generated by the MPC (input-4)

2.5.6 Case 3 – With Amplitude Constraints, Noise, Measured Disturbances and Plant Model Mismatch

According to Gracia and Morari [66], an important property of model predictive controllers is that no stability problems exist under perfect model conditions, even in the face of constraints on the manipulated variables. However, if the model is not the same as the plant, in particular if the steady state gain of the model is incorrect, then the plant output will reach an incorrect final value. This means that in the face of significant model inaccuracies, the control system generally is unable to satisfy all of the true performance criteria specified for the process

To observe this phenomenon, MPC designed earlier using the notion that an exact plant-model is available, is now subjected to this reality of plant-model mismatch. All the parameters of the original process model in Eqs. 2.9 – 2.11 are altered to a percentage of 6. This change is large considering the highly sensitive nature of the process A higher value than this causes the performance of MPC to degrade drastically and the process becomes unstable.

The amplitude constraints remain the same but the input rate constraints are not applied for this case. The disturbances are further reduced in this case, depending on the signal to nose ratio. The standard deviations of the two measured disturbances are selected as 0.03. The standard deviation of the noise is kept the same to 0.01. The response of the designed MPC is depicted in Figs. 2.54 - 2.57. The optimal inputs are shown in Figs. 2.58 – 2.61. The inputs are not allowed to violate the constraints imposed on them. The results show degradation in the performance of MPC and high oscillations are observed at the outputs. MPC is no longer able to exhibit robust tracking ability and is slow in meeting the set points changes. This is in fact the main motivation of this thesis. The plant model mismatch is the main problem in any MPC design scheme. Practical solution is to shut down the controller and model the actual open loop plant again. For the case of closed loop identification it is however, a unique opportunity to see if the actual open loop process model can be identified by using the closed loop data from such a worse case scenario. The input and output data from this plant model mismatch MPC scheme is collected as ‘case 3’ and will be used in the analysis of closed loop identification in Chapter 3.

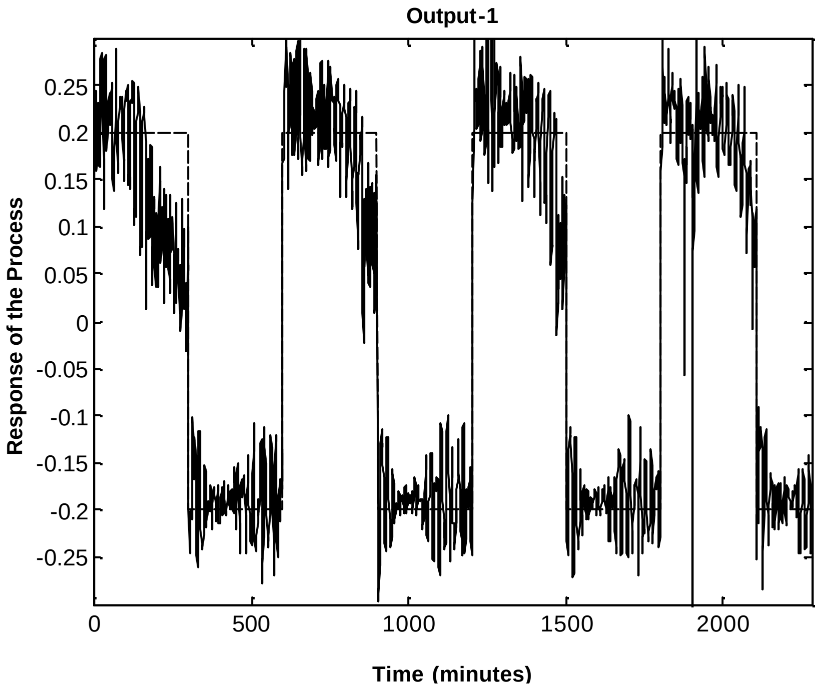

Figure 2.54: Case 3 - Tracking response of MPC (Output-1)

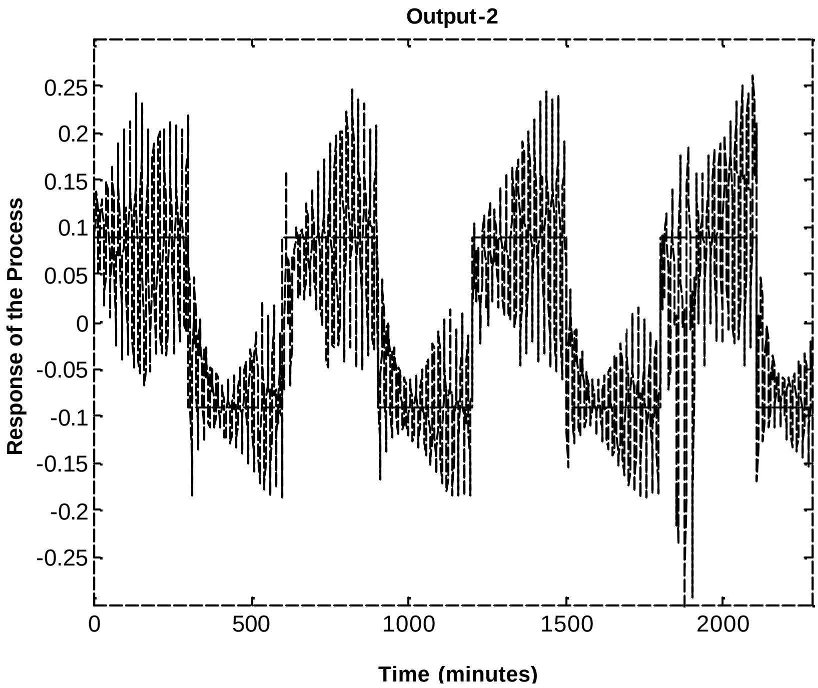

Figure 2.55: Case 3 - Tracking response of MPC (Output-2)

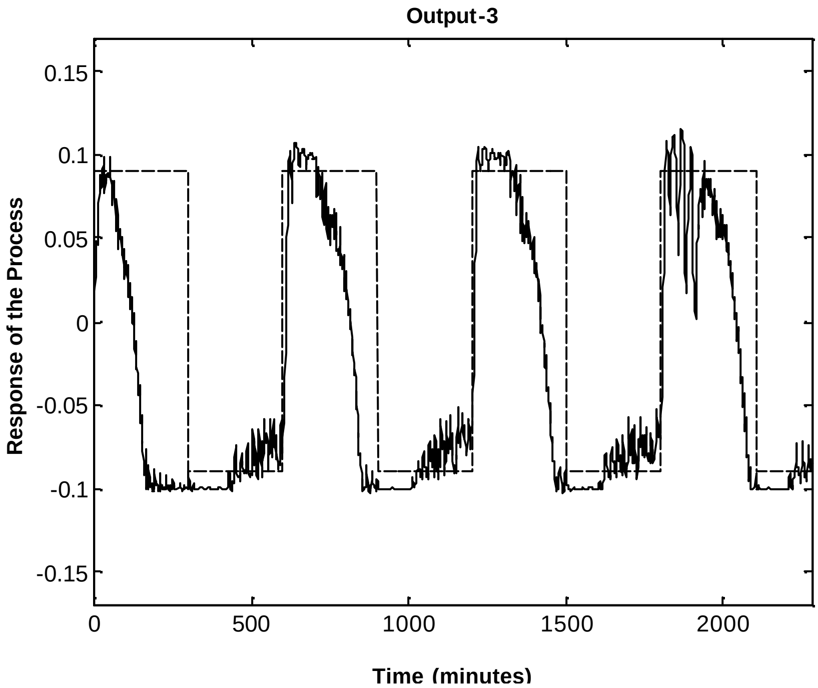

Figure 2.56: Case 3 - Tracking response of MPC (Output-3)

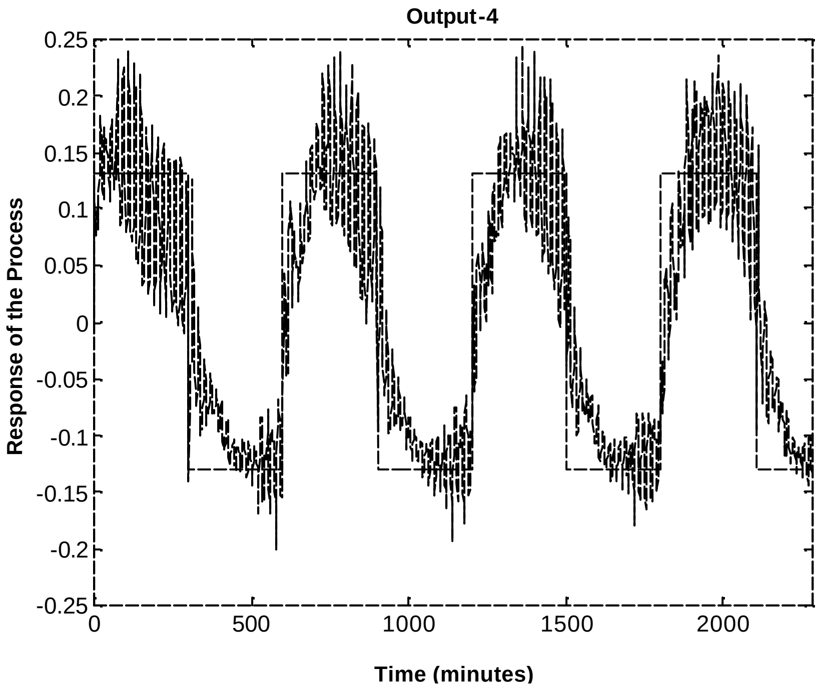

Figure 2.57: Case 3 - Tracking response of MPC (Output-4)

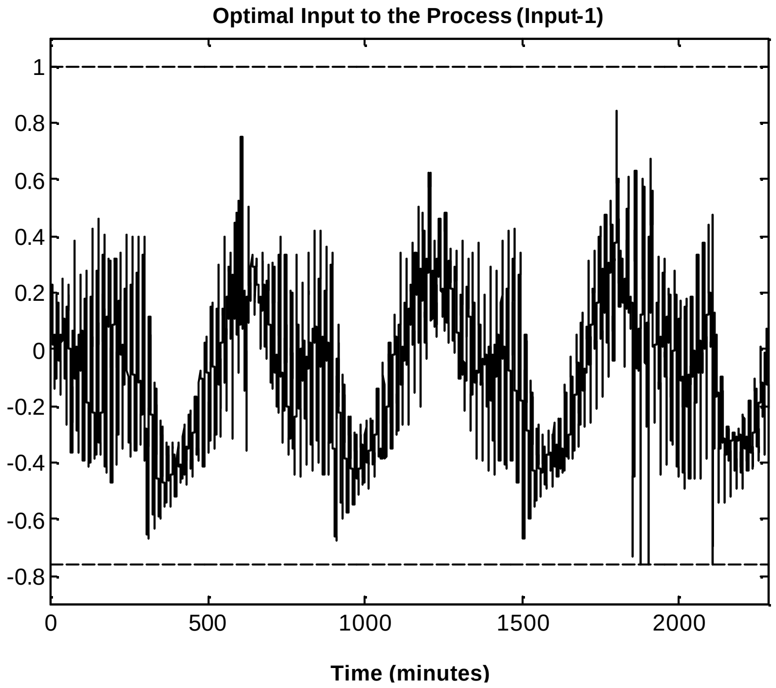

Figure 2.58: Case 3 - Optimal inputs generated by the MPC (Input-1)

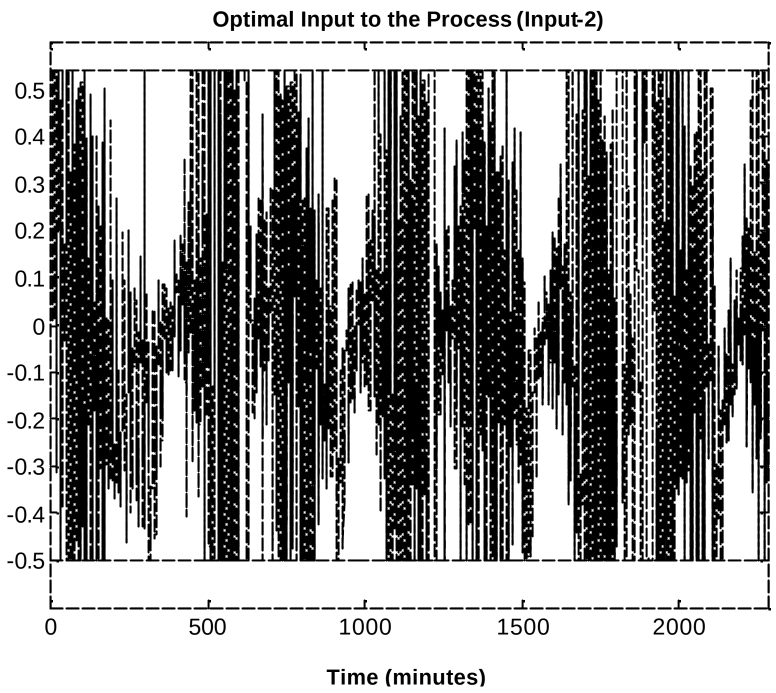

Figure 2.59: Case 3 - Optimal inputs generated by the MPC (Input-2)

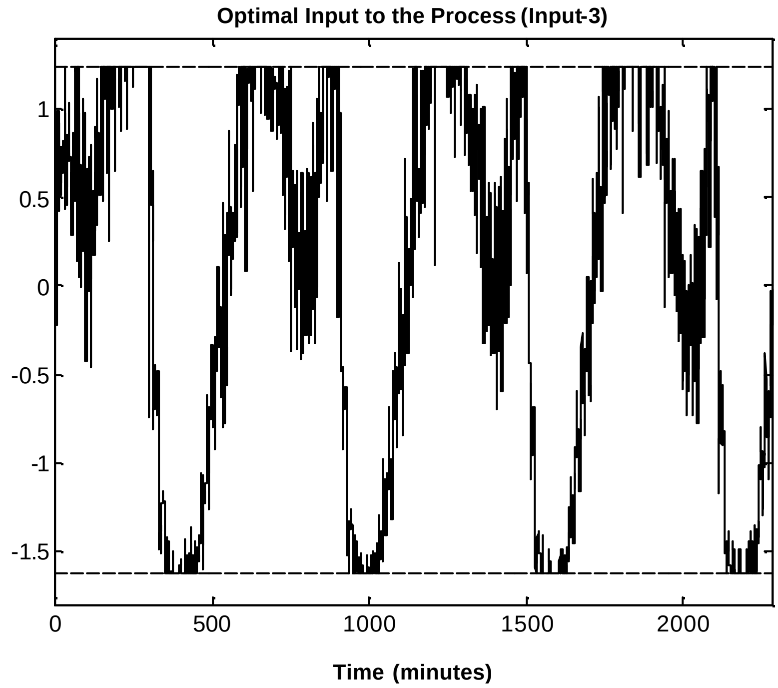

Figure 2.60: Case 3 - Optimal inputs generated by the MPC (Input-3)

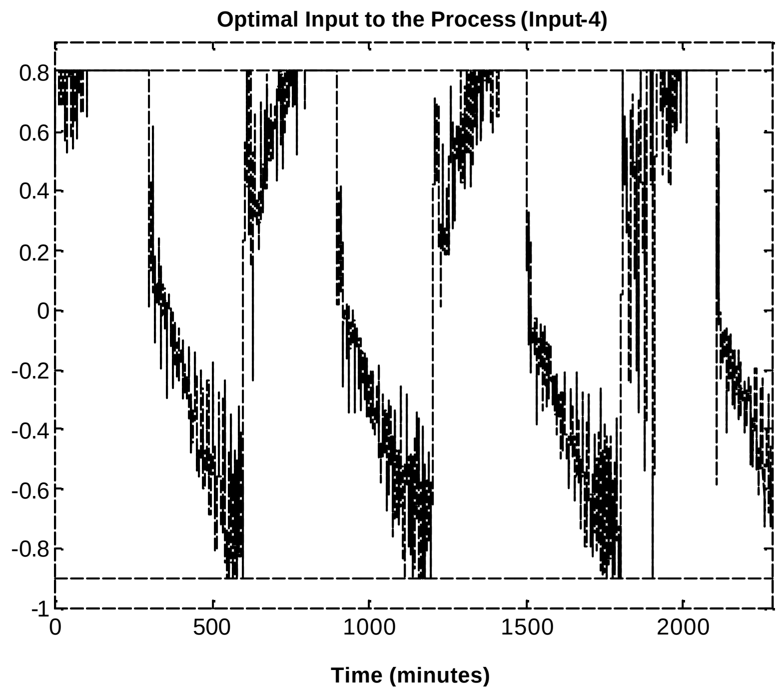

Figure 2.61: Case 3 - Optimal inputs generated by the MPC (Input-4)

2.6 Chapter Summary

In this chapter, subspace identification method is used to identify a state space process model for the Demethanizer column from open loop data collected from the actual process. It is found that the state space model gives accurate results and shows a very good performance when compared with the actual data. The errors remain mostly between ± 0.4.

The identified state space process model is then used to design MPC for three cases. In the first design, amplitude constraints, disturbances and noise are taken into account. For the second design, rate constraints are also added to the process; and in the last design, plant-model mismatch is taken into account. From all these simulations closed loop data, which is from here on referred to as 'simulated MPC closed loop data', is collected.

The goal of the next chapter is to study the feasibility of using closed loop data for identifying the open loop process model. This will offer a number of practical advantages such as better models, validation, controller maintenance and most of all no need for open loop identification which involves MPC controller shutdown. In addition to this, open loop identification schemes will be brought forward that give good modeling results from the simulated closed loop MPC data obtained from simulations in this chapter.

Chapter 3

Closed Loop Identification - MPC

3.1 Introduction

As mentioned in chapter 1, closed loop experiments are natural when the intended model use is control design. The three main categories of all closed loop identification methods are *direct approach, indirect approach* and *joint input-output approach*. As per Ljung and Forsell [22], the direct approach gives consistency and optimal accuracy, and therefore, the direct approach should be regarded as the first choice of methods for closed loop identification.

In the indirect approach, the main focus is on correct modeling of the closed loop system and consistency can be obtained even for incorrect noise models. This approach is more complex than the direct method. For MPC, this method is redundant and cannot be used as it requires complete knowledge of the controller as a structure. Unlike PID and other controller design techniques, MPC is an algorithm and is not expressed in terms of some linear relationship.

The joint input-output approach is an alternate approach to both direct and indirect approaches. In this approach, no explicit knowledge of the controller is required except that it must be known or assumed to be of a certain (linear) structure. This assumption again is not possible in the case of MPC.

In the following sections, direct identification approach is used and a benchmark of the performance of different parametric identification techniques is devised. In summary, four different model structures are used in all identification methods. They are:

- ARX
- ARMAX
- OE
- State space

3.2 Model Structures and Estimation Methods

In this section, some notations and model structures used in this thesis are introduced. Given a multivariable process with m manipulated variables (or inputs) and p controlled variables (or outputs) the data sequence collected from an identification test is

$$Z^N = \{u(1), u(2), u(3), \ldots\ldots u(N), y(1), y(2), y(3), \ldots\ldots y(N)\} \qquad (3.1)$$

where $u(t)$ is m-dimensional input vector (MVs), $y(t)$ is p-dimensional output vector (CVs) and N is the number of samples or data points. It is assumed that the data is generated by the following linear process:

$$y(t) = G(q^{-1})\, u(t) + H(q^{-1})\, e(t) \qquad (3.2)$$

Here q^{-1} is the unit time delay operator, $G(q^{-1})$ is the process transfer function and $H(q^{-1})$ is the noise model and e(t) is a p-dimensional white noise vector. The model to be identified is the same structure as in Eq. 3.2.

$$y(t) = \hat{G}(q^{-1})\, u(t) + \hat{H}(q^{-1})\, e(t) \tag{3.3}$$

Depending on how to parameterize the model in Eq. 3.3, different parameter estimation methods studied in literature can be derived.

3.2.1 ARX (AutoRegressive with eXternal input) Model

The ARX model structure corresponds to the choice

$$G(q^{-1}) = q^{-d} \frac{B(q^{-1})}{A(q^{-1})} \text{ and } H(q^{-1}) = \frac{1}{A(q^{-1})}$$

where, $A(q^{-1}) = 1 + \sum_{k=1}^{n_a} a_k q^{-k}$ and $B(q^{-1}) = \sum_{\boldsymbol{k=0}}^{n_b} b_k q^{-k}$ are polynomial matrices. d is the delay of the system. This model can be expressed as

$$A(q^{-1})\, y(t) = q^{-d}\, B(q^{-1})\, u(t) + e(t) \tag{3.4}$$

The coefficient of polynomials A and B are estimated by minimizing the sum of the squared equation error *e(t)* defined as the difference between the actual and estimated outputs.

$$e(t) = y(t) - \hat{y}(t\,|\,q) = y(t) - j^{\,T}(t)\, q \tag{3.5}$$

where q is the parameter vector and j is the regression vector, which contains all the past inputs and past outputs. They are defined as

$$\begin{aligned} j\,(t) &= [\,y(t-1) \cdots y(t-n_a)\;\; u(t-1) \cdots u(t-d-n_b)\,]^T \\ q &= [-a_1 \cdots - a_{n_a}\;\; b_0 \cdots b_{n_b}\,]^T \end{aligned} \tag{3.6}$$

This estimation method is called least squares and is explained in Appendix A (A.1).

3.2.2 ARMAX (AutoRegressive Moving Average with eXternal input) Model

This model structure has a more general structure than the ARX:

$$G(q^{-1}) = q^{-d}\,\frac{B(q^{-1})}{A(q^{-1})} \text{ and } H(q^{-1}) = \frac{C(q^{-1})}{A(q^{-1})}$$

Here $C(q^{-1}) = 1 + \sum_{k=1}^{n_c} c_k q^{-k}$ is a polynomial, the presence of which means that noise term is explicitly modeled. This model can be expressed as

$$A(q^{-1})y(t) = q^{-d}\,B(q^{-1})u(t) + C(q^{-1})e(t) \tag{3.7}$$

The coefficients of the polynomials A, B and C are estimated by minimizing the sum of the squared prediction error $e(t)$ which is defined as in Eq. 3.8.

$$e(t) = y(t) - \hat{y}(t\,|\,q) = y(t) - j^{T}(t,q)q \tag{3.8}$$

where q is the parameter vector and j is the regression vector defined as

$$\begin{aligned} j(t) &= [\,y(t-1)\cdots y(t-n_a)\;\; u(t-1)\cdots u(t-d-n_b)\;\; e(t,q)\cdots e(t-n_c,q)\,]^T \\ q &= [-a_1\cdots -a_{n_a}\;\; b_0\cdots b_{n_b}\;\; c_1\cdots c_{n_c}\,]^T \end{aligned} \tag{3.9}$$

The regression vector depends on the model parameters and is no longer linear as in the case of ARX. This makes the estimation of model parameters more complicated. For this case, the parameter estimation technique is referred to as Prediction Error Method (PEM) and is explained in Appendix A (A.2) of this thesis.

3.2.3 OE (Output Error) Model

The output error (or parallel) model structure is used in the case when the process output is disturbed by only white measurement noise. It corresponds to the following choice:

$$G(q^{-1}) = q^{-d}\,\frac{B(q^{-1})}{F(q^{-1})} \text{ and } H(q^{-1}) = 1$$

This model is expressed as

$$y(t) = q^{-d}\,\frac{B(q^{-1})}{F(q^{-1})}u(t) + e(t) \tag{3.10}$$

where $F(q^{-1}) = 1 + \sum_{k=1}^{n_f} F_k q^{-k}$ is a polynomial matrix. As in the case of ARMAX models, the polynomial F and B are estimated by using Prediction error Method. The regression vector j and parameter vector q are defined as

$$\begin{aligned} j\,(t) &= [\hat{y}(t-1)\cdots\hat{y}(t-n_f)\;\; u(t-1)\cdots u(t-d-n_f)\,]^T \\ q &= [-f_1\cdots-f_{n_f}\;\;\; b_0\cdots b_{n_b}]^T \end{aligned} \tag{3.11}$$

3.2.4 State Space model

State space model of a multivariable process is described by the following set of difference equations:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= \mathrm{C}x_k + Du_k + v_k \end{aligned} \tag{3.12}$$

where, w_k and v_k are zero mean, white noise sequences. The vectors $u_k \in \Re^m$ and

$y_k \in \Re^l$ are the measurements at time instant *k* of *m* inputs and *l* outputs of the process respectively. The vector x_k is the state vector of the process at discrete time instant *k*, $v_k \in \Re^{l\,x1}$ is called the measurement noise and $w_k \in \Re^{n\,x1}$ is called the process noise. The matrices *A*, *B*, *C* and *D* are estimated by using N4SID (numerical algorithm for subspace state space identification) method. A review of this method is given in Appendix B of this thesis.

In essence, a model of a dynamic system is a rule which makes it possible to construct some sort of an inference (relationship) based on observations of input-output data. ARX, ARMAX, OE and State space models are called parametric models and are shown to be more compact than nonparametric models such as FIR models [36].

3.3 Closed Loop Identification

In chapter 2, MPC was designed and simulated subject to various levels of external disturbances, noise and constraints. Closed loop input-output data was collected for three cases. It will be the objective of this section to bring forward identification schemes that will work with closed loop data. Direct Identification approach is used for closed loop identification. Recall that in the direct identification approach, the method is applied directly to measured input-output (*u*, *y*) data and no assumptions whatsoever are made on how the data was generated. Hence only the input and output data of the process need to be collected. This has already been done in Chapter 2. Using these data samples, closed loop models are identified and compared with the open loop state space process model used in the MPC design. The purpose is to observe which modeling technique, if any, will yield the best possible results by recovering the open loop process model from given closed loop data collected under feedback, which is contaminated with noise and has strong correlation between the input and process disturbances. Recommendations are made on the basis of these simulations.

3.3.1 Case 1 – Closed loop Identification

In the analysis of closed loop identification methods for use on MPC scheme, simulated input-output data is collected as in Figs. 2.38 - 2.41, which illustrate case 1 where the MPC is running under high disturbances d_1 and d_2 with standard deviation of 0.14. The standard deviation of noise is 0.01. The amplitude constraints are specified in Eqs. 2.18 - 2.21. The performance of ARX, ARMAX, OE and state space estimation and modeling schemes are compared and discussed respectively for this case in the following subsections.

3.3.1.1 Performance of ARX model

ARX modeling scheme is used to identify the open loop process model using the simulated closed loop data. Least squares method is utilized to estimate the unknown parameters. It is pertinent to mention here that modeling multivariable systems is often challenging. In particular, system with several outputs such as the Demathanizer column in this thesis is difficult to model [36]. A basic reason given in open loop identification is that the coupling between several inputs and outputs often lead to more complex models. Basically it is essentially just a matter of choosing the model order. It is even more difficult to model from closed loop data due to the presence of feedback. Recovering information about the original open loop process model from such a data (with strong correlation between the inputs and noise) is not simple.

The model orders (n_a) of the process outputs are selected as 2, 3, 2 and 2 respectively and the unknown model parameters are identified. For comparison, the step responses of the identified closed loop model and open loop model are plotted with each other in Figs. 3.1 - 3.16. The solid line indicates the current open loop model and the dashed line corresponds to the direct identified closed loop ARX model. There is a little or no mismatch for the first, second and fourth process outputs but the model shows a slight bias in process output three because of the high disturbance (nonlinearity) associated with this output. However, from a theoretical point of view it can be concluded that the current open loop model does not contain enough dynamics and there are higher order dynamics that need to be taken into account, which are represented in closed loop model. The results indicate that this simple conventional approach is very effective in identifying the open loop process model with MPC closed loop data. Good steady state gain fit is achieved to a large extent. The parameters of the identified model are given in Eqs. 3.13 - 3.16. It is clearly seen that the process and the noise/disturbance model have the same poles which is a characteristic of ARX modeling scheme.

$$
\begin{aligned}
y1(t) = {} & \frac{-0.0035 + 0.0026\,q^{-1}}{1 - 0.8498\,q^{-1} - 0.1395\,q^{-2}}\,q^{-2}\,u_1(t) \\
& + \frac{0.0063 - 0.0104\,q^{-1}}{1 - 0.8498\,q^{-1} - 0.1395\,q^{-2}}\,q^{-2}\,u_2(t) \\
& + \frac{0.0006 - 0.0014\,q^{-1}}{1 - 0.8498\,q^{-1} - 0.1395\,q^{-2}}\,q^{-2}\,u_3(t) \\
& + \frac{0.0035 - 0.0033\,q^{-1}}{1 - 0.8498\,q^{-1} - 0.1395\,q^{-2}}\,q^{-2}\,u_4(t) \\
& + \frac{1}{1 - 0.8498q^{-1} - 0.1395q^{-2}}\,e(t)
\end{aligned}
\tag{3.13}
$$

$$y2(t) = \frac{0.0055}{1-0.9406q^{-1} + 0.00706q^{-2} - 0.05457q^{-3}} q^{-4} u_1(t)$$

$$+ \frac{0.0029}{1-0.9406q^{-1} + 0.00706q^{-2} - 0.05457q^{-3}} q^{-4} u_2(t)$$

$$+ \frac{-0.000409}{1-0.9406q^{-1} + 0.00706q^{-2} - 0.05457q^{-3}} q^{-4} u_3(t)$$

$$+ \frac{0.0012}{1-0.9406q^{-1} + 0.00706q^{-2} - 0.05457q^{-3}} q^{-4} u_4(t)$$

$$+ \frac{1}{1-0.9406q^{-1} + 0.00706q^{-2} - 0.05457q^{-3}} e(t) \tag{3.14}$$

$$y3(t) = \frac{0.0016 + 0.0027q^{-1}}{1-1.078q^{-1} + 0.09108q^{-2}} q^{-1} u_1(t)$$

$$+ \frac{0.0118 - 0.0049q^{-1}}{1-1.078q^{-1} + 0.09108q^{-2}} q^{-1} u_2(t)$$

$$+ \frac{-0.00000029 + 0.0000007329\ q^{-1}}{1-1.078q^{-1} + 0.09108q^{-2}} q^{-1} u_3(t)$$

$$+ \frac{0.0022 + 0.0024q^{-1}}{1-1.078q^{-1} + 0.09108q^{-2}} q^{-1} u_4(t)$$

$$+ \frac{1}{1-1.078q^{-1} + 0.09108q^{-2}} e(t) \tag{3.15}$$

$$y4(t) = \frac{0.0009885\ q^{-1}}{1-0.594q^{-1} - 0.3902q^{-2}} q^{-8} u_1(t)$$

$$+ \frac{0.0100q^{-1}}{1-0.594q^{-1} - 0.3902q^{-2}} q^{-8} u_2(t)$$

$$+ \frac{0.00037182\ q^{-1}}{1-0.594q^{-1} - 0.3902q^{-2}} q^{-8} u_3(t)$$

$$+ \frac{0.0018q^{-1}}{1-0.594q^{-1} - 0.3902q^{-2}} q^{-8} u_4(t)$$

$$+ \frac{1}{1-0.594q^{-1} - 0.3902q^{-2}} e(t) \tag{3.16}$$

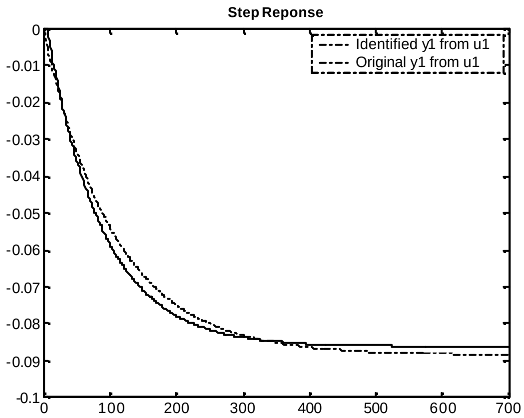

Figure 3.1 : Case 1- Step response of Output-1 from Input-1 (ARX)

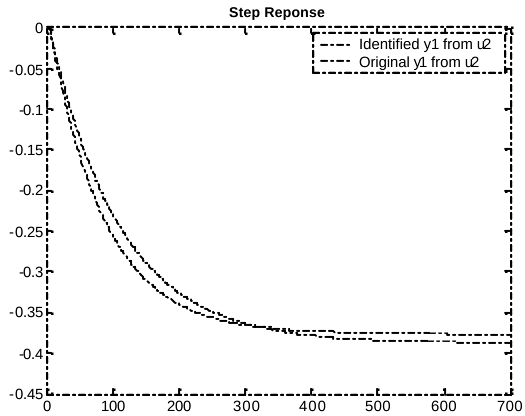

Figure 3.2 : Case 1- Step response of Output-1 from Input-2 (ARX)

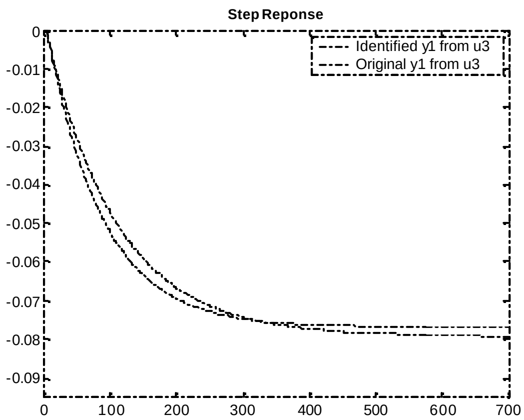

Figure 3.3 : Case 1- Step response of Output-1 from Input-3 (ARX)

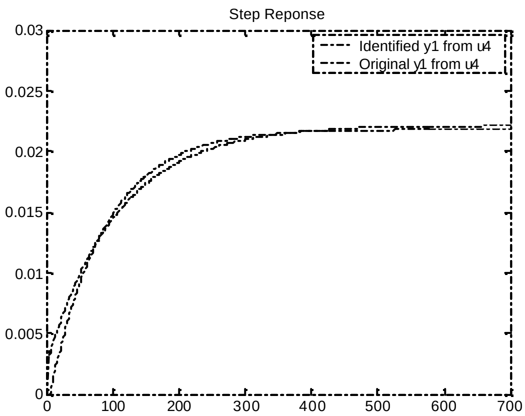

Figure 3.4 : Case 1- Step response of Output-1 from Input-4 (ARX)

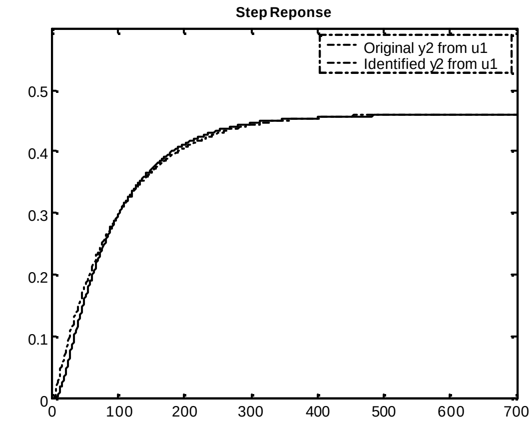

Figure 3.5 : Case 1- Step response of Output-2 from Input-1 (ARX)

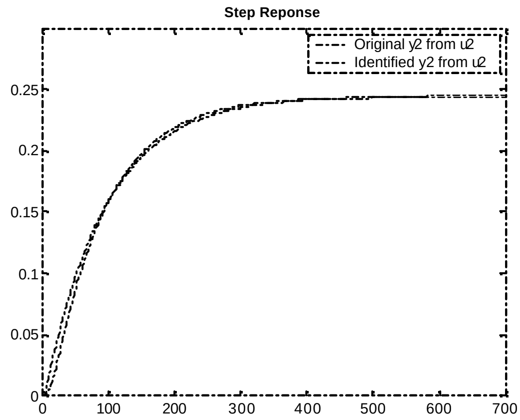

Figure 3.6 : Case 1- Step response of Output-2 from Input-2 (ARX)

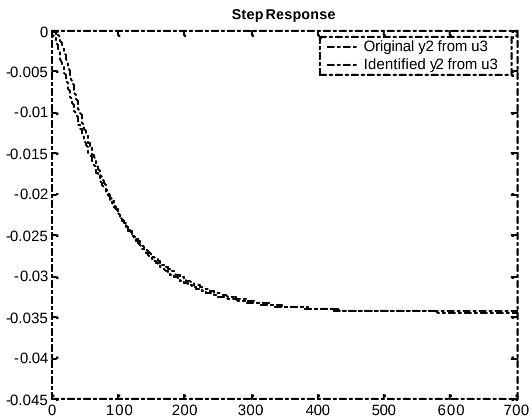

Figure 3.7 : Case 1- Step response of Output-2 from Input-3 (ARX)

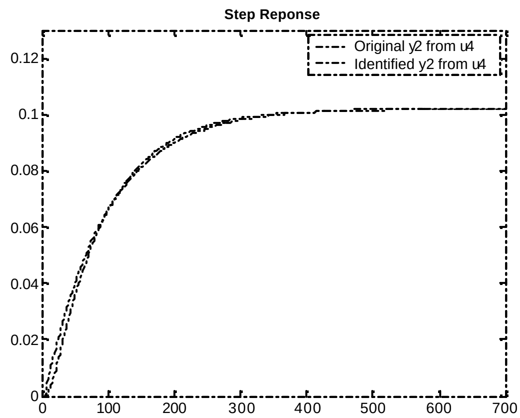

Figure 3.8 : Case 1- Step response of Output-2 from Input-4 (ARX)

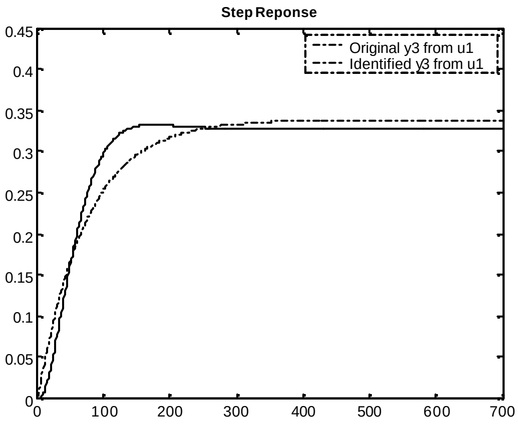

Figure 3.9 : Case 1- Step response of Output-3 from Input-1 (ARX)

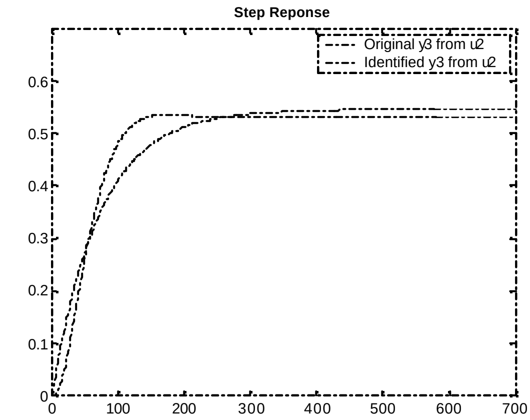

Figure 3.10 : Case 1- Step response of Output-3 from Input-2 (ARX)

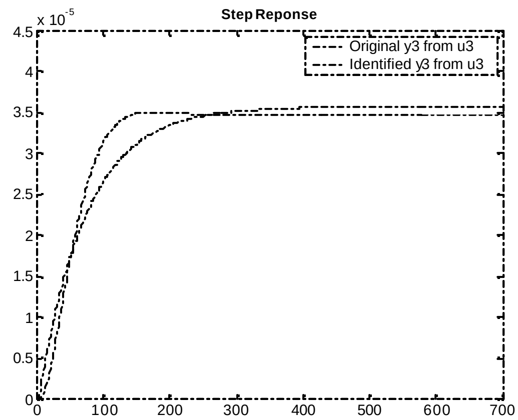

Figure 3.11 : Case 1- Step response of Output-3 from Input-3 (ARX)

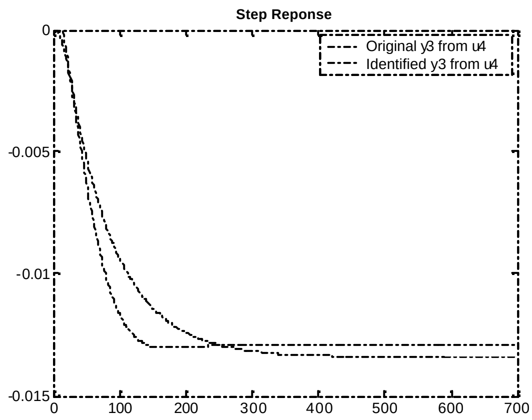

Figure 3.12 : Case 1- Step response of Output-3 from Input-4 (ARX)

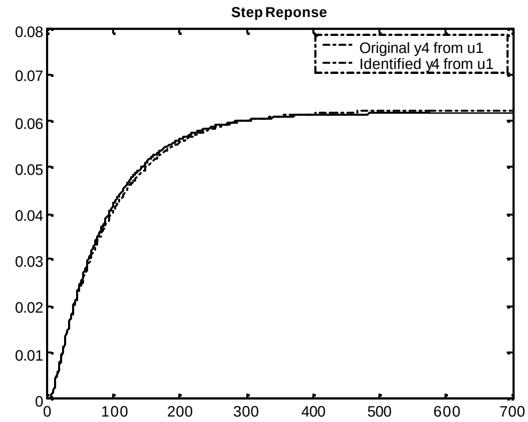

Figure 3.13 : Case 1- Step response of Output-4 from Input-1 (ARX)

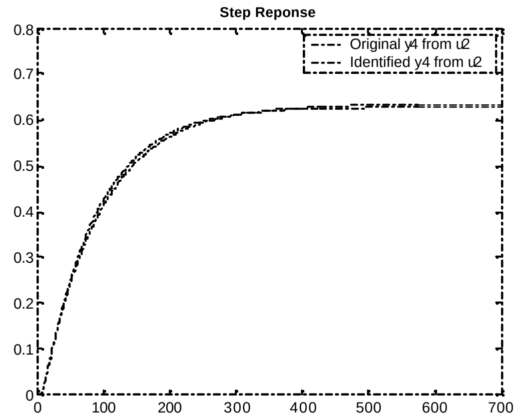

Figure 3.14 : Case 1- Step Response of Output-4 from Input-2 (ARX)

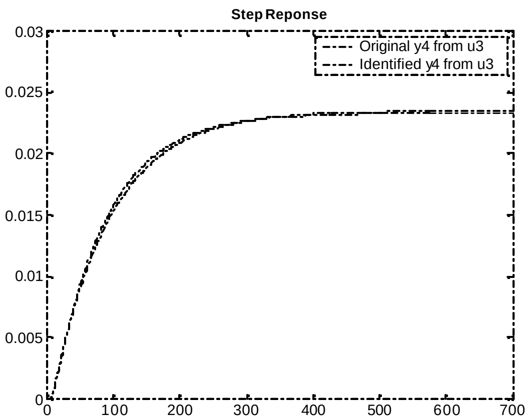

Figure 3.15 : Case 1- Step response of Output-4 from Input-3 (ARX)

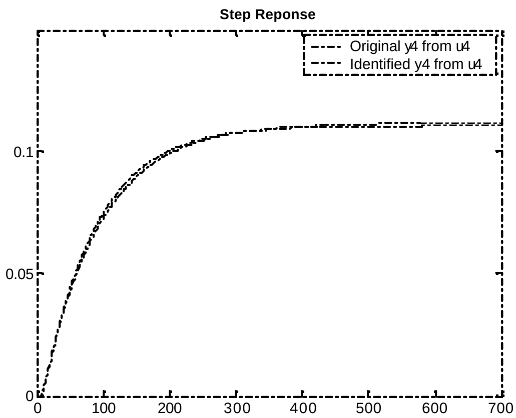

Figure 3.16 : Case 1- Step response of Output-4 from Input-4 (ARX)

3.3.1.2 Performance of ARMAX model

ARMAX models are another type of parametric models. They are sometimes referred in the literature as prediction error ARMAX models. The similarity between ARMAX and ARX models is that the noise and input are subjected to the same dynamics (same poles). This is reasonable if the dominating disturbances enter early in the process (together with the input). This is also a precondition for obtaining a stable model [36]. The difference between ARX and ARMAX models is that the noise in ARMAX modeling scheme is modeled explicitly. Also the numerical complexity is higher for ARMAX scheme because the prediction error involves complex optimization routines.

This modeling scheme is now used for estimating a closed loop model from the simulated MPC closed loop data. The parameters identified are given in Eqs. 3.17 - 3.20. The orders (n_a) of the process outputs are selected as 4, 3, 4 and 4 respectively. As in the case of ARX models, the step responses of both open loop process model and the identified closed loop ARMAX models are plotted in Figs. 3.17 - 3.32. The solid line indicates the current open loop process model and the dashed line corresponds to the direct identified closed loop AMARX model. The results show the excellent performance of the identified closed loop ARMAX model. There is non-trivial bias for the first output as is seen in Figs. 3.17 – 3.20. The results compared to ARX modeling scheme are much better as the steady state gain is a perfect fit. For output 2 the identification results are very accurate and again good steady state fit is achieved with no bias. However, as compared to ARX modeling scheme output 3 in Figs. 3.25- 3.28 has no bias (error) at all and the model manages to capture the steady state part much better. This can be attributed to the fact that in ARMAX modeling scheme noise is being explicitly modeled resulting in better open loop process model. Similarly for output 4 in Figs. 3.29 – 3.32, the results are very good and the open loop dynamics are recovered to a full extent. Thus, this direct identification based ARMAX modeling scheme is able to recover successfully the open loop process model from given closed loop data and captures the steady state part of the responses accurately.

$$
\begin{aligned}
y1(t) = & \frac{-0.0025 + 0.0055q^{-1} - 0.0035q^{-2}}{1 - 1.997q^{-1} + 1.335q^{-2} - 0.1596q^{-3} - 0.172q^{-4}} q^{-2} u_1(t) \\
& + \frac{-0.0086 + 0.0113q^{-1} - 0.0050q^{-2}}{1 - 1.997q^{-1} + 1.335q^{-2} - 0.1596q^{-3} - 0.172q^{-4}} q^{-2} u_2(t) \\
& + \frac{-0.0003 - 0.0014q^{-1} + 0.0012q^{-2}}{1 - 1.997q^{-1} + 1.335q^{-2} - 0.1596q^{-3} - 0.172q^{-4}} q^{-2} u_3(t) \\
& + \frac{0.0016 - 0.0025q^{-1} + 0.0011q^{-2}}{1 - 1.997q^{-1} + 1.335q^{-2} - 0.1596q^{-3} - 0.172q^{-4}} q^{-2} u_4(t) \\
& + \frac{1 - 1.253q^{-1} + 0.6585q^{-2}}{1 - 1.997q^{-1} + 1.335q^{-2} - 0.1596q^{-3} - 0.172q^{-4}} e(t)
\end{aligned}
\tag{3.17}
$$

$$
\begin{aligned}
y2(t) = & \frac{0.0042}{1 - 0.971q^{-1} - 0.1854q^{-2} + 0.1656q^{-3}} q^{-4} u_1(t) \\
& + \frac{0.0022}{1 - 0.971q^{-1} - 0.1854q^{-2} + 0.1656q^{-3}} q^{-4} u_2(t) \\
& + \frac{-0.0003160}{1 - 0.971q^{-1} - 0.1854q^{-2} + 0.1656q^{-3}} q^{-4} u_3(t) \\
& + \frac{0.0009376}{1 - 0.971q^{-1} - 0.1854q^{-2} + 0.1656q^{-3}} q^{-4} u_4(t) \\
& + \frac{1 - 0.2369q^{-1} - 0.2179q^{-2}}{1 - 0.971q^{-1} - 0.1854q^{-2} + 0.1656q^{-3}} e(t)
\end{aligned}
\tag{3.18}
$$

$$y3(t) = \frac{0.0026 - 0.0027q^{-1} + 0.0013q^{-2} + 0.0010q^{-3} - 0.0014q^{-4}}{1 - 1.78q^{-1} + 0.7663q^{-2} - 0.02225q^{-3} + 0.03882q^{-4}} q^{-3} u_1(t)$$

$$+ \frac{0.0009 - 0.0013q^{-1} + 0.0014q^{-2} - 0.0004q^{-3} + 0.0008q^{-4}}{1 - 1.78q^{-1} + 0.7663q^{-2} - 0.02225q^{-3} + 0.03882q^{-4}} q^{-3} u_2(t)$$

$$+ \frac{0.1318e^{-6} + 0.0446e^{-6}q^{-1} + 0.2712e^{-6}q^{-2} - 0.2749e^{-6}q^{-3} - 0.0793e^{-6}q^{-4}}{1 - 1.78q^{-1} + 0.7663q^{-2} - 0.02225q^{-3} + 0.03882q^{-4}} q^{-3} u_3(t)$$

$$+ \frac{-0.3833 + 0.2651q^{-1} + 0.1800q^{-2} - 0.2412q^{-3} + 0.1448q^{-4}}{1 - 1.78q^{-1} + 0.7663q^{-2} - 0.02225q^{-3} + 0.03882q^{-4}} q^{-3} u_4(t)$$

$$+ \frac{1 - 0.8689q^{-1} + 0.0148q^{-2}}{1 - 1.78q^{-1} + 0.7663q^{-2} - 0.02225q^{-3} + 0.03882q^{-4}} q^{-3} e(t) \tag{3.19}$$

$$y4(t) = \frac{0.0024 - 0.0031q^{-1} + 0.0006q^{-2} + 0.0002q^{-3}}{1 - 1.992q^{-1} + 0.7248q^{-2} + 0.6848q^{-3} - 0.4157q^{-4}} q^{-3} u_1(t)$$

$$+ \frac{0.0079 - 0.0057q^{-1} - 0.0150q^{-2} + 0.0025q^{-3}}{1 - 1.992q^{-1} + 0.7248q^{-2} + 0.6848q^{-3} - 0.4157q^{-4}} q^{-3} u_2(t)$$

$$+ \frac{0.0002917 - 0.0003470q^{-1} + 0.00022q^{-2} - 0.0001205q^{-3}}{1 - 1.992q^{-1} + 0.7248q^{-2} + 0.6848q^{-3} - 0.4157q^{-4}} q^{-3} u_3(t)$$

$$+ \frac{-0.0030 - 0.0005q^{-1} + 0.0069q^{-2} - 0.0031q^{-3}}{1 - 1.992q^{-1} + 0.7248q^{-2} + 0.6848q^{-3} - 0.4157q^{-4}} q^{-3} u_4(t)$$

$$+ \frac{1 - 1.629q^{-1} + 0.7376q^{-2}}{1 - 1.992q^{-1} + 0.7248q^{-2} + 0.6848q^{-3} - 0.4157q^{-4}} e(t) \tag{3.20}$$

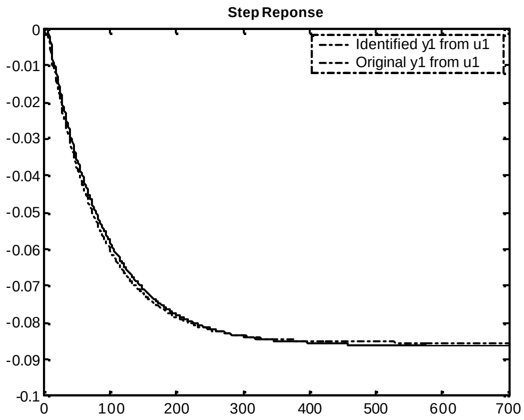

Figure 3.17 : Case 1- Step response of Output-1 from Input-1 (ARMAX)

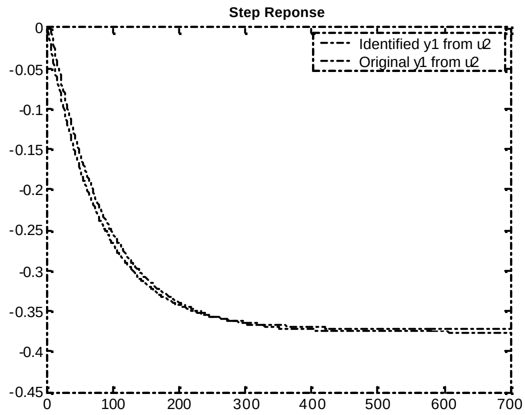

Figure 3.18 : Case 1- Step response of Output-1 from Input-2 (ARMAX)

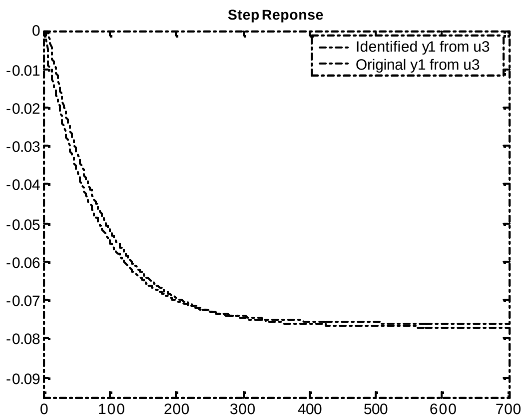

Figure 3.19 : Case 1- Step response of Output-1 from Input-3 (ARMAX)

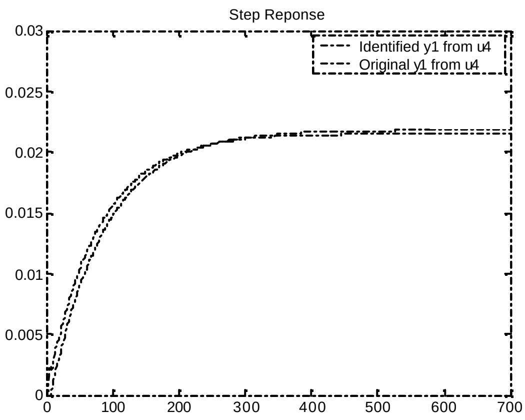

Figure 3.20 : Case 1- Step response of Output-1 from Input-4 (ARMAX)

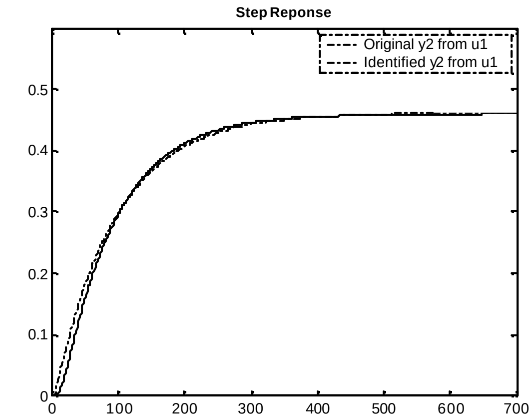

Figure 3.21 : Case 1- Step response of Output-2 from Input-1 (ARMAX)

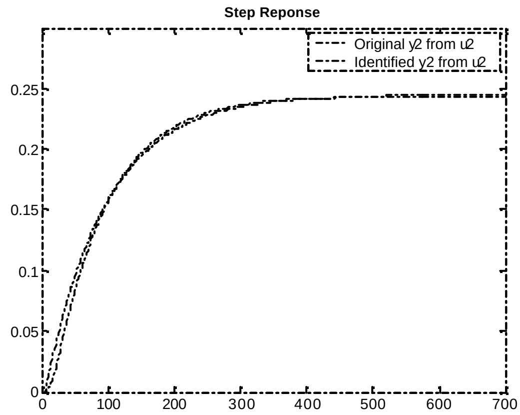

Figure 3.22 : Case 1- Step response of Output-2 from Input-2 (ARMAX)

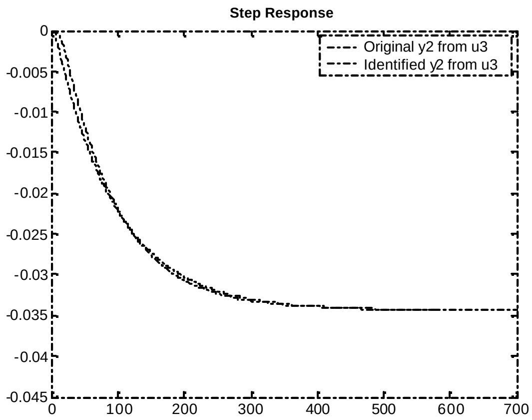

Figure 3.23 : Case 1- Step response of Output-2 from Input-3 (ARMAX)

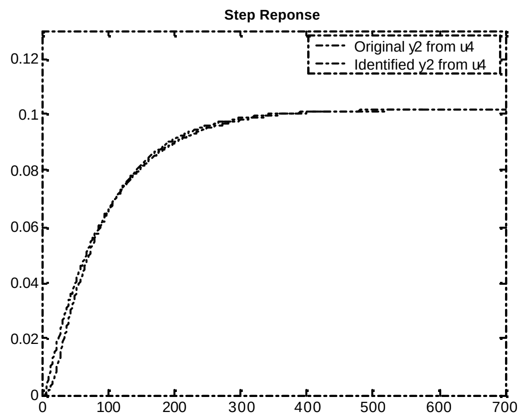

Figure 3.24 : Case 1- Step response of Output-2 from Input-4 (ARMAX)

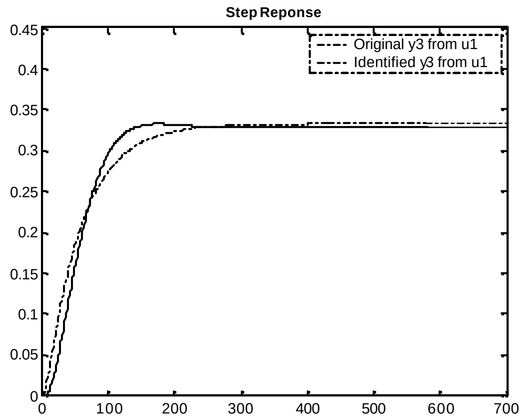

Figure 3.25 : Case 1- Step response of Output-3 from Input-1 (ARMAX)

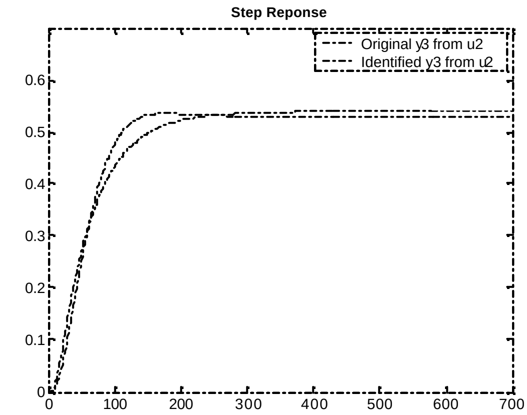

Figure 3.26 : Case 1- Step response of Output-3 from Input-2 (ARMAX)

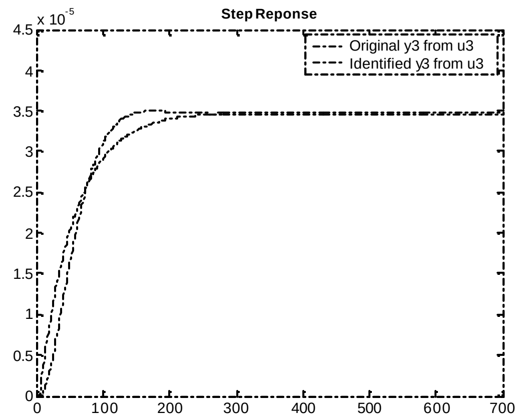

Figure 3.27 : Case 1- Step response of Output-3 from Input-3 (ARMAX)

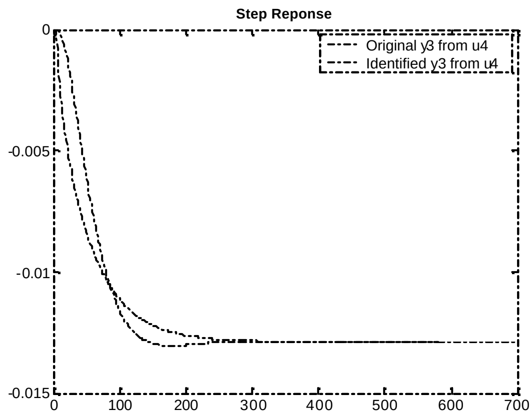

Figure 3.28 : Case 1- Step response of Output-3 from Input-4 (ARMAX)

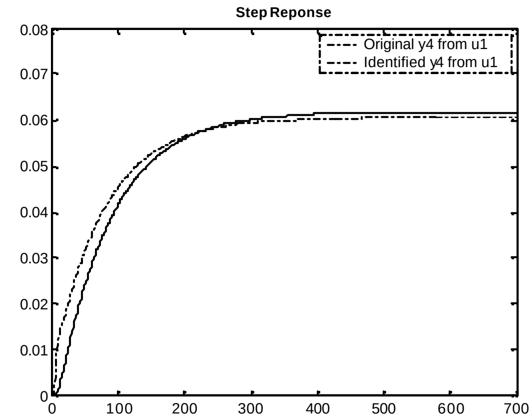

Figure 3.29 : Case 1- Step response of Output-4 from Input-1 (ARMAX)

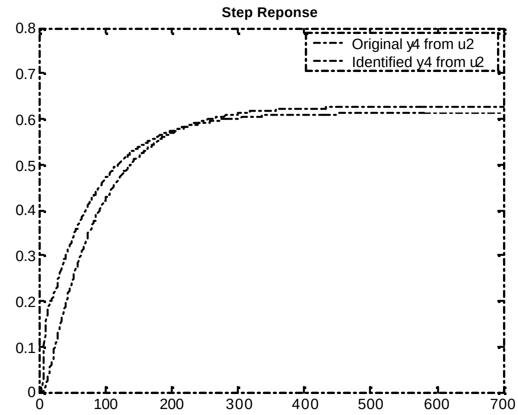

Figure 3.30 : Case 1- Step response of Output-4 from Input-2 (ARMAX)

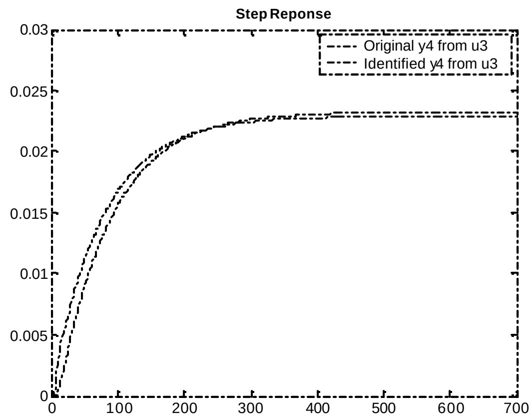

Figure 3.31 : Case 1- Step response of Output-4 from Input-3 (ARMAX)

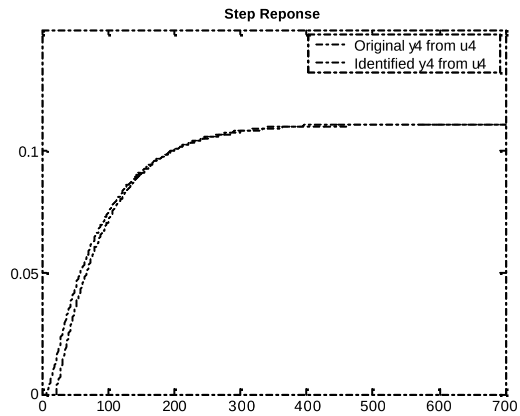

Figure 3.32 : Case 1- Step response of Output-4 from Input-4 (ARMAX)

3.3.1.3 Performance of OE model

The output error models are a special case in which the properties of the disturbance signals are not modeled, and the noise model is chosen fixed as 1. These are used in the case when the purpose is to model the system dynamics only. The noise source *e(t)* in this model is regarded as the difference (error) between the actual output and noise free output. They have been commonly used in literature and are considered a good option in open loop identification schemes, as they produce the most compact (minimum parameters) representation of a plant.

This modeling technique is now used to identify the open loop process model from the closed loop data obtained from simulation. The orders (n_a) of the process outputs are 2, 3, 3, and 3 respectively. The step responses of the identified closed loop OE model (dashed line) are compared with that of the original open loop demathanizer model (solid line) used in simulations of chapter 2. Figs. 3.33 - 3.36 show the step responses of process output 1. The results show large bias and high mismatch between the two models. Only the step response from input 2 is matched. The other dynamics related to inputs 1, 3 and 4 are not captured at all. Figs. 3.37 – 3.40 show the step responses of the process output 2. Again the dynamics related to input 1 and 2 are not modeled accurately. However, the steady state gain fit is captured for input 3 and 4. Figs. 3.41 -3.44 give the step responses of output 3. As is the case with the first two process outputs, the dynamics of input 3 and 4 are not modeled accurately. Figs. 3.45 – 3.48 show the step responses of process output 4. Again large bias and mismatch is observed. In general, the closed loop model exhibits large bias (error) and gives an inaccurate representation of the process dynamics. To an extent, the results are good for outputs 2 and 3, but the steady states are not reached completely. This is one drawback with having a fixed noise model in the OE scheme. The closed loop data from case 1 has high levels of disturbance and noise, which if not modeled at all will result in inaccurate identification of the process models. The parameters identified are given in Eqs. 3.21 - 3.24.

$$
\begin{aligned}
y1(t) = &\frac{-0.0046 + 0.0045q^{-1}}{1-1.586q^{-1}+0.58725q^{-2}} q^{-6}\, u_1(t) \\
&+ \frac{-0.0065+0.0019q^{-1}}{1-1.049q^{-1}+0.06125q^{-2}} q^{-6}\, u_2(t) \\
&+ \frac{-0.0014-0.0016q^{-1}}{1-0.0427q^{-1}-0.9171q^{-2}} q^{-6}\, u_3(t) \\
&+ \frac{0.0041-0.0037\,q^{-1}}{1-1.557\,q^{-1}+0.5752\,q^{-2}} q^{-2}\, u_4(t)
\end{aligned}
\tag{3.21}
$$

$$
\begin{aligned}
y2(t) = &\frac{-0.0026 + 0.0048q^{-1}}{1-1.257q^{-1}+0.5731q^{-2}-0.312q^{-3}} q^{-5}\, u_1(t) \\
&+ \frac{-0.0371+0.5409q^{-1}}{1-1.89q^{-1}+1.087q^{-2}-0.1954q^{-3}} q^{-5}\, u_2(t) \\
&+ \frac{0.6166-0.7674\,q^{-1}}{1-1.063\,q^{-1}-0.493\,q^{-2}+0.5603\,q^{-3}} q^{-5}\, u_3(t) \\
&+ \frac{0.0012-0.0013q^{-1}}{1-2.076q^{-1}+1.295q^{-2}-0.2175q^{-3}} q^{-5}\, u_4(t)
\end{aligned}
\tag{3.22}
$$

$$
\begin{aligned}
y3(t) = &\frac{-0.0006 + 0.0010q^{-1}}{1-1.702q^{-1}+0.49q^{-2}+0.2134q^{-3}} q^{-6}\, u_1(t) \\
&+ \frac{0.0001765+0.0004156q^{-1}}{1-1.704q^{-1}+0.4598q^{-2}+0.245q^{-3}} q^{-6}\, u_2(t) \\
&+ \frac{0.0000004032-0.0000001019q^{-1}}{1-1.925q^{-1}+1.613q^{-2}-0.6797q^{-3}} q^{-6}\, u_3(t) \\
&+ \frac{0.0002433-0.0002576\,q^{-1}}{1-1.427\,q^{-1}-0.08559\,q^{-2}+0.5133\,q^{-3}} q^{-6}\, u_4(t)
\end{aligned}
\tag{3.23}
$$

$$
\begin{aligned}
y4(t) =& \frac{0.0059 - 0.0055q^{-1}}{1-1.668q^{-1}+1.007q^{-2}-0.3339q^{-3}} q^{-3} u_1(t) \\
&+ \frac{-0.0005 + 0.0048q^{-1}}{1-1.889q^{-1}+1.584q^{-2}+0.6889q^{-3}} q^{-3} u_2(t) \\
&+ \frac{0.0013 - 0.0012q^{-1}}{1-2.09q^{-1}+1.279q^{-2}-0.1888q^{-3}} q^{-3} u_3(t) \\
&+ \frac{0.0002433 - 0.0002576\,q^{-1}}{1-1.612\,q^{-1}+1.144q^{-2}-0.494q^{-3}} q^{-3} u_4(t)
\end{aligned}
\tag{3.24}
$$

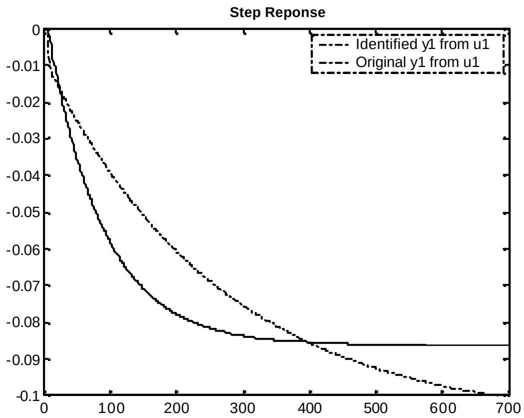

Figure 3.33 : Case 1- Step response of Output-1 from Input-1 (OE)

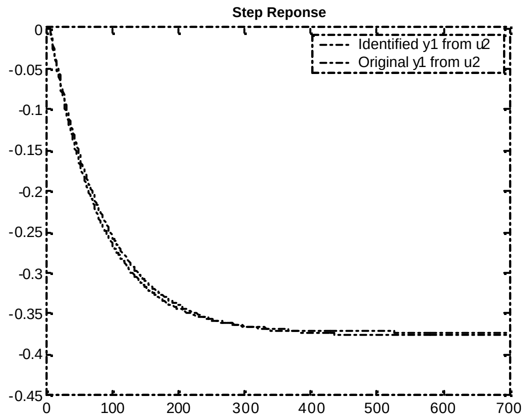

Figure 3.34 : Case 1- Step response of Output-1 from Input-2 (OE)

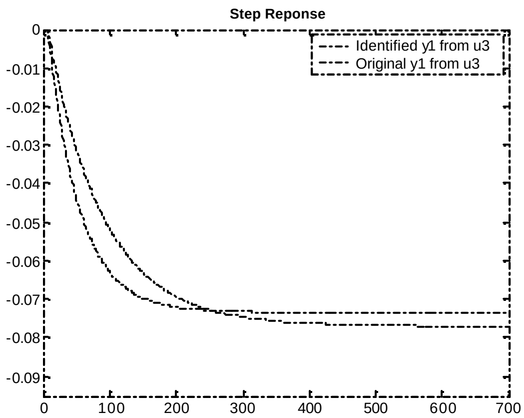

Figure 3.35 : Case 1- Step response of Output-1 from Input-3 (OE)

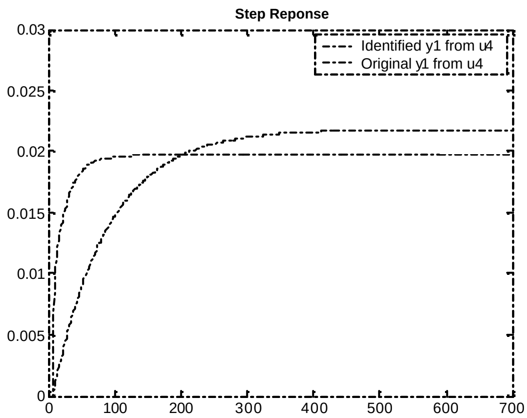

Figure 3.36 : Case 1- Step response of Output-1 from Input-4 (OE)

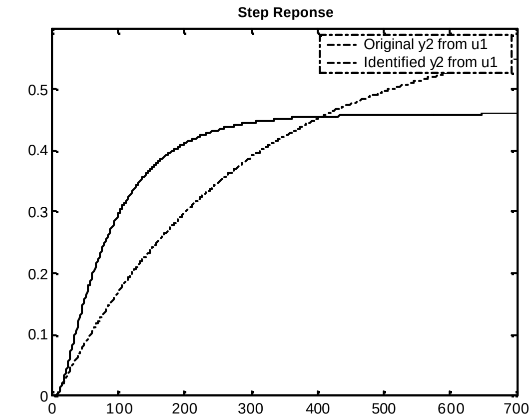

Figure 3.37 : Case 1- Step response of Output-2 from Input-1 (OE)

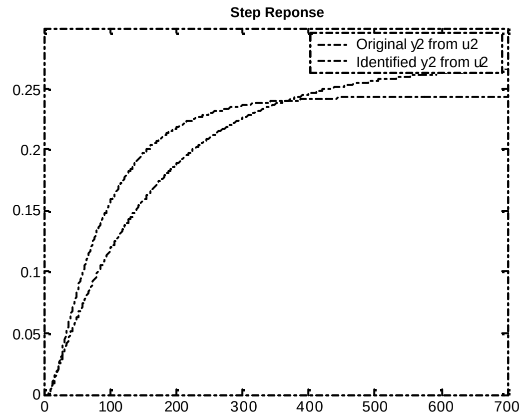

Figure 3.38 : Case 1- Step response of Output-2 from Input-2 (OE)

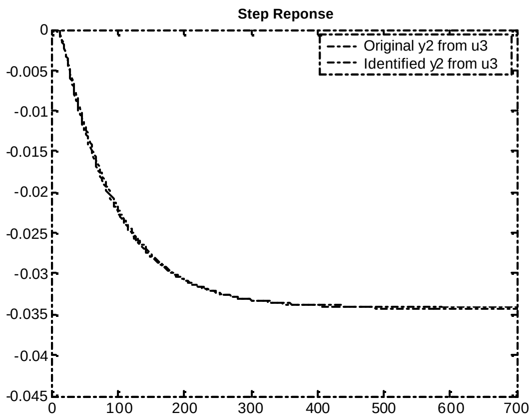

Figure 3.39 : Case 1- Step response of Output-2 from Input-3 (OE)

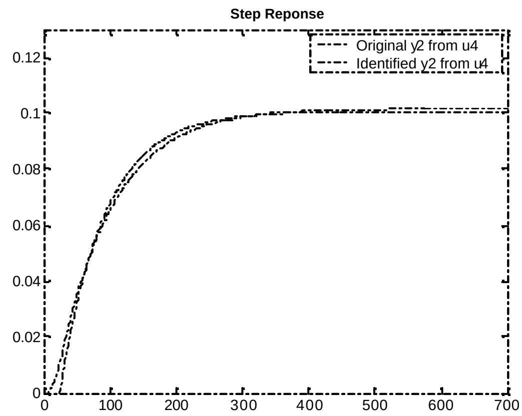

Figure 3.40 : Case 1- Step response of Output-2 from Input-4 (OE)

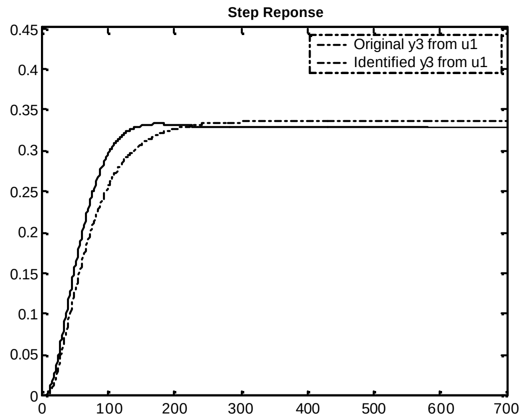

Figure 3.41 : Case 1- Step response of Output-3 from Input-1 (OE)

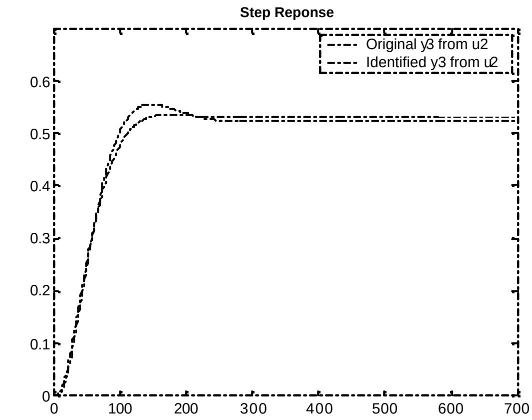

Figure 3.42 : Case 1- Step response of Output-3 from Input-2 (OE)

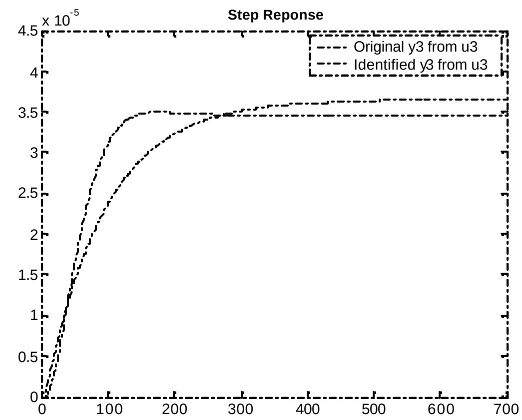

Figure 3.43 : Case 1- Step response of Output-3 from Input-3 (OE)

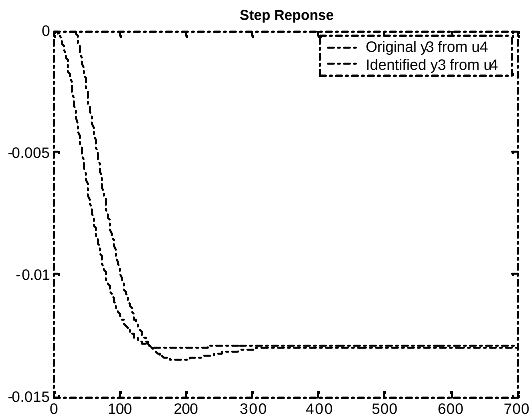

Figure 3.44 : Case 1- Step response of Output-3 from Input-4 (OE)

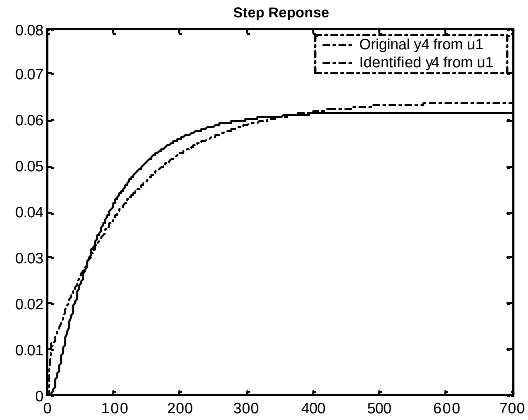

Figure 3.45 : Case 1- Step response of Output-4 from Input-1 (OE)

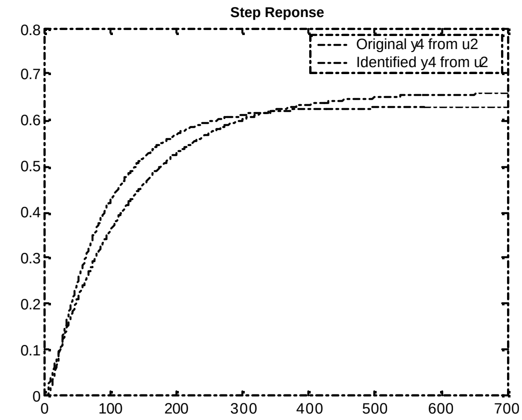

Figure 3.46 : Case 1- Step response of Output-4 from Input-2 (OE)

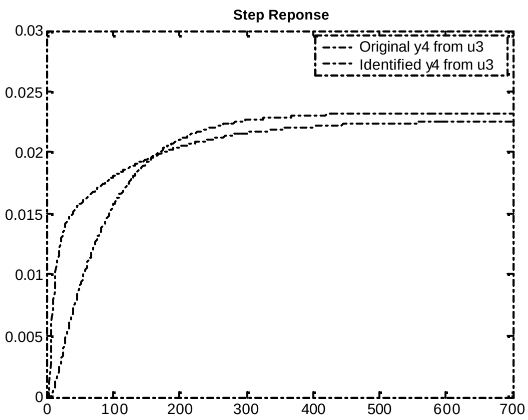

Figure 3.47 : Case 1- Step response of Output-4 from Input-3 (OE)

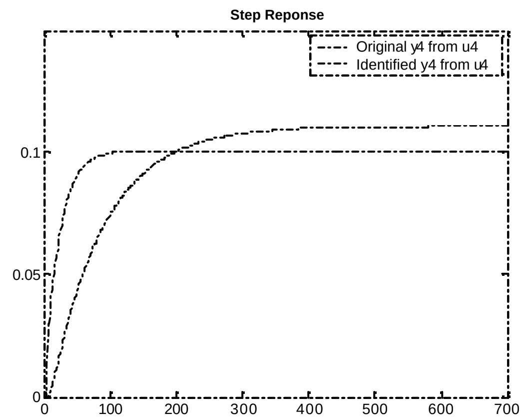

Figure 3.48 : Case 1- Step response of Output-4 from Input-4 (OE)

3.3.1.4 Performance of State space model

In recent years, the method of subspace has been proposed and studied in the literature. One of the most important conceptual ideas behind subspace algorithms is to introduce the concept of the state of a dynamic system within the system identification context. In contrast to 'classical' identification algorithms, subspace algorithms first estimate/calculate the state (sequence), while next the (state space) model is determined. The subspace methods have indeed proven to be a valuable alternative for classical prediction error methods. However, so far it has been shown to be effective and consistent for open loop identification by Overschee and De Moor [65]. Closed loop identification properties of this method have not been investigated thoroughly yet. In this section, this method is applied for closed loop identifying a state space closed loop process model for MPC.

Simulated closed loop data is now used to recover the open loop process model by using state space modeling scheme. The step responses are plotted for the identified closed loop state space model (dashed) and the open loop process model (solid) to assess the performance of this scheme. The results are show in Figs. 3.49 - 3.64, which indicate that this method does not give accurate estimates for all process outputs. Although in all cases the steady state gain has the same sign, but the bias (error) between the original open loop process and the closed loop identified model is very high. The model fails to give accurate description of the system dynamics although the steady state gains of the step responses have the correct sign. Only process output 3 in Figs. 3.57 – 3.60 has a match to an extent but the steady state gain is not a complete fit. The identified model in state space format is given in Eqs. 3.25 - 3.28.

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.96819 & -0.071027 & 0.0469\\ -0.040761 & 0.80309 & 0.33781\\ 0.074647 & -0.02629 & 0.63382\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.044807 & 0.0052186 & -0.0029525 & 0.0079519\\ 0.11228 & 0.016713 & 0.011026 & 0.0085565\\ -0.057569 & -0.014228 & -0.01696 & -0.0024863\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.074384\\ 0.0016245\\ -0.068065\end{bmatrix}e(k)$$

$$y_1(k)=\begin{bmatrix}2.4632 & 0.57793 & -0.30489\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix} \tag{3.25}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.836011 & 0.28918 & 0.049467\\ 0.2386 & 0.52723 & -0.60091\\ 0.046287 & 0.13475 & 0.50331\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.19044 & 0.00027902 & 0.0036205 & 0.050323\\ -0.26384 & 0.031112 & -0.034611 & -0.10683\\ 0.0093863 & 0.043636 & -0.0050572 & -0.04199\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}1.0547\\ 1.0209\\ 0.3111\end{bmatrix}e(k)$$

$$y_2(k)=\begin{bmatrix}0.92618 & 0.071411 & -0.013629\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix} \tag{3.26}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.99028 & -0.028028\\ 0.030334 & 0.84702\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.0023384 & 0.0023438 & -0.00029465 & -0.0010842\\ -0.023894 & -0.0033342 & 0.0050969 & -0.010134\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.16681\\ 0.15793\end{bmatrix}e(k)$$

$$y_3(k)=\begin{bmatrix}3.3089 & 0.063117\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix} \tag{3.27}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix} 0.94559 & -0.23082 & 0.14362 & -0.0056429 \\ -0.007434 & -0.63926 & -0.38118 & -0.070632 \\ 0.17658 & 0.28295 & 0.42116 & -0.4315 \\ 0.0092646 & -0.069312 & 0.26443 & 0.72467 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}$$

$$+\begin{bmatrix} 0.011858 & 0.0063464 & 0.01364 & 0.0016204 \\ -0.088882 & 0.11027 & 0.035391 & -0.20138 \\ -0.025166 & 0.015467 & -0.029619 & -0.076897 \\ 0.014731 & -0.0047745 & 0.022939 & 0.012409 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \end{bmatrix}+\begin{bmatrix} 0.30847 \\ -0.090141 \\ 0.13446 \\ -0.24557 \end{bmatrix}e(k)$$

$$y_4(k)=\begin{bmatrix} 3.0081 & 0.025866 & 0.072346 & -0.020968 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \tag{3.28}$$

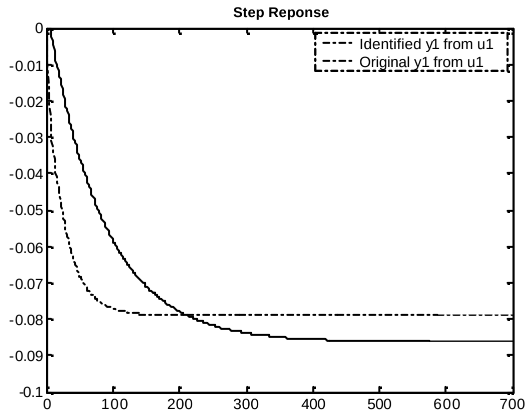

Figure 3.49 : Case 1- Step response of Output-1 from Input-1 (State space)

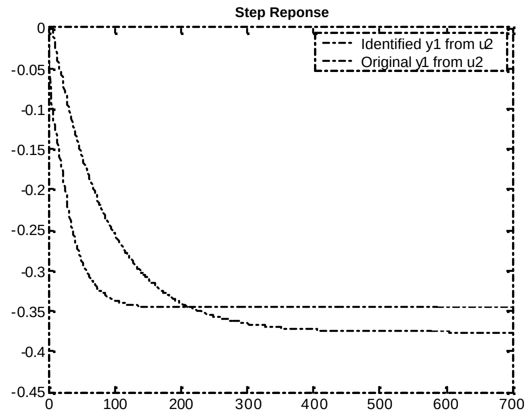

Figure 3.50 : Case 1- Step response of Output-1 from Input-2 (State space)

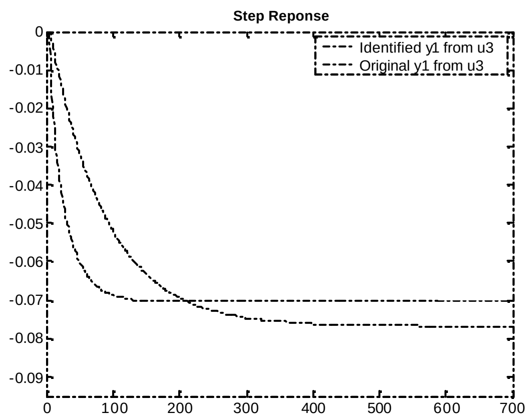

Figure 3.51 : Case 1- Step response of Output-1 from Input-3 (State space)

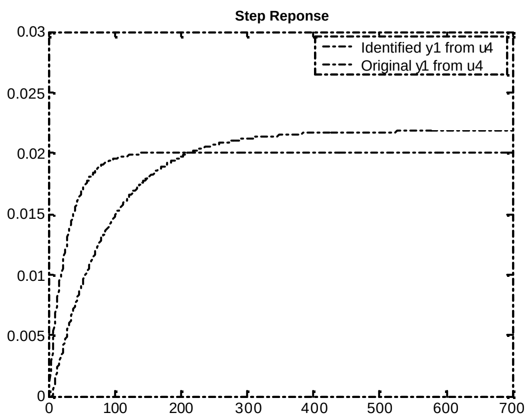

Figure 3.52 : Case 1- Step response of Output-1 from Input-4 (State space)

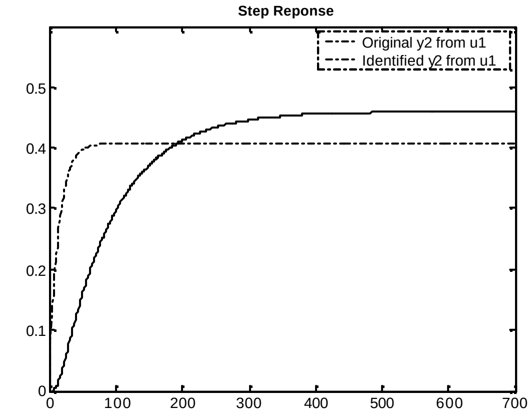

Figure 3.53 : Case 1- Step response of Output-2 from Input-1 (State space)

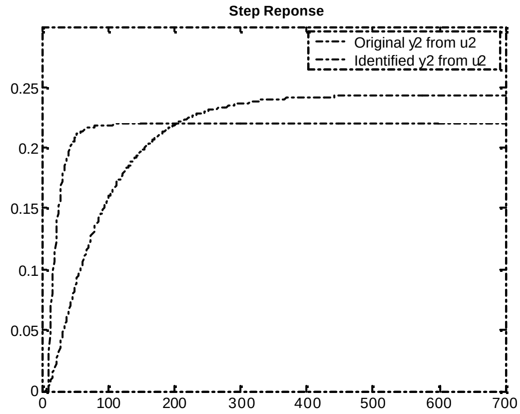

Figure 3.54 : Case 1- Step response of Output-2 from Input-2 (State space)

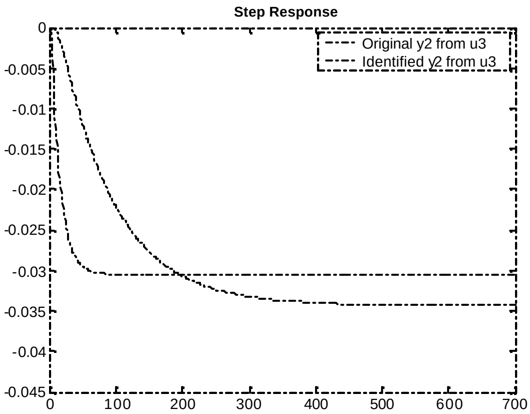

Figure 3.55 : Case 1- Step response of Output-2 from Input-3 (State space)

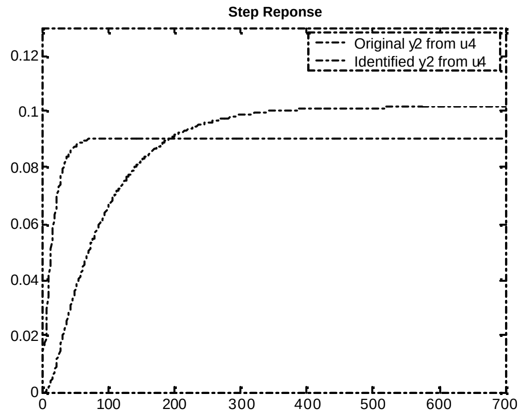

Figure 3.56 : Case 1- Step response of Output-2 from Input-4 (State space)

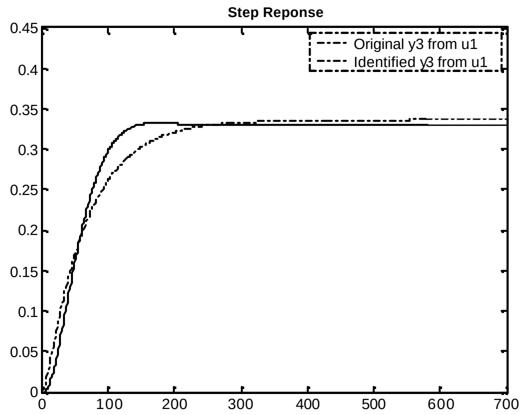

Figure 3.57 : Case 1- Step response of Output-3 from Input-1 (State space)

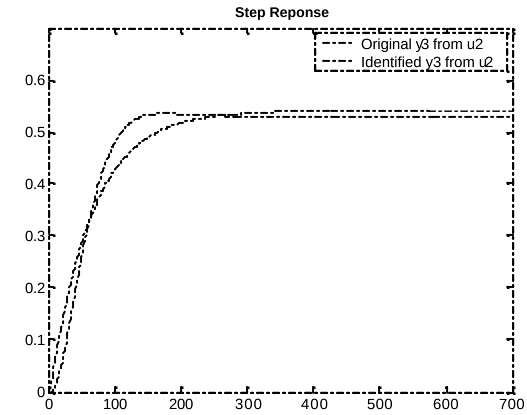

Figure 3.58 : Case 1- Step response of Output-3 from Input-2 (State space)

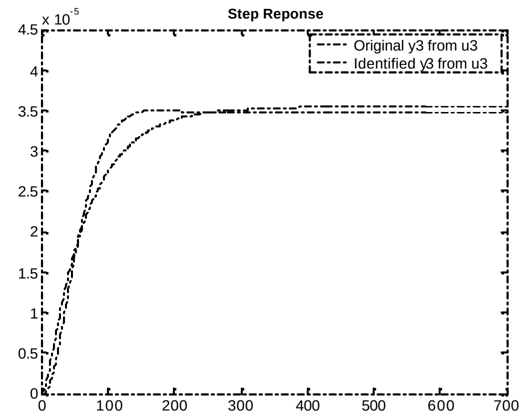

Figure 3.59 : Case 1- Step response of Output-3 from Input-3 (State space)

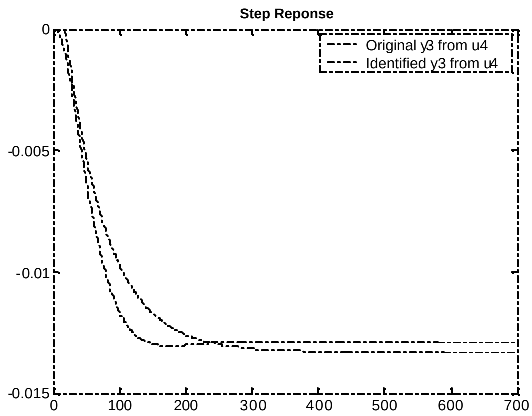

Figure 3.60 : Case 1- Step response of Output-3 from Input-4 (State space)

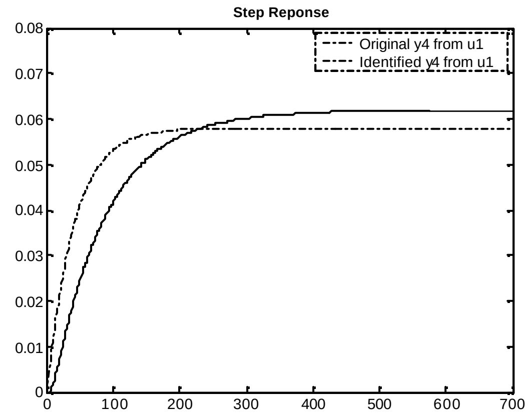

Figure 3.61 : Case 1- Step response of Output-4 from Input-1 (State space)

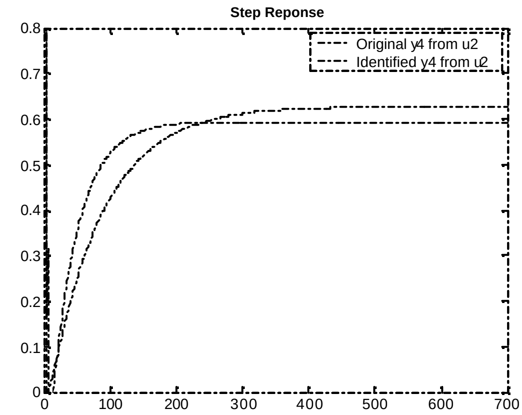

Figure 3.62 : Case 1- Step response of Output-4 from Input-2 (State space)

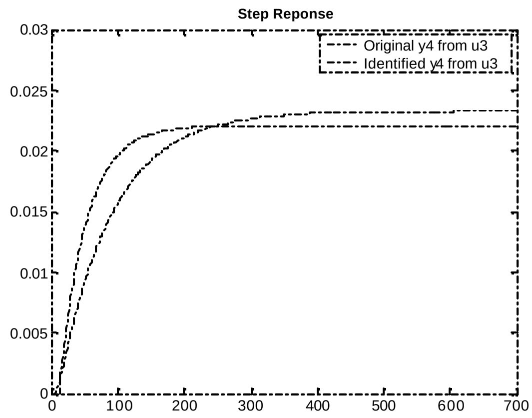

Figure 3.63 : Case 1- Step response of Output-4 from Input-3 (State space)

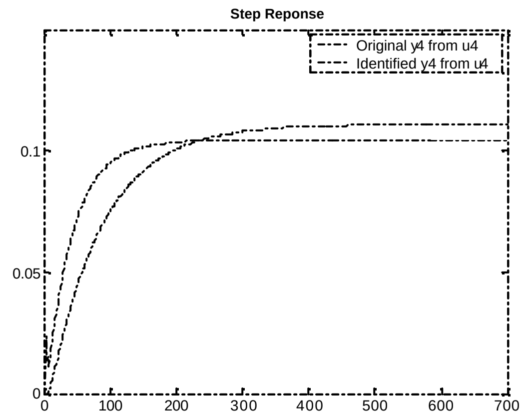

Figure 3.64 : Case 1- Step response of Output-4 from Input-4 (State space)

3.3.2 Case 2 – Closed loop Identification

The performance of the four modeling techniques namely ARX, ARMAX, OE and state space are now analyzed on MPC simulated closed loop data obtained by reducing the disturbances and adding rate constraints (see Sec. 2.5.5). The simulations in case-1 have shown that OE and state space models give biased results with closed loop MPC data probably due to the presence of high disturbance and nonlinearities in the data. In this set of identification simulations, the disturbances have been reduced to a standard deviation of 0.08 and constraints on the rate of change of inputs have been added. Taking this into consideration, all of the modeling techniques are once again employed and the results are discussed individually.

3.3.2.1 Performance of ARX model

ARX modeling scheme is used to identify the open loop process model using the simulated closed loop data. The unknown parameters are estimated by using least squares. The orders (n_a) of the process outputs are 2, 3, 5 and 3 respectively. For comparison the step responses of the closed loop identified ARX model (dashed line) are plotted against the actual open loop process model (solid line). Figs. 3.65 - 3.68 show the step responses of the identified process output 1 and the actual open loop process output 1. There is practically non mismatch between the two models. The closed loop identified ARX model has recovered the information of the open loop process output 1 exactly. Figs. 3.71 – 3.72 give the step responses of the identified model with that of the actual open loop model. Again there is a perfect match between the two models. Figs. 3.73 – 3.76 give the step responses of the identified process output 3 with that of the open loop process output 3.

Although perfect steady state gain fit is not achieved but overall there is non-trivial bias with very little mismatch between the two models. Figs. 3.77 – 3.78 shows the step responses of the closed loop identified process output 4 with that of the original open loop process output 4. Again there is a perfect match between the two and the steady state part of the response is captured accurately. Generally, the four-input four-output closed loop model has no mismatch with the open loop process model, revealing the extraordinary performance of ARX modeling technique on closed loop MPC data. It is by now clear that ARX modeling scheme is an ideal candidate for closed loop identification of processes running with MPC. The parameters identified are given in Eqs. 3.29 - 3.32. It can be seen that the poles of the process model and the noise model are the same which is a characteristic of the ARX scheme. As compared to case 1, the results for this case are very good. It can be concluded that with reduced disturbances the performance of ARX improves towards perfection.

$$
\begin{aligned}
y1(t) = {} & \frac{-0.0009292 + 0.0000402\, q^{-1}}{1 - 1.114\, q^{-1} + 0.1248\, q^{-2}} q^{-4}\, u_1(t) \\
& + \frac{0.0026 - 0.0065\, q^{-1}}{1 - 1.114\, q^{-1} + 0.1248\, q^{-2}} q^{-4}\, u_2(t) \\
& + \frac{-0.0012 + 0.0005\, q^{-1}}{1 - 1.114\, q^{-1} + 0.1248\, q^{-2}} q^{-4}\, u_3(t) \\
& + \frac{-0.0017 + 0.0020 q^{-1}}{1 - 1.114 q^{-1} + 0.1248 q^{-2}} q^{-4}\, u_4(t) \\
& + \frac{1}{1 - 1.114\, q^{-1} + 0.1248\, q^{-2}} e(t)
\end{aligned}
\tag{3.29}
$$

$$y2(t) = \frac{0.0059}{1-0.9914q^{-1}+0.1072q^{-2}-0.103q^{-3}}q^{-4}u_1(t)$$

$$+\frac{0.0031}{1-0.9914q^{-1}+0.1072q^{-2}-0.103q^{-3}}q^{-4}u_2(t)$$

$$+\frac{-0.0004371}{1-0.9914q^{-1}+0.1072q^{-2}-0.103q^{-3}}q^{-4}u_3(t)$$

$$+\frac{0.0013}{1-0.9914q^{-1}+0.1072q^{-2}-0.103q^{-3}}q^{-4}u_4(t)$$

$$+\frac{1}{1-0.9914q^{-1}+0.1072q^{-2}-0.103q^{-3}}e(t) \tag{3.30}$$

$$y3(t) = \frac{0.0028}{1-1.131q^{-1}+0.03485q^{-2}+0.05062q^{-3}-0.0551q^{-4}+0.1094q^{-5}}q^{-1}u_1(t)$$

$$+\frac{0.0046}{1-1.131q^{-1}+0.03485q^{-2}+0.05062q^{-3}-0.0551q^{-4}+0.1094q^{-5}}q^{-1}u_2(t)$$

$$+\frac{0.0000002989}{1-1.131q^{-1}+0.03485q^{-2}+0.05062q^{-3}-0.0551q^{-4}+0.1094q^{-5}}q^{-1}u_3(t)$$

$$+\frac{-0.0001115}{1-1.131q^{-1}+0.03485q^{-2}+0.05062q^{-3}-0.0551q^{-4}+0.1094q^{-5}}q^{-1}u_4(t)$$

$$+\frac{1}{1-1.131q^{-1}+0.03485q^{-2}+0.05062q^{-3}-0.0551q^{-4}+0.1094q^{-5}}e(t) \tag{3.31}$$

$$y4(t) = \frac{0.0008196}{1-0.7321q^{-1}-0.2386q^{-2}-0.01624q^{-3}}q^{-1}u_1(t)$$

$$+\frac{0.0083}{1-0.7321q^{-1}-0.2386q^{-2}-0.01624q^{-3}}q^{-1}u_2(t)$$

$$+\frac{0.0003084}{1-0.7321q^{-1}-0.2386q^{-2}-0.01624q^{-3}}q^{-1}u_3(t)$$

$$+\frac{0.0015}{1-0.7321q^{-1}-0.2386q^{-2}-0.01624q^{-3}}q^{-1}u_4(t)$$

$$+\frac{1}{1-0.7321q^{-1}-0.2386q^{-2}-0.01624q^{-3}}e(t) \tag{3.32}$$

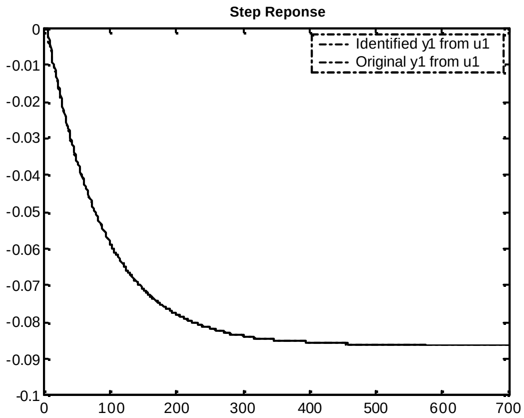

Figure 3.65 : Case 2- Step response of Output-1 from Input-1 (ARX)

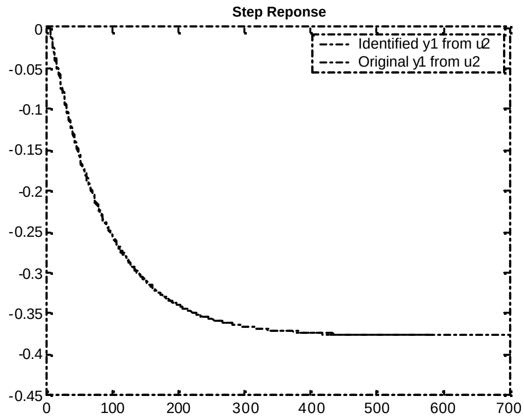

Figure 3.66 : Case 2- Step response of Output-1 from Input-2 (ARX)

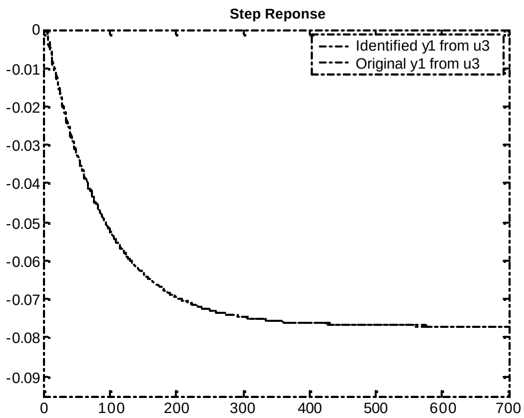

Figure 3.67 : Case 2- Step response of Output-1 from Input-3 (ARX)

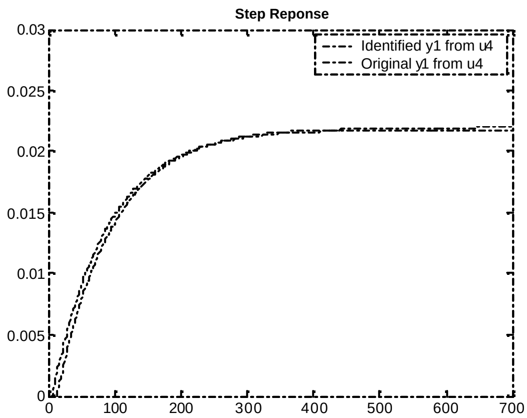

Figure 3.68 : Case 2- Step response of Output-1 from Input-4 (ARX)

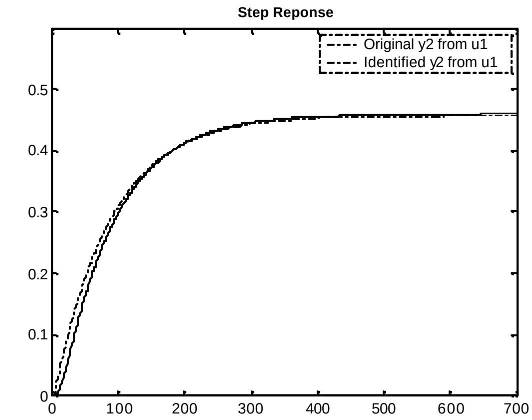

Figure 3.69 : Case 2- Step response of Output-2 from Input-1 (ARX)

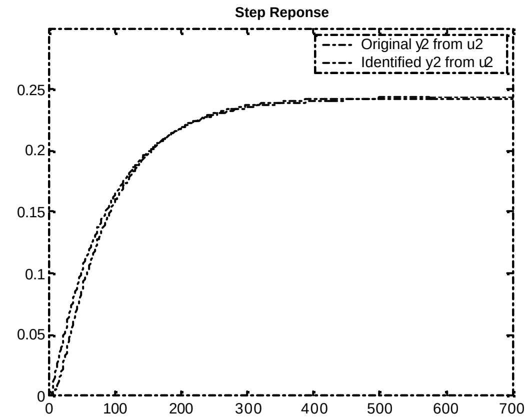

Figure 3.70 : Case 2- Step response of Output-2 from Input-2 (ARX)

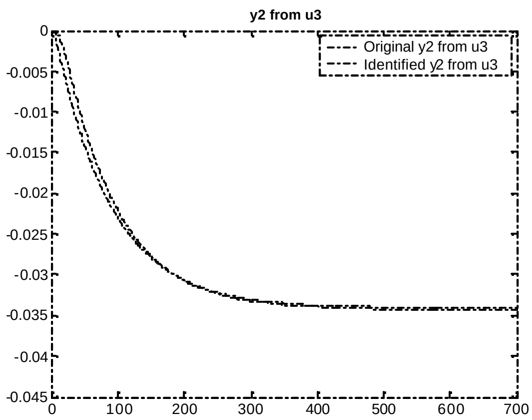

Figure 3.71 : Case 2- Step response of Output-2 from Input-3 (ARX)

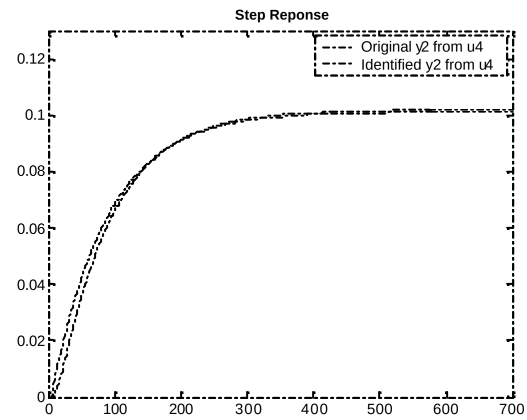

Figure 3.72 : Case 2- Step response of Output-2 from Input-4 (ARX)

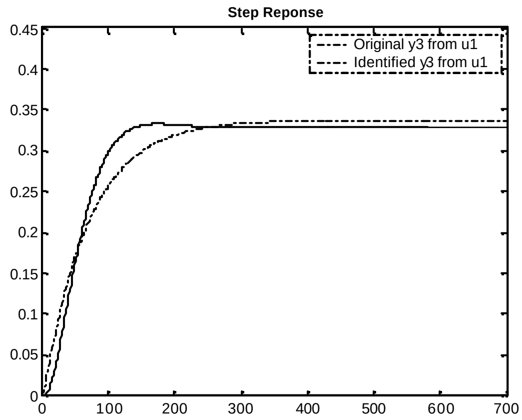

Figure 3.73 : Case 2- Step response of Output-3 from Input-1 (ARX)

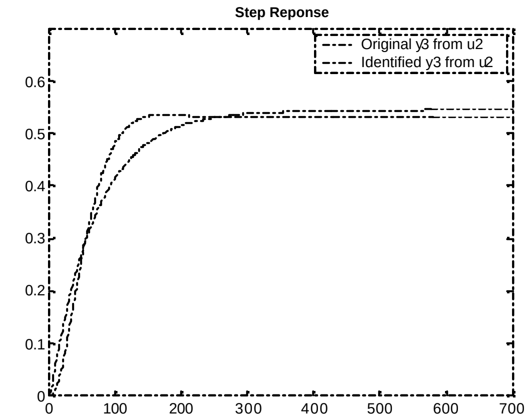

Figure 3.74 : Case 2- Step response of Output-3 from Input-2 (ARX)

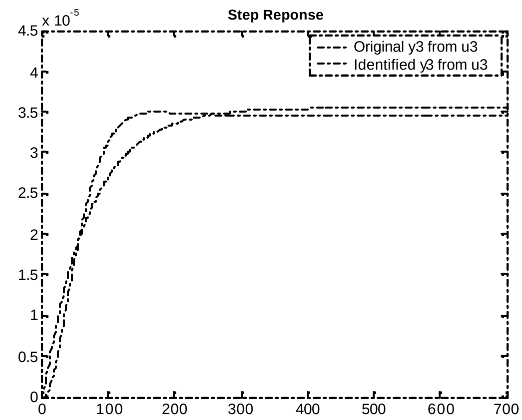

Figure 3.75 : Case 2- Step response of Output-3 from Input-3 (ARX)

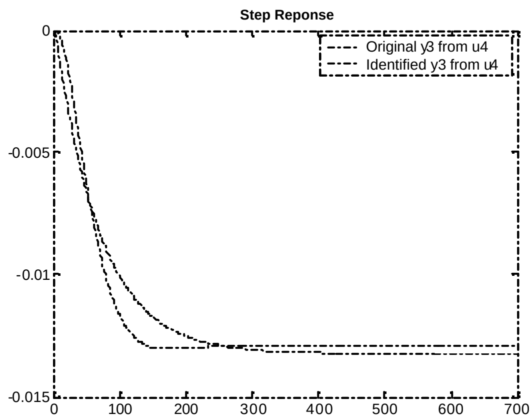

Figure 3.76 : Case 2- Step response of Output-3 from Input-4 (ARX)

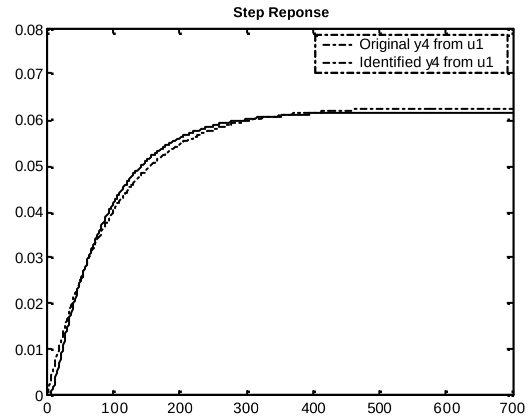

Figure 3.77 : Case 2- Step response of Output-4 from Input-1 (ARX)

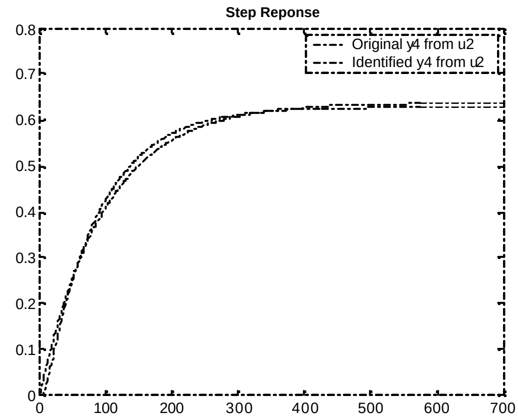

Figure 3.78 : Case 2- Step response of Output-4 from Input-2 (ARX)

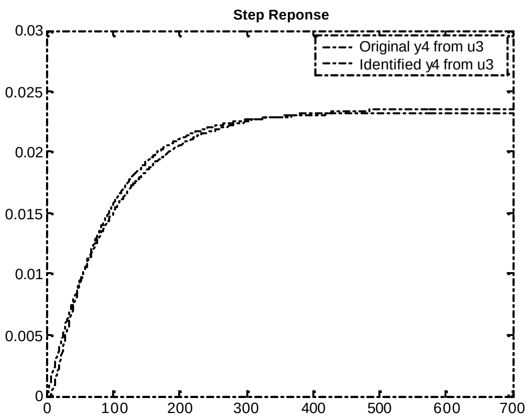

Figure 3.79 : Case 2- Step response of Output-4 from Input-3 (ARX)

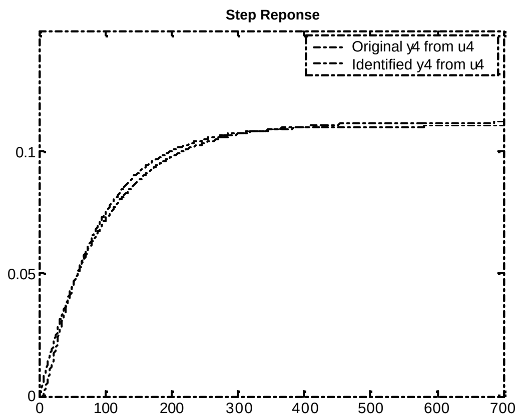

Figure 3.80 : Case 2- Step response of Output-4 from Input-4 (ARX)

3.3.2.2 Performance of ARMAX model

As mentioned in case 1, the difference between ARX and ARMAX models is that the noise in ARMAX modeling scheme is modeled explicitly. This in turn requires complex computations and optimization routines resulting in higher numerical complexity. However, it has also been observed that ARMAX modeling and estimation technique gives good estimates of the open loop process in the case where MPC is subjected to high disturbance. In case 2, it is again used for identifying a closed loop model from simulated MPC closed loop data.

Prediction error method is used to estimate the unknown parameters of the ARMAX model. The orders (n_a) of the process outputs are selected as 5, 3, 2 and 4 respectively. Figs. 3.81 – 3.84 show the step responses of the identified closed loop process model (dashed) versus the actual open loop process model (solid) for output 1. The results indicate a perfect match of the two models. There is no bias and the steady state part has been captured accurately. Figs. 3.85 -3.87 shows the step responses of the closed loop identified ARMAX model with that of the open loop model for process output 2. Again there is no mismatch between the two models. Figs. 3.89 – 3.92 shows the step response of the closed loop identified ARMAX model with that of the open loop process model for output 3. Compared to ARX the mismatch is trivial. This can be attributed to the fact that the disturbance is explicitly modeled in ARMAX scheme and therefore the process model is much more accurate. Similarly, the step responses of the ARMAX closed loop identified model are shown against the actual open loop process model for output 4 in Figs. 3.93 – 3.96. The results again demonstrate that closed loop ARMAX model gives the best fit and manages to capture the steady state part accurately. The parameters identified are given in Eqs. 3.33 - 3.36.

$$y1(t) = \frac{-0.0011}{1-0.9045q^{-1}-0.06391q^{-2}-0.007259q^{-3}+0.0044q^{-4}-0.016q^{-5}} q^{-1}\,u_1(t)$$

$$+\frac{-0.0048}{1-0.9045q^{-1}-0.06391q^{-2}-0.007259q^{-3}+0.0044q^{-4}-0.016q^{-5}} q^{-1}\,u_2(t)$$

$$+\frac{-0.0009823}{1-0.9045q^{-1}-0.06391q^{-2}-0.007259q^{-3}+0.0044q^{-4}-0.016q^{-5}} q^{-1}\,u_3(t)$$

$$+\frac{0.0002786}{1-0.9045q^{-1}-0.06391q^{-2}-0.007259q^{-3}+0.0044q^{-4}-0.016q^{-5}} q^{-1}\,u_4(t)$$

$$+\frac{1+0.1293\,q^{-1}}{1-0.9045\,q^{-1}-0.06391\,q^{-2}-0.007259\,q^{-3}+0.0044\,q^{-4}-0.016\,q^{-5}} e(t) \tag{3.33}$$

$$y2(t) = \frac{0.0018}{1-1.572q^{-1}-0.5296q^{-2}+0.0489q^{-3}} q^{-4}\,u_1(t)$$

$$+\frac{0.0009314}{1-1.572q^{-1}-0.5296q^{-2}+0.0489q^{-3}} q^{-4}\,u_2(t)$$

$$+\frac{-0.0001310}{1-1.572q^{-1}-0.5296q^{-2}+0.0489q^{-3}} q^{-4}\,u_3(t)$$

$$+\frac{0.0003886}{1-1.572q^{-1}-0.5296q^{-2}+0.0489q^{-3}} q^{-4}\,u_4(t)$$

$$+\frac{1-0.7295\,q^{-1}-0.007288\,q^{-2}}{1-1.572\,q^{-1}-0.5296\,q^{-2}+0.0489\,q^{-3}} e(t) \tag{3.34}$$

$$y3(t) = \frac{0.0062 - 0.0054\,q^{-1}}{1 - 1.834\,q^{-1} + 0.8368\,q^{-2}} q^{-1}\,u_1(t)$$

$$+ \frac{0.0041 - 0.0028\,q^{-1}}{1 - 1.834\,q^{-1} + 0.8368\,q^{-2}} q^{-1}\,u_2(t)$$

$$+ \frac{0.000001132 - 0.000001051\,q^{-1}}{1 - 1.834\,q^{-1} + 0.8368\,q^{-2}} q^{-1}\,u_3(t)$$

$$+ \frac{0.0000822 - 0.1128\,q^{-1}}{1 - 1.834\,q^{-1} + 0.8368\,q^{-2}} q^{-1}\,u_4(t)$$

$$+ \frac{1 - 0.6897\,q^{-1}}{1 - 1.834\,q^{-1} + 0.8368\,q^{-2}} q^{-1} e(t) \tag{3.35}$$

$$y4(t) = \frac{0.0012}{1 - 0.1843q^{-1} - 0.6085q^{-2} - 0.1837q^{-3} - 0.003761q^{-4}} q^{-1}\,u_1(t)$$

$$+ \frac{0.0127}{1 - 0.1843q^{-1} - 0.6085q^{-2} - 0.1837q^{-3} - 0.003761q^{-4}} q^{-1}\,u_2(t)$$

$$+ \frac{0.0004686}{1 - 0.1843q^{-1} - 0.6085q^{-2} - 0.1837q^{-3} - 0.003761q^{-4}} q^{-1}\,u_3(t)$$

$$+ \frac{0.0020}{1 - 0.1843q^{-1} - 0.6085q^{-2} - 0.1837q^{-3} - 0.003761q^{-4}} q^{-1}\,u_4(t)$$

$$+ \frac{1 + 0.5136q^{-1}}{1 - 0.1843q^{-1} - 0.6085q^{-2} - 0.1837q^{-3} - 0.003761q^{-4}} e(t) \tag{3.36}$$

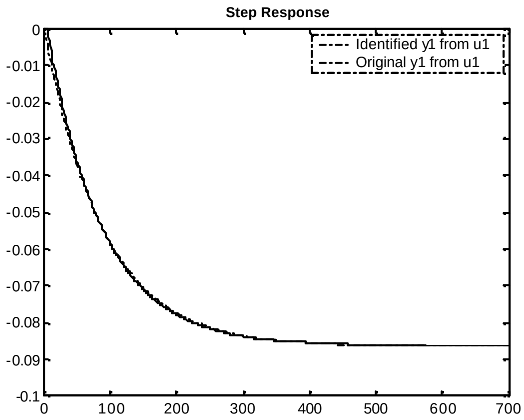

Figure 3.81 : Case 2- Step response of Output-1 from Input-1 (ARMAX)

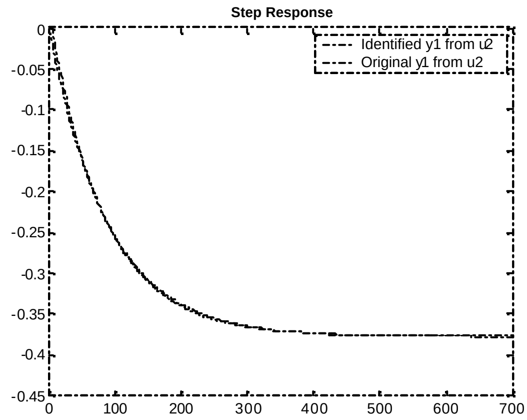

Figure 3.82 : Case 2- Step response of Output-1 from Input-2 (ARMAX)

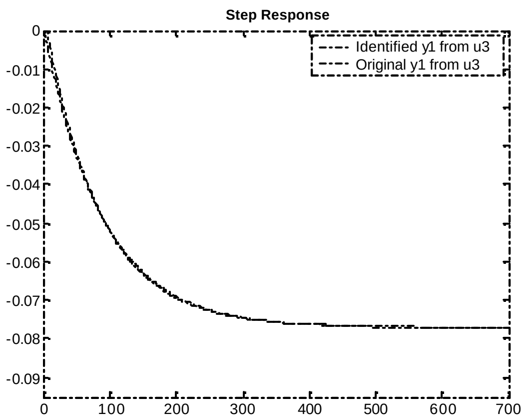

Figure 3.83 : Case 2- Step response of Output-1 from Input-3 (ARMAX)

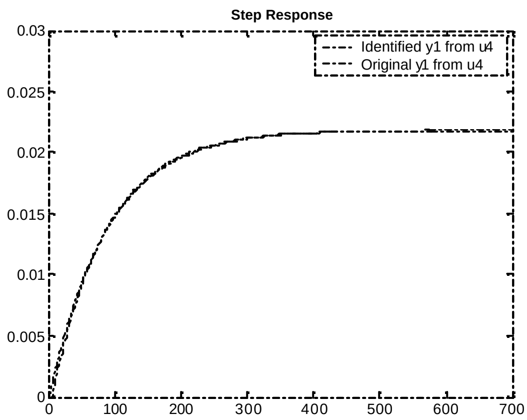

Figure 3.84 : Case 2- Step response of Output-1 from Input-4 (ARMAX)

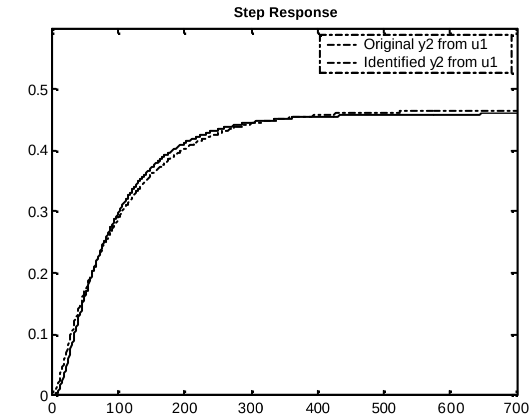

Figure 3.85 : Case 2- Step response of Output-2 from Input-1 (ARMAX)

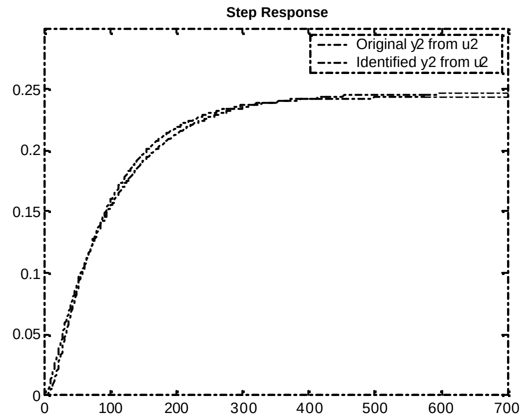

Figure 3.86 : Case 2- Step response of Output-2 from Input-2 (ARMAX)

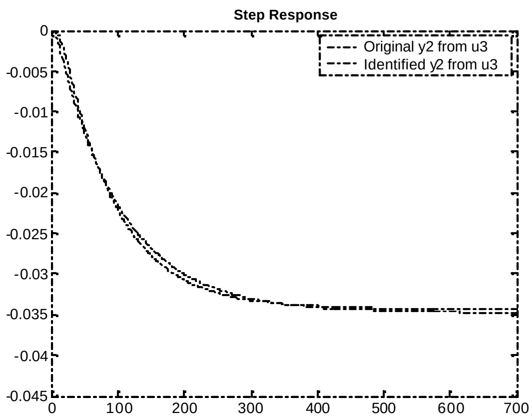

Figure 3.87 : Case 2- Step response of Output-2 from Input-3 (ARMAX)

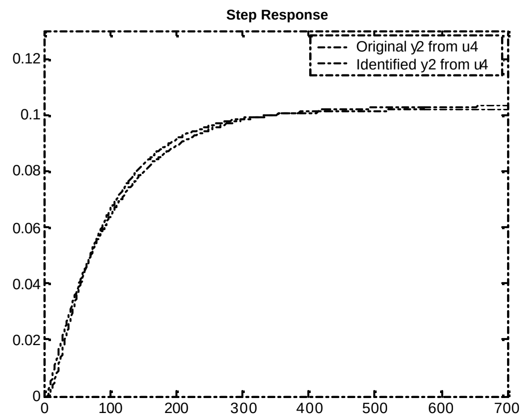

Figure 3.88 : Case 2- Step response of Output-2 from Input-4 (ARMAX)

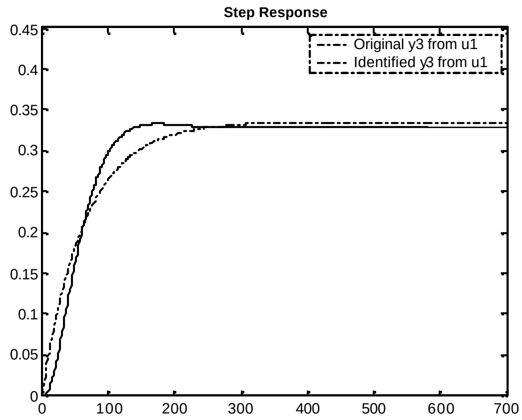

Figure 3.89 : Case 2- Step response of Output-3 from Input-1 (ARMAX)

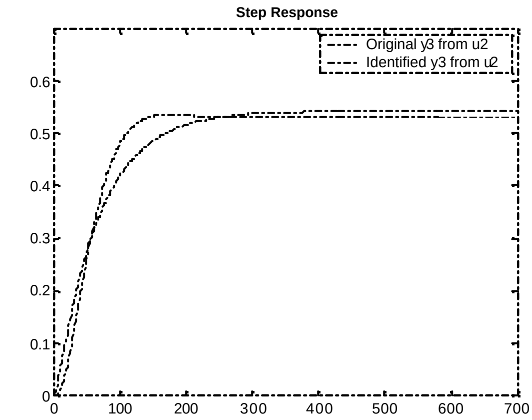

Figure 3.90 : Case 2- Step response of Output-3 from Input-2 (ARMAX)

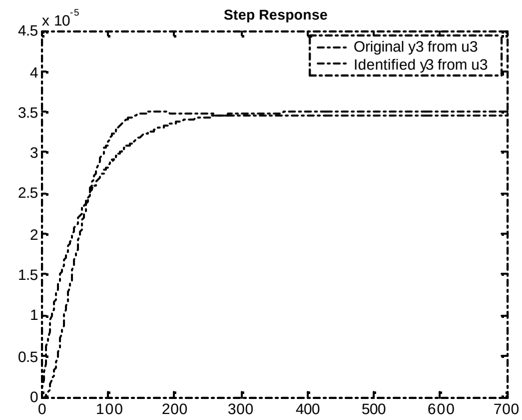

Figure 3.91 : Case 2- Step response of Output-3 from Input-3 (ARMAX)

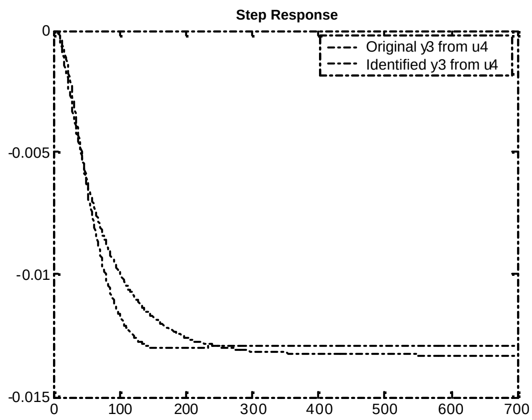

Figure 3.92 : Case 2- Step response of Output-3 from Input-4 (ARMAX)

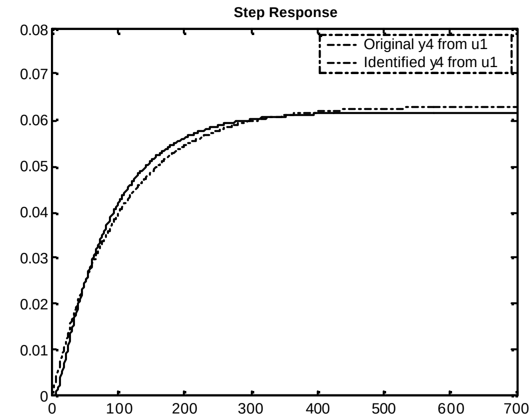

Figure 3.93 : Case 2- Step response of Output-4 from Input-1 (ARMAX)

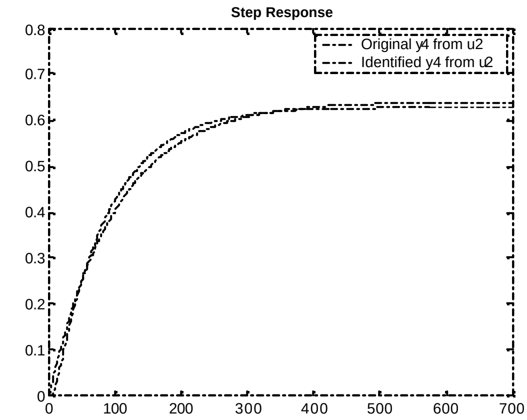

Figure 3.94 : Case 2- Step response of Output-4 from Input-2 (ARMAX)

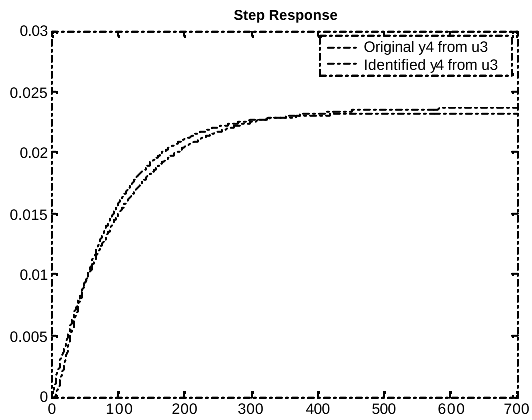

Figure 3.95 : Case 2- Step response of Output-4 from Input-3 (ARMAX)

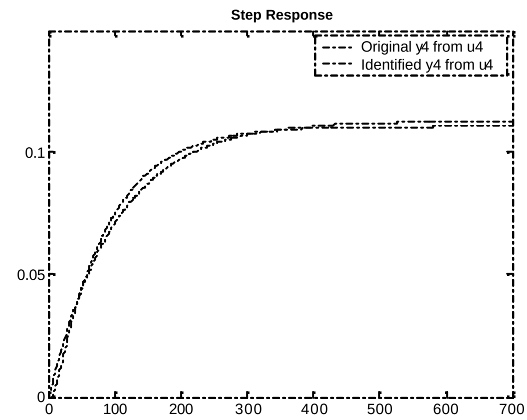

Figure 3.96 : Case 2- Step response of Output-4 from Input-4 (ARMAX)

3.3.2.3 Performance of OE model

In case 1 it was seen that this technique did not give good models due to the presence of high disturbances and noise in the simulated closed loop data. The identification capability of this modeling scheme is now tested for this case where the disturbances have been reduced to σ =0.08. With reduced disturbances, it is hoped that this scheme will yield better results than in case 1.

The procedure adopted is similar to case 1. The orders (n_a) of the process outputs are selected as 4, 2, 3 and 2 respectively. The step responses of the actual open loop (solid) and identified closed loop OE model (dashed) for process output 1 are shown in Figs. 3.97 - 3.100. The results show only a slight improvement in the performance of OE model. The step responses from inputs 1 and 2 are clearly mismatched. However, compared to case 1 the step responses from input 3 and 4 are greatly improved and the mismatch is non-trivial. In Figs. 3.101- 3.104, the step responses of the actual open loop (solid) and identified closed loop OE model (dashed) for process output 2 are shown. Again a marked improvement is observed at the step response from input 1. But the other dynamics from input 2, 3 and 4 are highly mismatched and unstable. Figs. 3.105 - 3.108 gives the step responses for process output 3. There is very little mismatch between the step responses from inputs 1, 2 and 4. However, the step response from input 3 indicates that the model is mismatched and is not able to capture the entire dynamics. The step responses of the process output 4 are shown in Figs. 3.109 – 3.112. It is obvious that the models are mismatched with large bias. Overall, mismatch remains and the closed loop model is largely inaccurate. The parameters identified are given in Eqs. 3.37 - 3.40.

$$y1(t) = \frac{-0.0147 + 0.0145q^{-1}}{1 - 1.506q^{-1} + 0.3559q^{-2} + 0.1536q^{-3} - 0.004191q^{-4}} q^{-1} u_1(t)$$
$$+ \frac{-0.0092 - 0.0044q^{-1}}{1 - 0.4794q^{-1} - 0.3684q^{-2} + 0.1814q^{-3} - 0.2955q^{-4}} q^{-1} u_2(t)$$
$$+ \frac{-0.0014 - 0.0016q^{-1}}{1 - 0.8088q^{-1} + 0.2115q^{-2} - 0.4407q^{-3} + 0.05092q^{-4}} q^{-1} u_3(t)$$
$$+ \frac{0.0041 - 0.0037q^{-1}}{1 - 1.733q^{-1} + 1.058q^{-2} - 0.3393q^{-3} + 0.01878q^{-4}} q^{-1} u_4(t) \tag{3.37}$$

$$y2(t) = \frac{-0.0051 + 0.0100q^{-1}}{1 - 1.064q^{-1} + 0.07846q^{-2}} q^{-6} u_1(t)$$
$$+ \frac{0.0006202 - 0.0003725q^{-1}}{1 - 1.77q^{-1} + 0.771q^{-2}} q^{-6} u_2(t)$$
$$+ \frac{-0.2593 + 0.1164q^{-1}}{1 - 0.05609q^{-1} - 0.9437q^{-2}} q^{-6} u_3(t)$$
$$+ \frac{-0.0074 - 0.0088q^{-1}}{1 - 1.72q^{-1} + 0.7355q^{-2}} q^{-6} u_4(t) \tag{3.38}$$

$$y3(t) = \frac{0.0067}{1 - 2.063q^{-1} + 1.983q^{-2} - 0.8988q^{-3}} q^{-3} u_1(t)$$
$$+ \frac{0.0031}{1 - 1.56q^{-1} + 0.3813q^{-2} + 0.1851q^{-3}} q^{-3} u_2(t)$$
$$+ \frac{0.00000001259}{1 - 1.208q^{-1} - 0.3784q^{-2} + 0.5864q^{-3}} q^{-3} u_3(t)$$
$$+ \frac{-0.000009756}{1 - 2.201q^{-1} + 1.447q^{-2} - 0.2454q^{-3}} q^{-3} u_4(t) \tag{3.39}$$

$$
\begin{aligned}
y4(t) = {} & \frac{0.0557 - 0.0567q^{-1}}{1 - 0.6725q^{-1} - 0.291q^{-2}} q^{-5} u_1(t) \\
& + \frac{0.0108 + 0.0109q^{-1}}{1 + 0.02613q^{-1} - 0.9901q^{-2}} q^{-5} u_2(t) \\
& + \frac{0.0001882 + 0.0001859q^{-1}}{1 + 0.01375q^{-1} - 0.9987q^{-2}} q^{-5} u_3(t) \\
& + \frac{0.0426 - 0.0404\, q^{-1}}{1 - 0.7543\, q^{-1} - 0.2246\, q^{-2}} q^{-5} u_4(t)
\end{aligned}
\tag{3.40}
$$

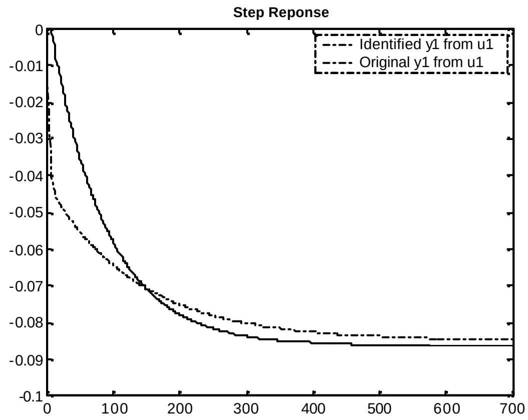

Figure 3.97 : Case 2- Step response of Output-1 from Input-1 (OE)

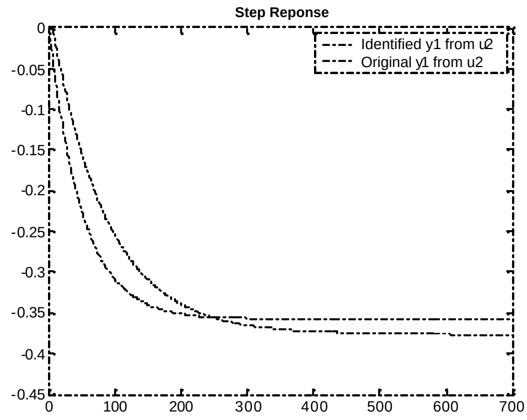

Figure 3.98 : Case 2- Step response of Output-1 from Input-2 (OE)

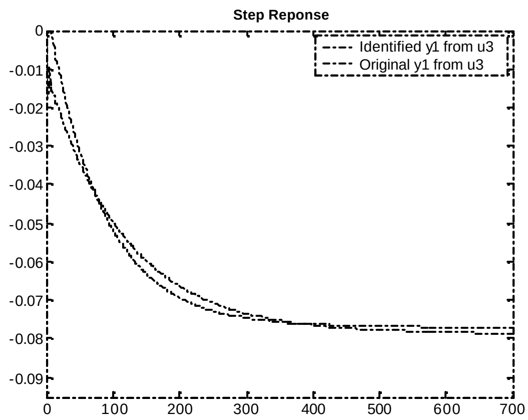

Figure 3.99 : Case 2- Step response of Output-1 from Input-3 (OE)

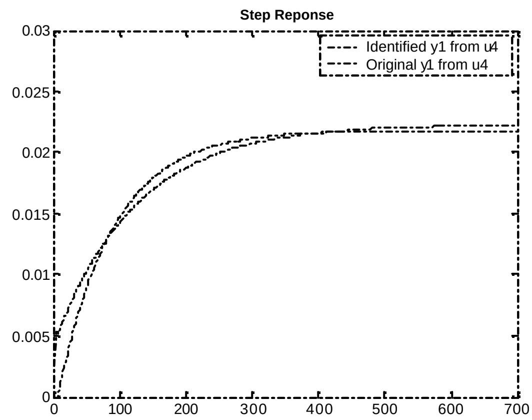

Figure 3.100 : Case 2- Step response of Output-1 from Input-4 (OE)

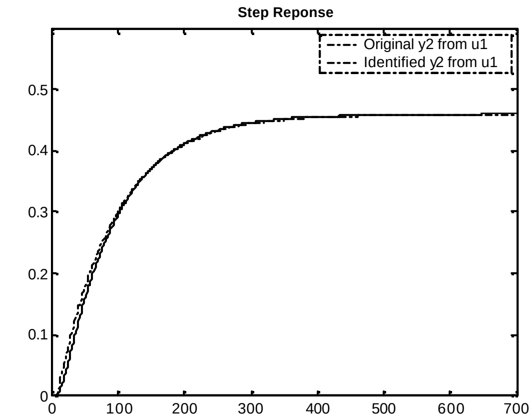

Figure 3.101 : Case 2- Step response of Output-2 from Input-1 (OE)

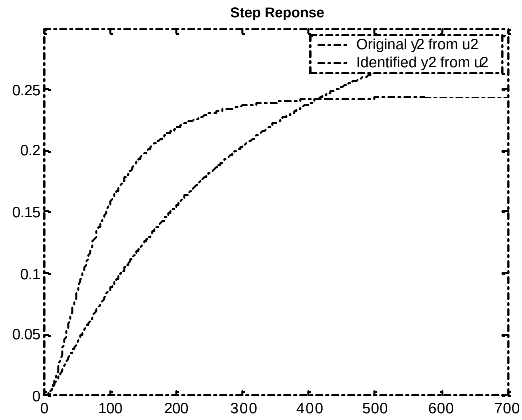

Figure 3.102 : Case 2- Step response of Output-2 from Input-2 (OE)

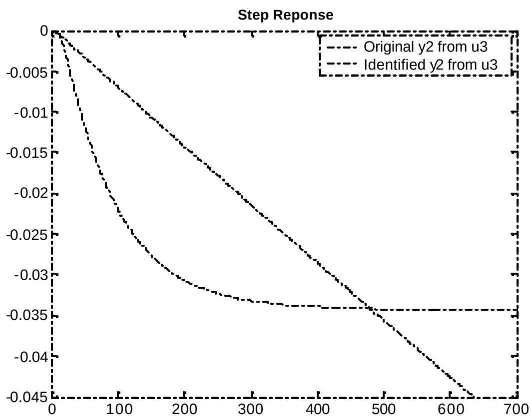

Figure 3.103 : Case 2- Step response of Output-2 from Input-3 (OE)

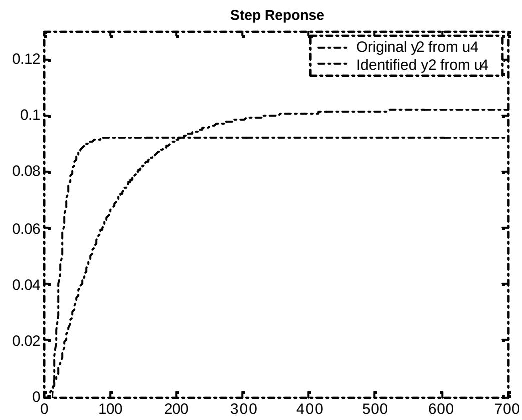

Figure 3.104 : Case 2- Step response of Output-2 from Input-4 (OE)

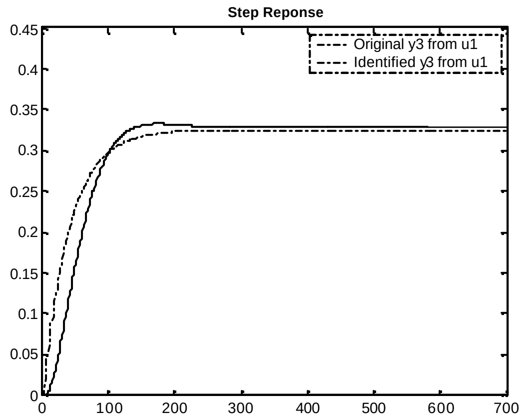

Figure 3.105 : Case 2- Step response of Output-3 from Input-1 (OE)

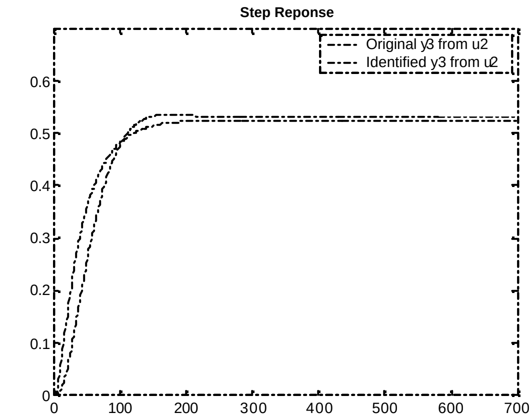

Figure 3.106 : Case 2- Step response of Output-3 from Input-2 (OE)

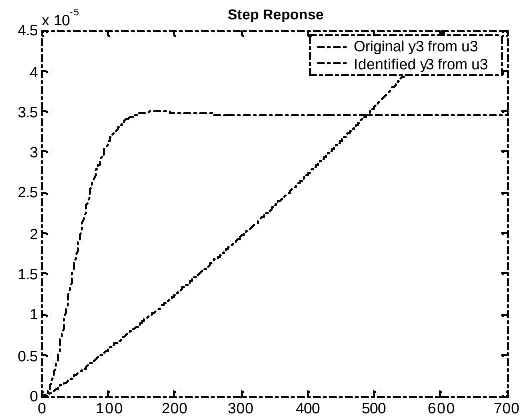

Figure 3.107 : Case 2- Step response of Output-3 from Input-3 (OE)

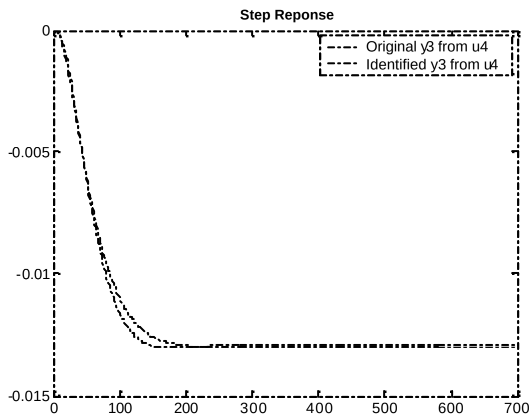

Figure 3.108 : Case 2- Step response of Output-3 from Input-4 (OE)

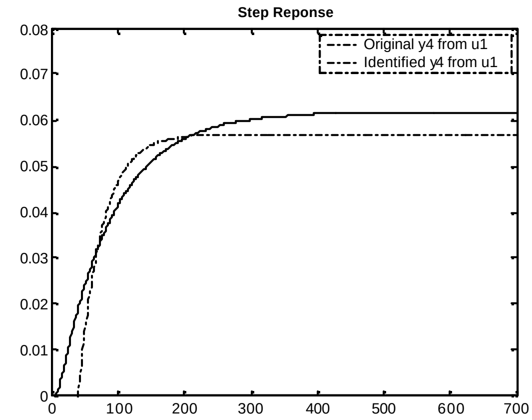

Figure 3.109 : Case 2- Step response of Output-4 from Input-1 (OE)

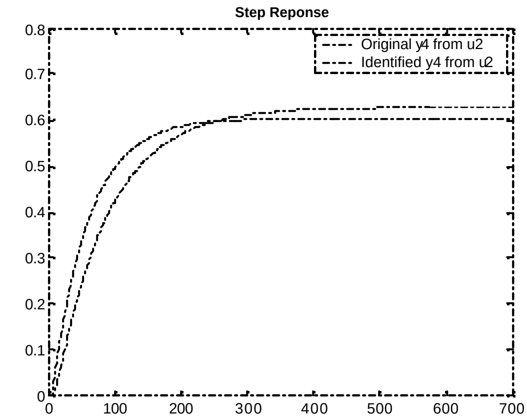

Figure 3.110 : Case 2- Step response of Output-4 from Input-2 (OE)

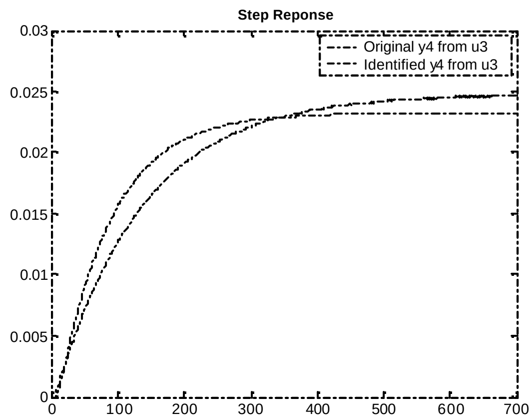

Figure 3.111 : Case 2- Step response of Output-4 from Input-3 (OE)

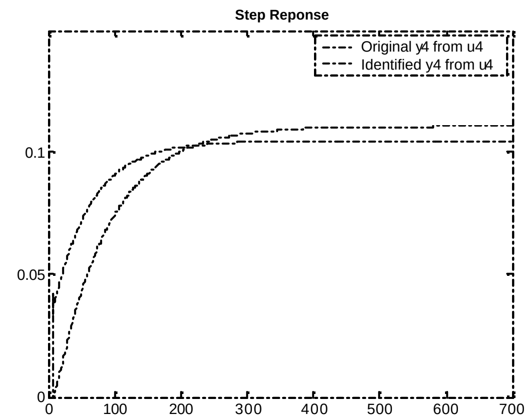

Figure 3.112 : Case 2- Step response of Output-4 from Input-4 (OE)

3.3.2.4 Performance of State space model

The performance of State space modeling scheme in Case 1 was not very good. The closed loop model identified was largely mismatched with the original open loop model. In this case, the disturbance has been reduced and the performance of this scheme is tested on the simulated closed loop data.

Sub space method is used to estimate the unknown parameters of the state space model. The order for the four process outputs are selected as 2, 4, 2 and 4 respectively. In Figs. 3.113 – 3.116, the step responses of the actual open loop model (solid) and the identified state space model (solid) for process output 1 are shown. Compared to case 1, the results have improved somewhat but large inaccuracy remains. The step response from inputs 1, 3 and 4 show large bias and mismatch between the two models. Figs. 3.117 – 3.120 give the step response for the process output 2. The results indicate oscillatory response of the closed loop model and high mismatch with the open loop model. The step responses for process output 3 are shown in Figs. 3.121 – 3.124. The response from input 4 has a different steady state gain sign than the open loop model. Clearly the model is in accurate. The results for process output 4 are shown in Figs. 3.125 – 3.128. Compared to case 1 the results are improved but nevertheless inaccurate. As before, the state space model is unable to identify the open loop process dynamics correctly from the given simulated closed loop data collected from the Demathanizer column running with MPC. The identified model in state space format is given in Eqs. 3.41 - 3.44.

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.98847 & -0.047315\\ 0.0065881 & 0.79971\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.046805 & 0.00472 & -0.0063396 & 0.0096097\\ 0.18525 & -0.0065141 & -0.025554 & 0.036218\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.035466\\ -0.039401\end{bmatrix}e(k)$$

$$y_1(k)=\begin{bmatrix}2.4275 & 0.52839\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix} \tag{3.41}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.72365 & -0.29564 & -0.0077776 & -0.059383\\ -0.20491 & 0.73882 & -0.02126 & -0.19224\\ 0.06944 & 0.078293 & 0.96406 & 0.26514\\ -0.046124 & 0.052029 & -0.40146 & 0.8077\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\\ x_4(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.37748 & 0.016853 & 0.011662 & 0.074732\\ 0.32165 & 0.032578 & -0.089006 & 0.015449\\ -0.12594 & -0.06271 & -0.089006 & 0.060477\\ -0.070661 & -0.11562 & 0.030935 & 0.040228\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}1.353\\ -1.3258\\ -0.34394\\ -0.67003\end{bmatrix}e(k)$$

$$y_2(k)=\begin{bmatrix}0.45911 & -0.026338 & -0.0021416 & 0.00072995\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\\ x_4(k)\end{bmatrix} \tag{3.42}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.98969 & -0.022804\\ 0.027006 & 0.84625\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.0041921 & 0.0028503 & -0.0016139 & 0.0012099\\ 0.0067762 & -0.016666 & -0.0095591 & 0.0081103\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.35291\\ -0.1423\end{bmatrix}e(k)$$

$$y_3(k)=\begin{bmatrix}1.92 & 0.0040989\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix} \tag{3.43}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix} 0.93645 & -0.22429 & -0.11084 & 0.014209 \\ -0.019805 & -0.28123 & 0.73249 & -0.023358 \\ -0.020421 & -0.010395 & 0.097749 & -0.3132 \\ 0.027503 & 0.25533 & 0.015148 & 0.7948 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}$$

$$+\begin{bmatrix} 0.054329 & 0.00014505 & 0.031244 & 0.042559 \\ -0.1684 & 0.20209 & 0.093518 & -0.34534 \\ 0.33938 & -0.12668 & 0.061327 & 0.39369 \\ -0.0096254 & -0.008164 & -0.025039 & -0.0059887 \end{bmatrix}\begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \end{bmatrix}+\begin{bmatrix} 0.45798 \\ -0.25092 \\ -0.19456 \\ -0.33826 \end{bmatrix}e(k)$$

$$y_4(k)=\begin{bmatrix} 1.5201 & 0.030776 & -0.09924 & 0.0079822 \end{bmatrix}\begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \tag{3.44}$$

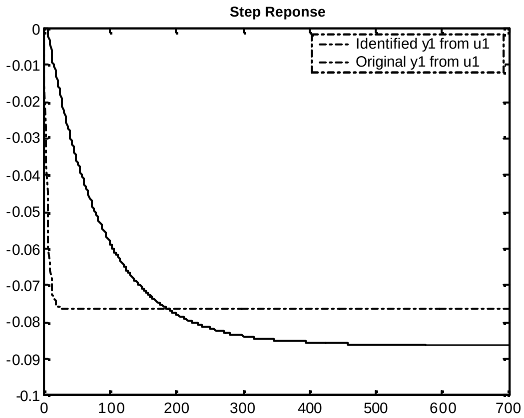

Figure 3.113 : Case 2- Step response of Output-1 from Input-1 (State space)

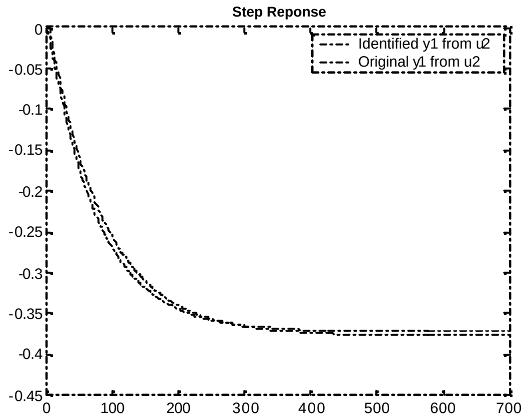

Figure 3.114 : Case 2- Step response of Output-1 from Input-2 (State space)

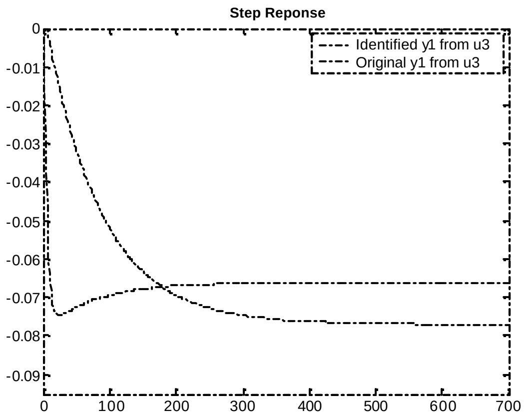

Figure 3.115 : Case 2- Step response of Output-1 from Input-3 (State space)

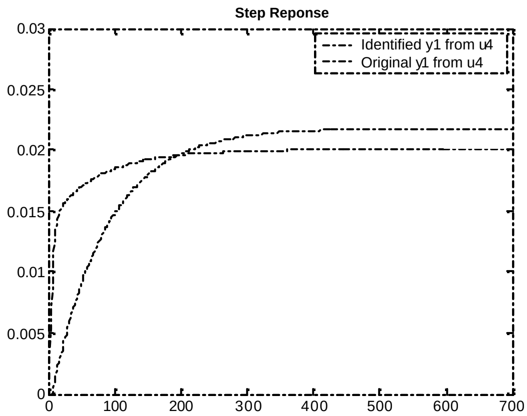

Figure 3.116 : Case 2- Step response of Output-1 from Input-4 (State space)

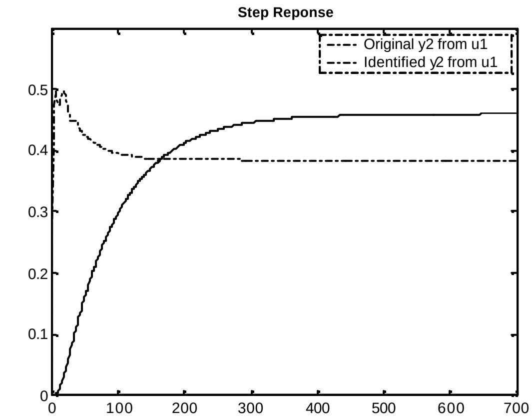

Figure 3.117 : Case 2- Step response of Output-2 from Input-1 (State space)

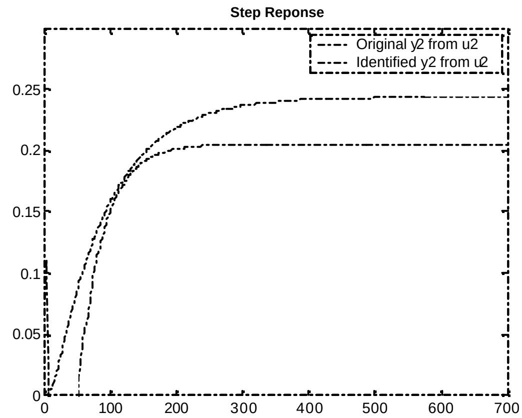

Figure 3.118 : Case 2- Step response of Output-2 from Input-2 (State space)

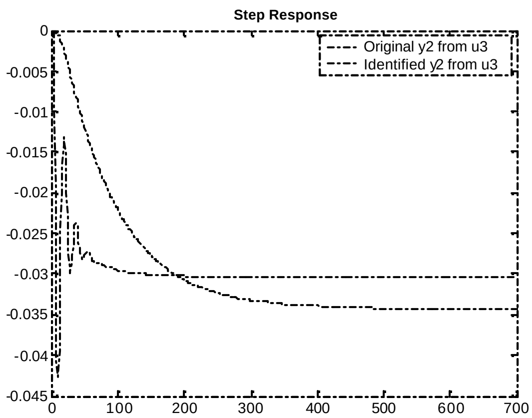

Figure 3.119 : Case 2- Step response of Output-2 from Input-3 (State space)

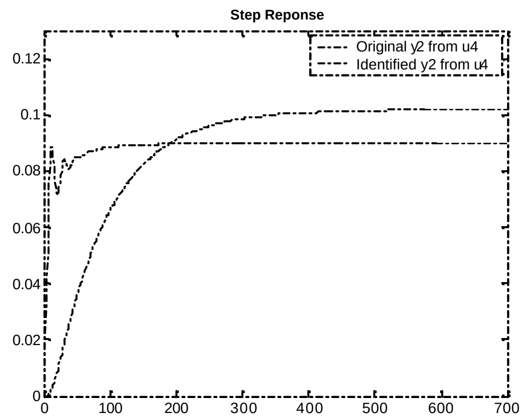

Figure 3.120 : Case 2- Step response of Output-2 from Input-4 (State space)

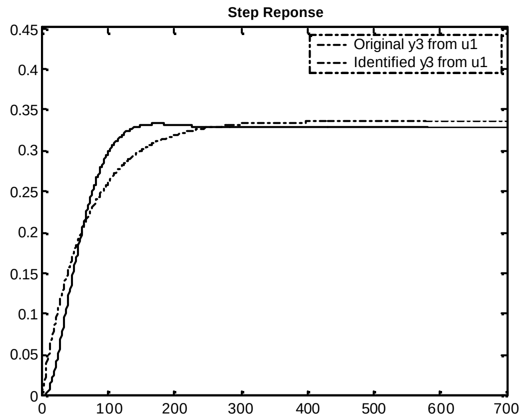

Figure 3.121 : Case 2- Step response of Output-3 from Input-1 (State space)

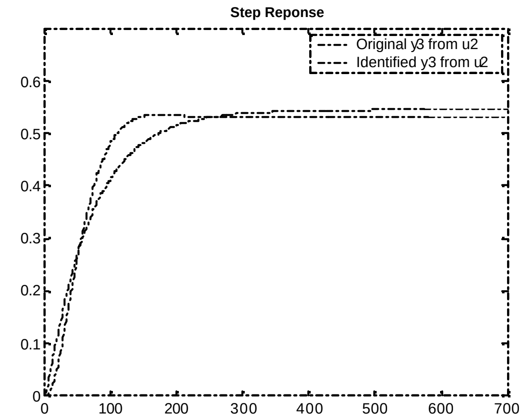

Figure 3.122 : Case 2- Step response of Output-3 from Input-2 (State space)

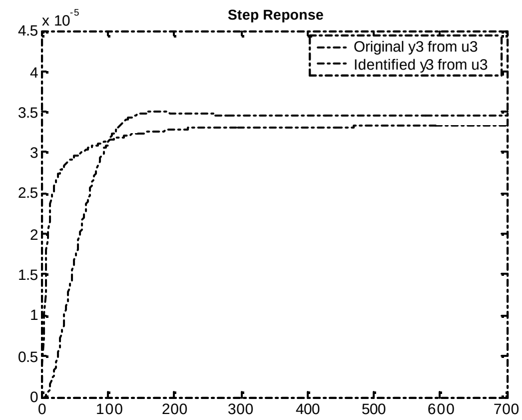

Figure 3.123 : Case 2- Step response of Output-3 from Input-3 (State space)

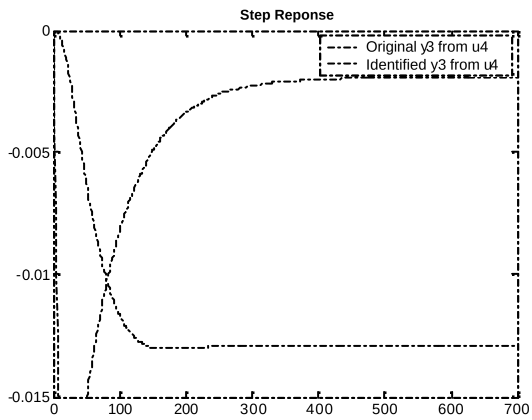

Figure 3.124 : Case 2- Step response of Output-3 from Input-4 (State space)

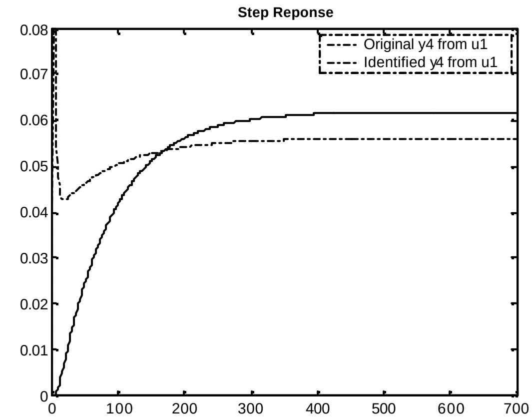

Figure 3.125 : Case 2- Step response of Output-4 from Input-1 (State space)

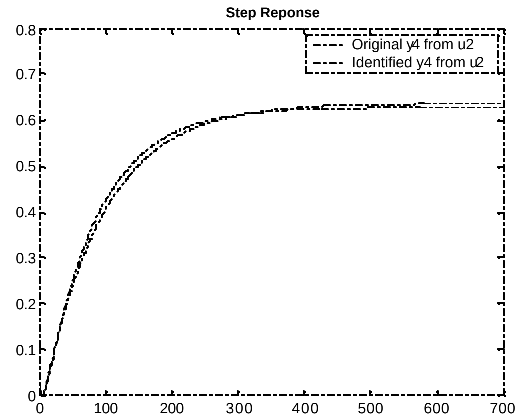

Figure 3.126 : Case 2- Step response of Output-4 from Input-2 (State space)

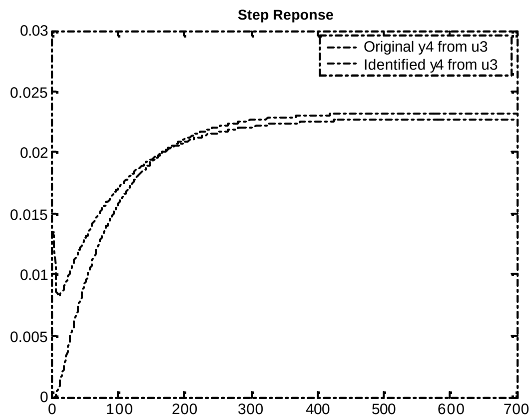

Figure 3.127 : Case 2- Step response of Output-4 from Input-3 (State space)

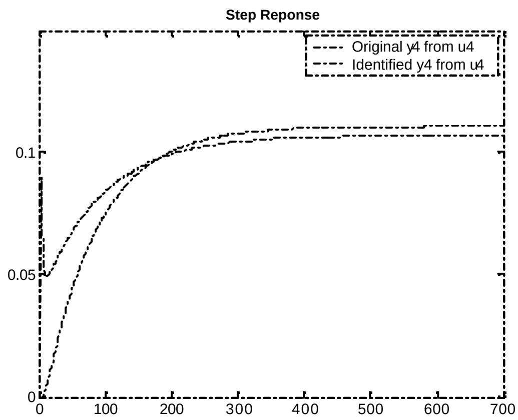

Figure 3.128 : Case 2- Step response of Output-4 from Input-4 (State space)

3.3.3 Case 3 – Closed loop Identification

The first two cases for which closed loop identification is performed are somewhat ideal, in a sense that no mismatch between the actual plant and model is considered. In this last set of simulations (on simulated MPC closed loop data), performance of ARX, ARMAX, OE and state space modeling schemes is analyzed individually on closed loop data from MPC process subject to high plant-model mismatch (see Sec. 2.5.6).

Before presenting the results, a comment is in order here. Output error and state space modeling schemes gave inaccurate estimation when applied directly to closed loop MPC data. On the other hand, least squares (ARX) method and prediction error (ARMAX) method work fine and give consistent estimates of the open loop system. This is later discussed at the end of the chapter. However, to benchmark the performance of all these models all four schemes are used for this case as well. Recall that the disturbance present in this data is nominal.

3.3.3.1 Performance of ARX model

ARX modeling scheme is now tested on simulated closed loop data obtained from a MPC scheme with plant model mismatch. Least squares method is sued to estimate the unknown parameters of the model. The orders (n_a) of the process outputs are selected as 7, 10, 2, and 7, respectively. Figs. 3.129 – 3.132 show the step responses for the closed loop ARX model (dashed) and actual plant model (solid) for process output 1. The results are very good and the steady state part is captured accurately. Figs. 3.133 – 3.136 show the step responses of the process output 2. The results show no bias and good steady state gain fit achieved. For process output 3, the step responses are shown in Figs. 3.137 –

3.140. There is non-trivial bias and very little mismatch between the open loop and closed loop identified models. Similar results are achieved for process output 4 whose step responses are shown in Figs. 3.141 – 3.144. Overall, the ARX modeling scheme again gives an accurate description of the process dynamics with trivial mismatch. The identified model is given in Eqs. 3.45 - 3.48, which show that in the presence of high plant-model mismatch, a high order ARX will generally give consistent and unbiased results.

$$
\begin{aligned}
y1(t) = {} & \frac{-0.0013}{1-1.145q^{-1}+0.02547q^{-2}+0.1479q^{-3}+0.171q^{-4}+0.07763q^{-5}} \cdots \\
& \cdots \frac{}{-0.088q^{-6}-0.1738q^{-7}} q^{-6} u_1(t) \\
& + \frac{-0.0058}{1-1.145q^{-1}+0.02547q^{-2}+0.1479q^{-3}+0.171q^{-4}+0.07763q^{-5}} \cdots \\
& \cdots \frac{}{-0.088q^{-6}-0.1738q^{-7}} q^{-6} u_2(t) \\
& + \frac{-0.0012}{1-1.145q^{-1}+0.02547q^{-2}+0.1479q^{-3}+0.171q^{-4}+0.07763q^{-5}} \cdots \\
& \cdots \frac{}{-0.088q^{-6}-0.1738q^{-7}} q^{-6} u_3(t) \\
& + \frac{0.0003335}{1-1.145q^{-1}+0.02547q^{-2}+0.1479q^{-3}+0.171q^{-4}+0.07763q^{-5}} \\
& \cdots \frac{}{-0.088q^{-6}-0.1738q^{-7}} q^{-6} u_4(t) \\
& + \frac{1}{1-1.145q^{-1}+0.02547q^{-2}+0.1479q^{-3}+0.171q^{-4}+0.07763q^{-5}} \\
& \cdots \frac{}{-0.088q^{-6}-0.1738q^{-7}} e(t)
\end{aligned}
\tag{3.45}
$$

$$y2(t) = \frac{-0.0001 + 0.0062q^{-1} + 0.0044q^{-2} - 0.0040q^{-3}}{1 - 1.261q^{-1} + 0.2206q^{-2} + 0.245q^{-3} - 0.08079q^{-4} - 0.1169q^{-5}} \cdots$$

$$\cdots \frac{}{+0.05817q^{-6} + 0.08545q^{-7} - 0.03464q^{-8} - 0.0521q^{-9} - 0.2453q^{-10}} q^{-10} u_1(t)$$

$$+ \frac{-0.0979 + 0.0854q^{-1} + 0.1273q^{-2} - 0.1113q^{-3}}{1 - 1.261q^{-1} + 0.2206q^{-2} + 0.245q^{-3} - 0.08079q^{-4} - 0.1169q^{-5}} \cdots$$

$$\cdots \frac{}{+0.05817q^{-6} + 0.08545q^{-7} - 0.03464q^{-8} - 0.0521q^{-9} - 0.2453q^{-10}} q^{-10} u_2(t)$$

$$+ \frac{-0.4114 - 0.4221q^{-1} + 0.2857q^{-2} + 0.0579q^{-3}}{1 - 1.261q^{-1} + 0.2206q^{-2} + 0.245q^{-3} - 0.08079q^{-4} - 0.1169q^{-5}} \cdots$$

$$\cdots \frac{}{+0.05817q^{-6} + 0.08545q^{-7} - 0.03464q^{-8} - 0.0521q^{-9} - 0.2453q^{-10}} q^{-10} u_3(t)$$

$$+ \frac{-0.0216 + 0.0144q^{-1} + 0.0252q^{-2} - 0.0165q^{-3}}{1 - 1.261q^{-1} + 0.2206q^{-2} + 0.245q^{-3} - 0.08079q^{-4} - 0.1169q^{-5}} \cdots$$

$$\cdots \frac{}{+0.05817q^{-6} + 0.08545q^{-7} - 0.03464q^{-8} - 0.0521q^{-9} - 0.2453q^{-10}} q^{-10} u_4(t)$$

$$+ \frac{1}{1 - 1.261q^{-1} + 0.2206q^{-2} + 0.245q^{-3} - 0.08079q^{-4} - 0.1169q^{-5}} \cdots$$

$$\cdots \frac{}{+0.05817q^{-6} + 0.08545q^{-7} - 0.03464q^{-8} - 0.0521q^{-9} - 0.2453q^{-10}} e(t) \tag{3.46}$$

$$y3(t) = \frac{0.0019 + 0.0043q^{-1}}{1 - 1.072q^{-1} + 0.09089q^{-2}} q^{-2} u_1(t)$$

$$+ \frac{0.0015 - 0.0017q^{-1}}{1 - 1.072q^{-1} + 0.09089q^{-2}} q^{-2} u_2(t)$$

$$+ \frac{0.0000006617 - 0.0000000185q^{-1}}{1 - 1.072q^{-1} + 0.09089q^{-2}} q^{-2} u_3(t)$$

$$+ \frac{0.0014 - 0.0016q^{-1}}{1 - 1.072q^{-1} + 0.09089q^{-2}} q^{-2} u_4(t)$$

$$+ \frac{1}{1 - 1.072q^{-1} + 0.09089q^{-2}} e(t) \tag{3.47}$$

$$y4(t) = \frac{0.0007248}{1-1.102q^{-1} - 0.01237q^{-2} + 0.1264q^{-3} + 0.1009q^{-4} + 0.0671q^{-5}} \cdots$$

$$\cdots \frac{}{-0.08977q^{-6} - 0.07907q^{-7}} q^{-2}\, u_1(t)$$

$$+ \frac{0.0074}{1-1.102q^{-1} - 0.01237q^{-2} + 0.1264q^{-3} + 0.1009q^{-4} + 0.0671q^{-5}} \cdots$$

$$\cdots \frac{}{-0.08977q^{-6} - 0.07907q^{-7}} q^{-2}\, u_2(t)$$

$$+ \frac{0.0002727}{1-1.102q^{-1} - 0.01237q^{-2} + 0.1264q^{-3} + 0.1009q^{-4} + 0.0671q^{-5}} \cdots$$

$$\cdots \frac{}{-0.08977q^{-6} - 0.07907q^{-7}} q^{-2}\, u_3(t)$$

$$+ \frac{0.0013}{1-1.102q^{-1} - 0.01237q^{-2} + 0.1264q^{-3} + 0.1009q^{-4} + 0.0671q^{-5}} \cdots$$

$$\cdots \frac{}{-0.08977q^{-6} - 0.07907q^{-7}} q^{-2}\, u_4(t)$$

$$+ \frac{1}{1-1.102q^{-1} - 0.01237q^{-2} + 0.1264q^{-3} + 0.1009q^{-4} + 0.0671q^{-5}} \cdots$$

$$\cdots \frac{}{-0.08977q^{-6} - 0.07907q^{-7}} e(t) \qquad (3.48)$$

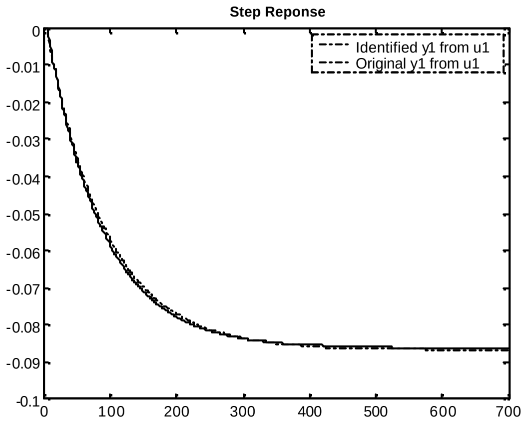

Figure 3.129 : Case 3- Step response of Output-1 from Input-1 (ARX)

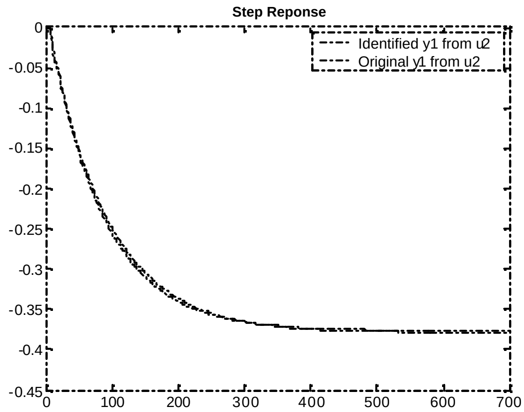

Figure 3.130 : Case 3- Step response of Output-1 from Input-2 (ARX)

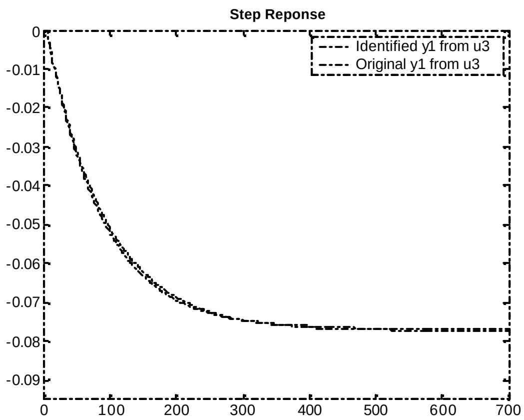

Figure 3.131 : Case 3- Step response of Output-1 from Input-3 (ARX)

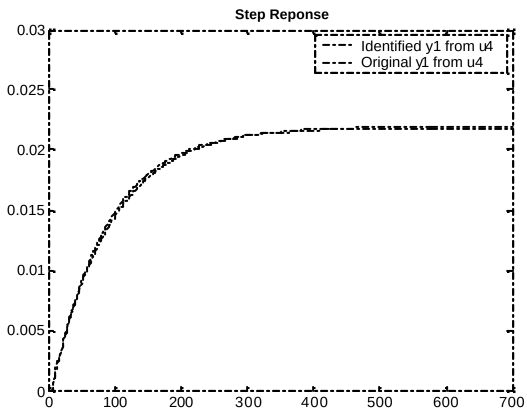

Figure 3.132 : Case 3- Step response of Output-1 from Input-4 (ARX)

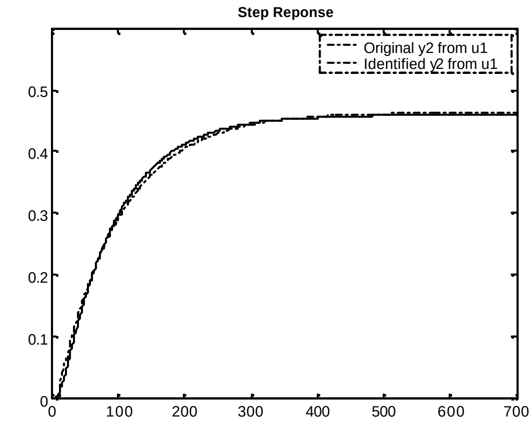

Figure 3.133 : Case 3- Step response of Output-2 from Input-1 (ARX)

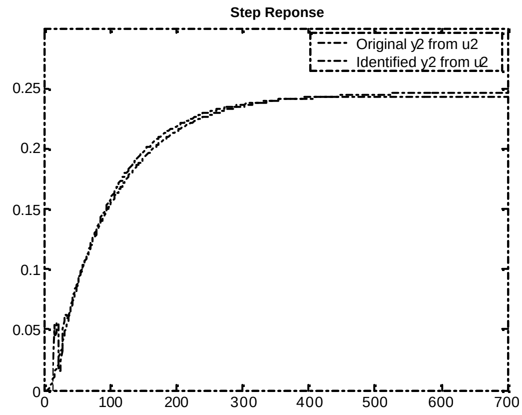

Figure 3.134 : Case 3- Step response of Output-2 from Input-1 (ARX)

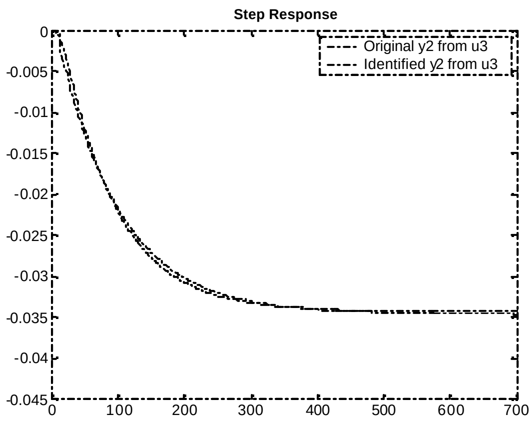

Figure 3.135 : Case 3- Step response of Output-2 from Input-3 (ARX)

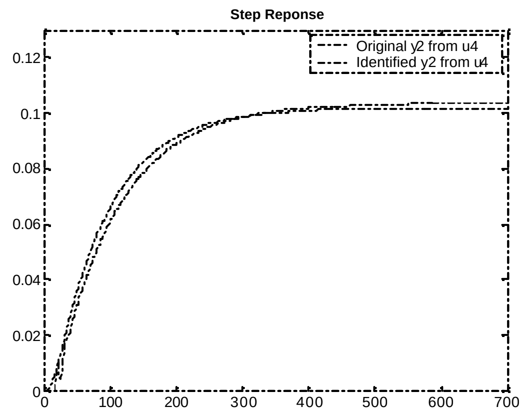

Figure 3.136 : Case 3- Step response of Output-2 from Input-4 (ARX)

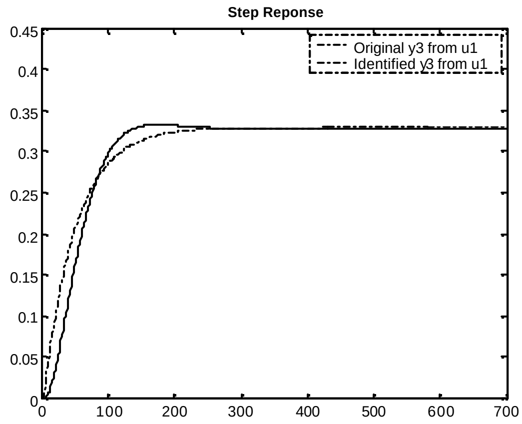

Figure 3.137 : Case 3- Step response of Output-3 from Input-1 (ARX)

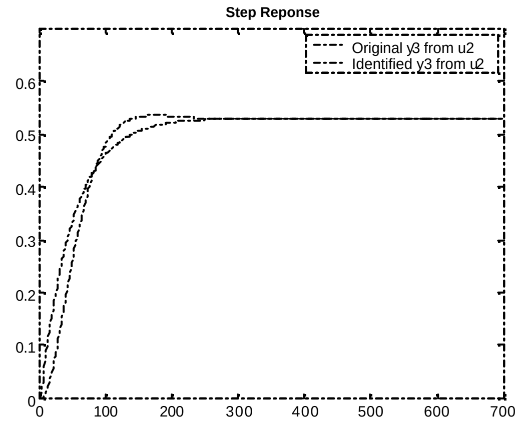

Figure 3.138 : Case 3- Step response of Output-3 from Input-2 (ARX)

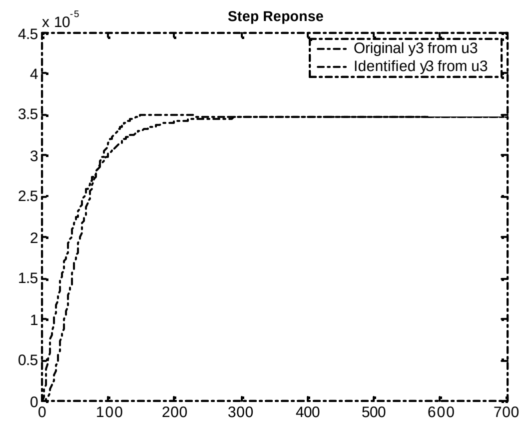

Figure 3.139 : Case 3- Step response of Output-3 from Input-3 (ARX)

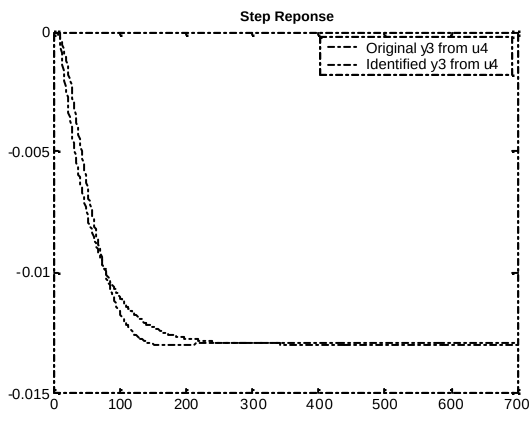

Figure 3.140 : Case 3- Step response of Output-3 from Input-4 (ARX)

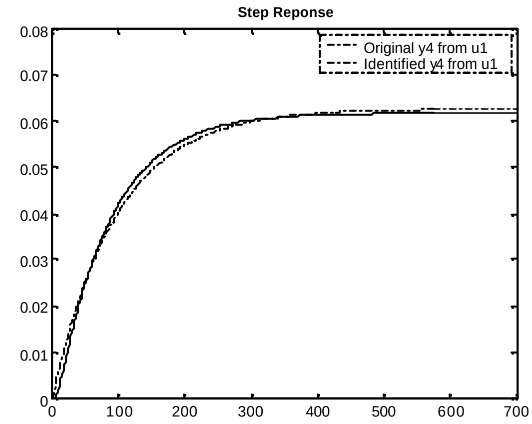

Figure 3.141 : Case 3- Step response of Output-4 from Input-1 (ARX)

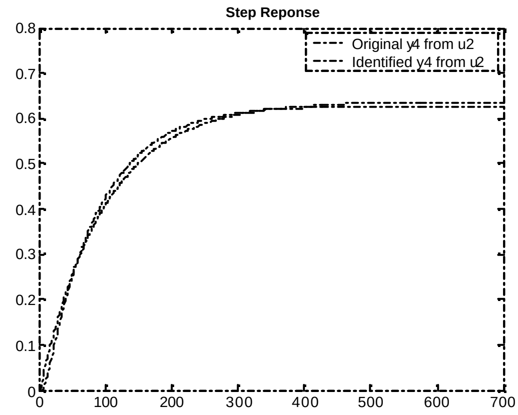

Figure 3.142 : Case 3- Step response of Output-4 from Input-2 (ARX)

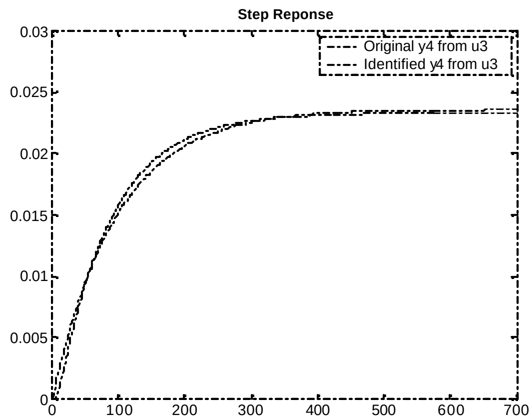

Figure 3.143 : Case 3- Step response of Output-4 from Input-3 (ARX)

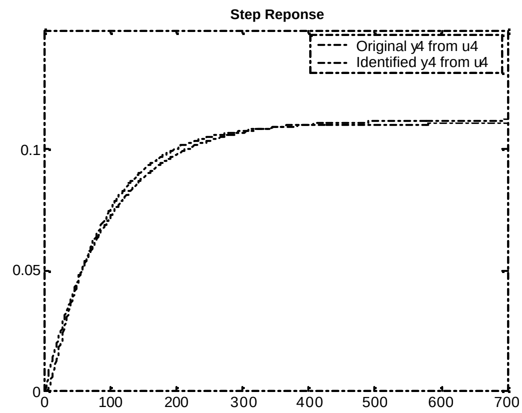

Figure 3.144 : Case 3- Step response of Output-4 from Input-4 (ARX)

3.3.3.2 Performance of ARMAX model

ARMAX modeling scheme have also shown excellent performance in the two previous cases. Now it is tested for its consistency and performance in this case of plant-model mismatch MPC scheme.

Prediction error method is used to estimate the unknown parameters of the ARMAX model. The orders (n_a) of the process outputs are selected as 7, 8, 4 and 7 respectively. Figs. 3.145 – 3.148 shows the step responses of the identified closed loop process model (dashed) with that of the actual open loop process model (solid) for output 1. The results indicate a perfect match of the two models. There is no bias and the steady state part has been captured accurately. Figs. 3.149 -3.152 shows the step responses of the closed loop identified ARMAX model with that of the open loop model for process output 2. Again there is no mismatch at all between the two models. Figs. 3.153 – 3.156 shows the step response of the closed loop identified ARMAX model with that of the open loop process model for output 3. Unlike ARX modeling scheme, ARMAX model does not have a perfect fit at the steady state part, but the slight mismatch is insignificant. Similarly, the step responses for output 4 are shown in Figs. 3.157 – 3.160. The results again demonstrate that closed loop ARMAX model gives accurate fit and manages to capture the steady state part accurately. The identified model is given in Eqs. 3.49 - 3.52.

$$y1(t) = \frac{-0.0013}{1-1.2q^{-1}+0.09449q^{-2}+\ 0.1566q^{-3}+\ 0.1443q^{-4}+0.07684q^{-5}}\cdots$$

$$\cdots\frac{}{-0.08908q^{-6}-0.168q^{-7}}\,q^{-6}\,u_1(t)$$

$$+\ \frac{-0.0059}{1-1.2q^{-1}+0.09449q^{-2}+\ 0.1566q^{-3}+\ 0.1443q^{-4}+0.07684q^{-5}}\cdots$$

$$\cdots\frac{}{-0.08908q^{-6}-0.168q^{-7}}\,q^{-6}\,u_2(t)$$

$$+\ \frac{-0.0012}{1-1.2q^{-1}+0.09449q^{-2}+\ 0.1566q^{-3}+\ 0.1443q^{-4}+0.07684q^{-5}}\cdots$$

$$\cdots\frac{}{-0.08908q^{-6}-0.168q^{-7}}\,q^{-6}\,u_3(t)$$

$$+\frac{0.3391}{1-1.2q^{-1}+0.09449q^{-2}+\ 0.1566q^{-3}+\ 0.1443q^{-4}+0.07684q^{-5}}\cdots$$

$$\cdots\frac{}{-0.08908q^{-6}-0.168q^{-7}}\,q^{-6}\,u_4(t)$$

$$+\frac{1-0.06754\,q^{-1}}{1-1.2q^{-1}+0.09449\,q^{-2}+\ 0.1566\,q^{-3}+\ 0.1443\,q^{-4}+0.07684\,q^{-5}}\cdots \tag{3.49}$$

$$\cdots\frac{}{-0.08908\,q^{-6}-0.168\,q^{-7}}\,e(t)$$

$$y1(t)=\frac{0.0023}{1-2.017q^{-1}+1.23q^{-2}+0.008153q^{-3}-0.2353q^{-4}-0.02559q^{-5}+0.077q^{-6}+0.05969q^{-7}-0.09236q^{-8}}q^{-9}u_1(t)$$

$$+\frac{0.0021}{1-2.017q^{-1}+1.23q^{-2}+0.008153q^{-3}-0.2353q^{-4}-0.02559q^{-5}+0.077q^{-6}+0.05969q^{-7}-0.09236q^{-8}}q^{-9}u_2(t)$$

$$+\frac{-0.1682}{1-2.017q^{-1}+1.23q^{-2}+0.008153q^{-3}-0.2353q^{-4}-0.02559q^{-5}+0.077q^{-6}+0.05969q^{-7}-0.09236q^{-8}}q^{-9}u_3(t)$$

$$+\frac{0.4991}{1-2.017q^{-1}+1.23q^{-2}+0.008153q^{-3}-0.2353q^{-4}-0.02559q^{-5}+0.077q^{-6}+0.05969q^{-7}-0.09236q^{-8}}q^{-9}u_4(t)$$

$$+\frac{1-0.6268q^{-1}}{1-2.017q^{-1}+1.23q^{-2}+0.008153q^{-3}-0.2353q^{-4}-0.02559q^{-5}+0.077q^{-6}+0.05969q^{-7}-0.09236q^{-8}}e(t) \tag{3.50}$$

$$y3(t)=\frac{0.0026+0.0019q^{-1}}{1-1.312q^{-1}+0.4404q^{-2}+0.4404q^{-2}-0.1215q^{-3}+0.006652q^{-4}}q^{-2}u_1(t)$$

$$+\frac{0.0124-0.0052q^{-1}}{1-1.312q^{-1}+0.4404q^{-2}+0.4404q^{-2}-0.1215q^{-3}+0.006652q^{-4}}q^{-2}u_2(t)$$

$$+\frac{0.0000009512-0.0000004838q^{-1}}{1-1.312q^{-1}+0.4404q^{-2}+0.4404q^{-2}-0.1215q^{-3}+0.006652q^{-4}}q^{-2}u_3(t)$$

$$+\frac{0.0014-0.0015q^{-1}}{1-1.312q^{-1}+0.4404q^{-2}+0.4404q^{-2}-0.1215q^{-3}+0.006652q^{-4}}q^{-2}u_4(t)$$

$$+\frac{1-0.2754q^{-1}+0.2349q^{-2}}{1-1.312q^{-1}+0.4404q^{-2}+0.4404q^{-2}-0.1215q^{-3}+0.006652q^{-4}}e(t) \tag{3.51}$$

$$y4(t) = \frac{0.00006517}{1-3.325q^{-1}+4.182q^{-2}-2.158q^{-3}+0.2072q^{-4}-0.09961q^{-5}}\cdots$$

$$\cdots\frac{}{-0.3442q^{-6}-0.1498q^{-7}}q^{-2}u_1(t)$$

$$+\frac{0.0006623}{1-3.325q^{-1}+4.182q^{-2}-2.158q^{-3}+0.2072q^{-4}-0.09961q^{-5}}\cdots$$

$$\cdots\frac{}{-0.3442q^{-6}-0.1498q^{-7}}q^{-2}u_2(t)$$

$$+\frac{0.00002452}{1-3.325q^{-1}+4.182q^{-2}-2.158q^{-3}+0.2072q^{-4}-0.09961q^{-5}}\cdots$$

$$\cdots\frac{}{-0.3442q^{-6}-0.1498q^{-7}}q^{-2}u_3(t)$$

$$+\frac{0.0001169}{1-3.325q^{-1}+4.182q^{-2}-2.158q^{-3}+0.2072q^{-4}-0.09961q^{-5}}\cdots$$

$$\cdots\frac{}{-0.3442q^{-6}-0.1498q^{-7}}q^{-2}u_4(t)$$

$$+\frac{1-2.472q^{-1}+2.192q^{-2}-0.679q^{-2}}{1-3.325q^{-1}+4.182q^{-2}-2.158q^{-3}+0.2072q^{-4}-0.09961q^{-5}}\cdots$$

$$\cdots\frac{}{-0.3442q^{-6}-0.1498q^{-7}}e(t) \tag{3.52}$$

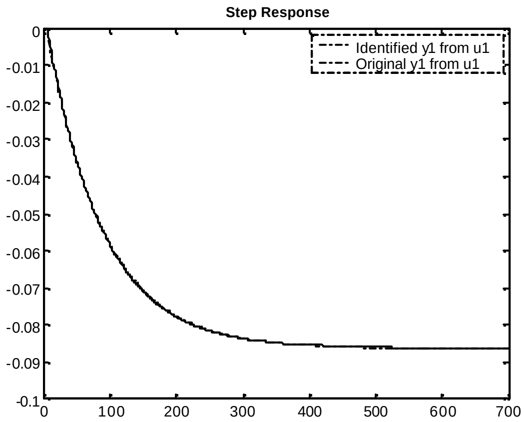

Figure 3.145 : Case 3- Step response of Output-1 from Input-1 (ARMAX)

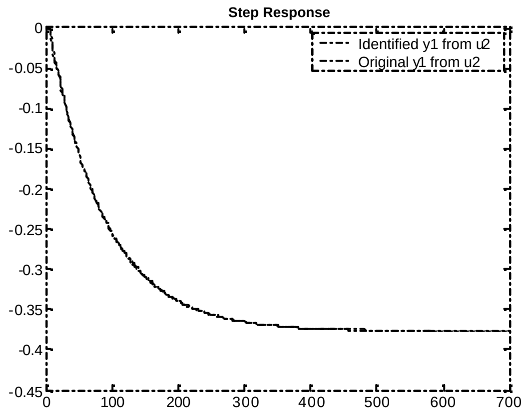

Figure 3.146 : Case 3- Step response of Output-1 from Input-2 (ARMAX)

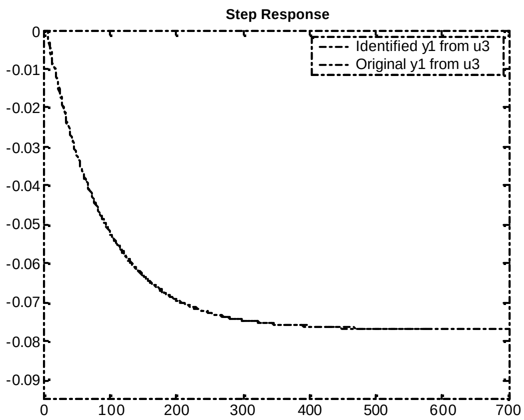

Figure 3.147 : Case 3- Step response of Output-1 from Input-3 (ARMAX)

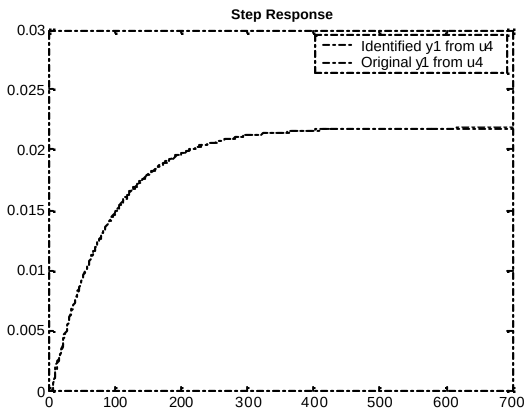

Figure 3.148 : Case 3- Step response of Output-1 from Input-4 (ARMAX)

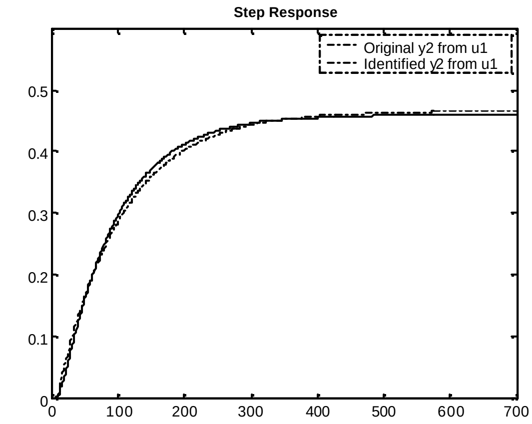

Figure 3.149 : Case 3- Step response of Output-2 from Input-1 (ARMAX)

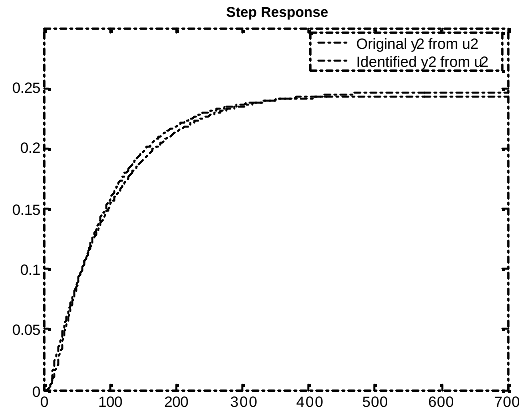

Figure 3.150 : Case 3- Step response of Output-2 from Input-2 (ARMAX)

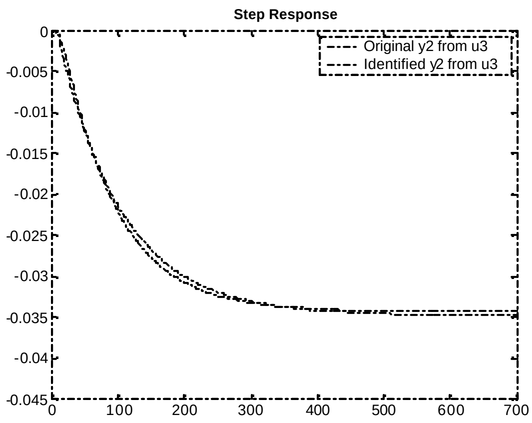

Figure 3.151 : Case 3- Step response of Output-2 from Input-3 (ARMAX)

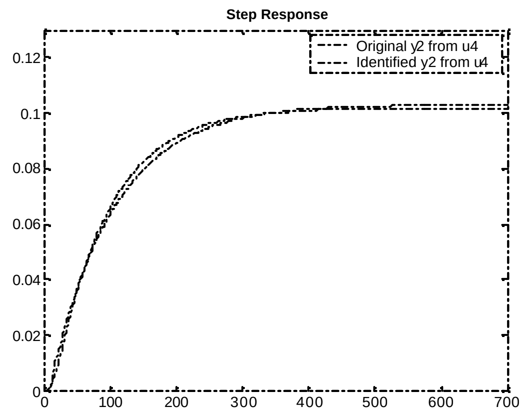

Figure 3.152 : Case 3- Step response of Output-2 from Input-4 (ARMAX)

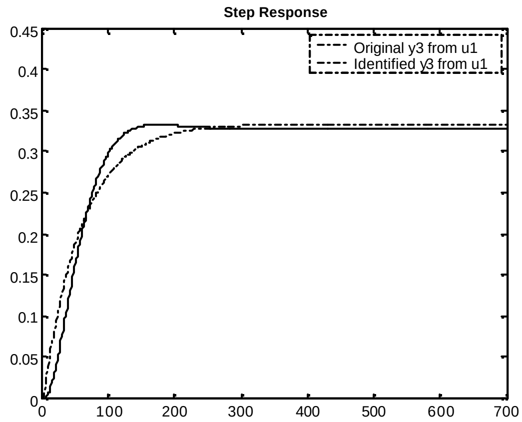

Figure 3.153 : Case 3- Step response of Output-3 from Input-1 (ARMAX)

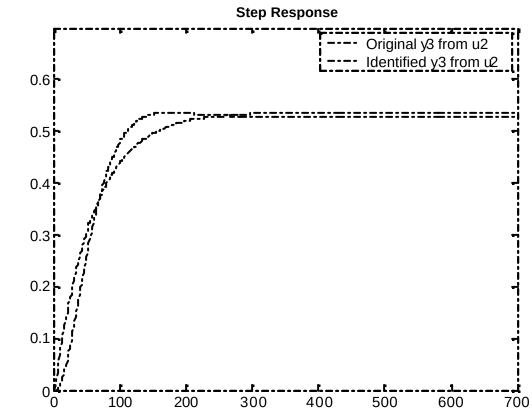

Figure 3.154 : Case 3- Step response of Output-3 from Input-2 (ARMAX)

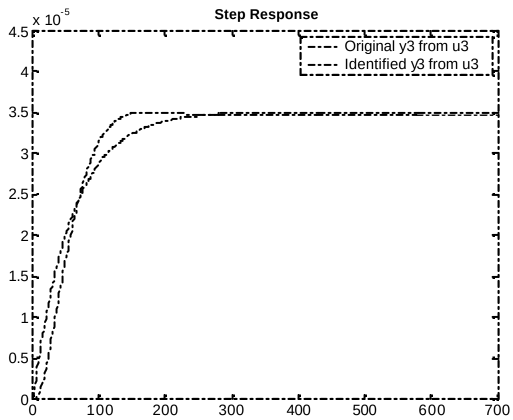

Figure 3.155 : Case 3- Step response of Output-3 from Input-3 (ARMAX)

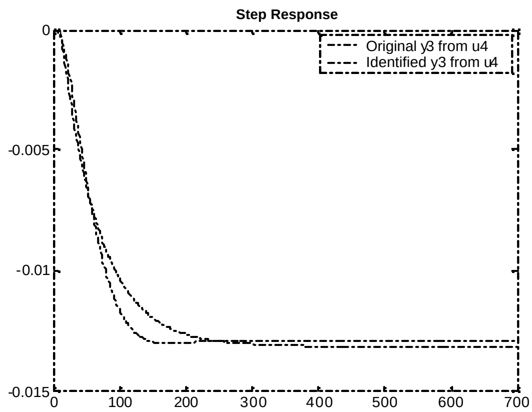

Figure 3.156 : Case 3- Step response of Output-3 from Input-4 (ARMAX)

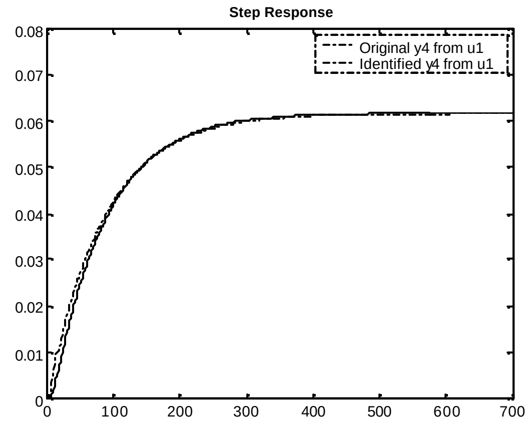

Figure 3.157 : Case 3- Step response of Output-4 from Input-1 (ARMAX)

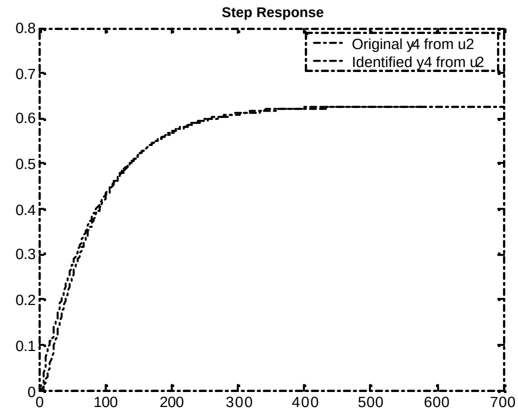

Figure 3.158 : Case 3- Step response of Output-4 from Input-2 (ARMAX)

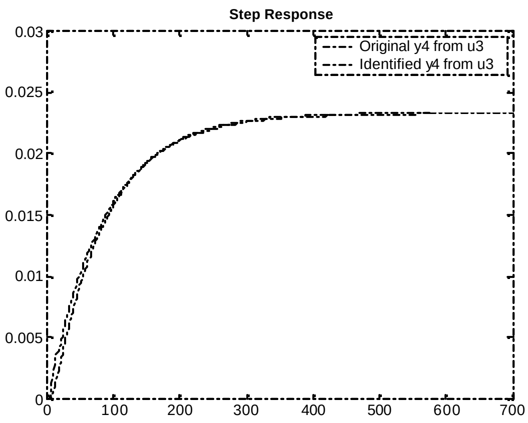

Figure 3.159 : Case 3- Step response of Output-4 from Input-3 (ARMAX)

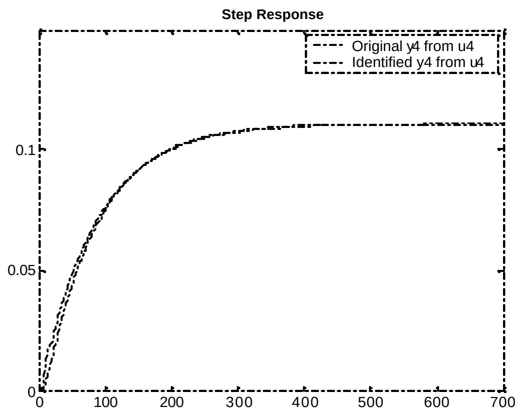

Figure 3.160 : Case 3- Step response of Output-4 from Input-4 (ARMAX)

3.3.3.3 Performance of OE model

In case 1 and 2, it was seen that this technique did not give good estimates of the open loop process model from closed loop simulated data. The identification capability of this modeling scheme is now tested for this case where the MPC scheme had a mismatch between the plant and the model

Prediction error method is used to estimate the parameters of this modeling scheme The orders (n_a) of the process outputs are 2, 6, 3 and 2 respectively.. The step responses of the actual open loop (solid) and identified closed loop OE model (dashed) for process output 1 are shown in Figs. 3.161 - 3.1164. The results show large mismatch between the two models. The closed loop OE model has not captured any of the dynamics of the open loop model. The step responses from all inputs are clearly biased. In Figs. 3.165- 3.168, the step responses of the process output 2 is shown. Again the model is unable to capture the steady state part of the open loop process model. Figs. 3.169 - 3.1172 give the step responses for process output 3. Unlike for process output 1 and 2, there is very little mismatch between the step responses from inputs 1, 3 and 4. However, the step response from input 2 indicates that the model is slightly mismatched. The step responses of the process output 4 are shown in Figs. 3.173 – 3.176. It is obvious that the models are mismatched with large bias. Except for the response from input 2, the rest of the step responses are unable to capture the steady state part accurately. Overall, the closed loop model is largely inaccurate. The identified model is given in Eqs. 3.53 - 3.56.

$$y1(t)=\frac{-0.0084}{1-0.1674q^{-1}-0.7264q^{-2}}q^{-3}\,u_1(t)\ +\ \frac{-0.0006269}{1-1.899q^{-1}+0.901q^{-2}}q^{-3}\,u_2(t)$$

$$+\ \frac{-0.0018}{1-1.805q^{-1}+0.8315q^{-2}}q^{-3}\,u_3(t)\ +\frac{0.0208}{1-1.1953q^{-1}+0.2827q^{-2}}q^{-1}\,u_4(t) \tag{3.53}$$

$$y2(t)=\frac{0.3059-\ 0.5873q^{-1}+\ 0.4882q^{-2}-\ 0.2054q^{-3}}{1-2.378q^{-1}+2.576q^{-2}-1.726q^{-3}+0.6613q^{-4}-0.1557q^{-5}}$$

$$\cdots\frac{}{+0.02662q^{-6}}q^{-2}\,u_1(t)$$

$$+\frac{0.0493+\ 0.0147q^{-1}-\ 0.0488q^{-2}-\ 0.0141q^{-3}}{1-0.4884q^{-1}-1.072q^{-2}+0.4654q^{-3}-0.1468q^{-4}+0.02488q^{-5}}\cdots$$

$$\cdots\frac{}{+0.2226q^{-6}}q^{-2}\,u_2(t)$$

$$+\frac{-0.0006-\ 0.0008q^{-1}-\ 0.0024q^{-2}-\ 0.0002q^{-3}}{1-0.6209q^{-1}-0.7523q^{-2}+1.016q^{-3}-0.5647q^{-4}-0.6154q^{-5}}\cdots$$

$$\cdots\frac{}{+0.0002296q^{-6}}q^{-2}\,u_3(t)$$

$$+\frac{0.0008-\ 0.0025q^{-1}+\ 0.0032q^{-2}-\ 0.0015q^{-3}}{1-4.511q^{-1}+9.333q^{-2}-11.32q^{-3}+8.457q^{-4}-3.673q^{-5}}\cdots$$

$$\cdots\frac{}{+0.7175q^{-6}}q^{-2}\,u_4(t) \tag{3.54}$$

$$y3(t) = \frac{0.0060}{1-0.5878q^{-1}-0.9419q^{-2}+0.5479q^{-3}}\, q^{-1}\, u_1(t)$$

$$+\frac{0.0064}{1-1.591q^{-1}+0.5709q^{-2}+0.03236q^{-3}}\, q^{-1}\, u_2(t)$$

$$+\frac{0.000001083}{1-0.4844q^{-1}-0.3062q^{-2}+0.1783q^{-3}}\, q^{-1}\, u_3(t)$$

$$+\frac{-0.0002721}{1-1.1q^{-1}+0.4968\, q^{-2}-0.3759\, q^{-3}}\, q^{-1}\, u_4(t) \tag{3.55}$$

$$y4(t) = \frac{-0.0158+\ 0.0284q^{-1}}{1-1.371q^{-1}+0.6021q^{-2}}\, q^{-5}\, u_1(t)$$

$$+\frac{0.0216-\ 0.0193q^{-1}}{1-1.692q^{-1}+0.696q^{-2}}\, q^{-5}\, u_2(t)$$

$$+\frac{0.0149-\ 0.0147q^{-1}}{1-1.145q^{-1}+0.1542q^{-2}}\, q^{-5}\, u_3(t)$$

$$+\frac{0.0308\ -\ 0.0292\, q^{-1}}{1-0.8743\, q^{-1}-0.1105\, q^{-2}}\, q^{-5}\, u_4(t) \tag{3.56}$$

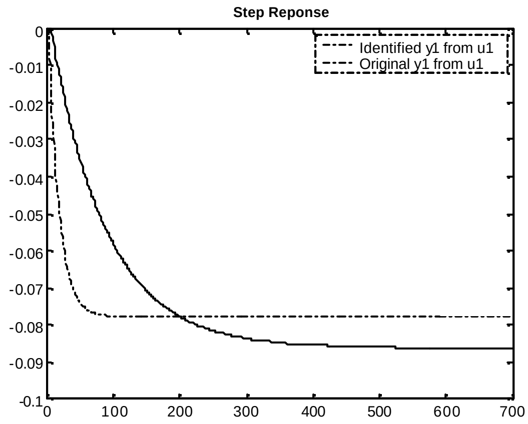

Figure 3.161 : Case 3- Step response of Output-1 from Input-1 (OE)

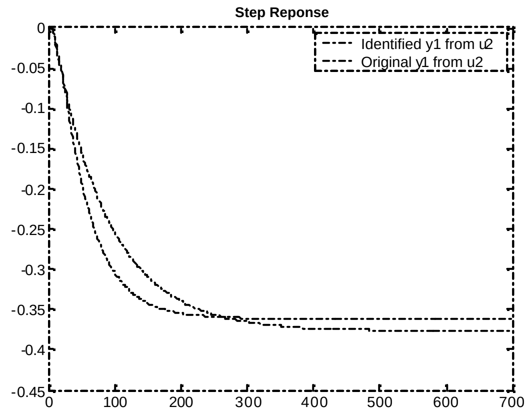

Figure 3.162 : Case 3- Step response of Output-1 from Input-2 (OE)

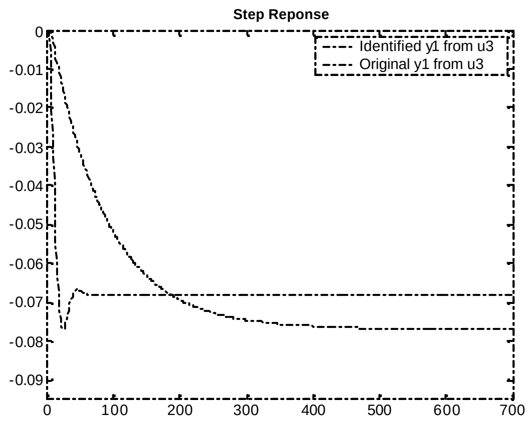

Figure 3.163 : Case 3- Step response of Output-1 from Input-3 (OE)

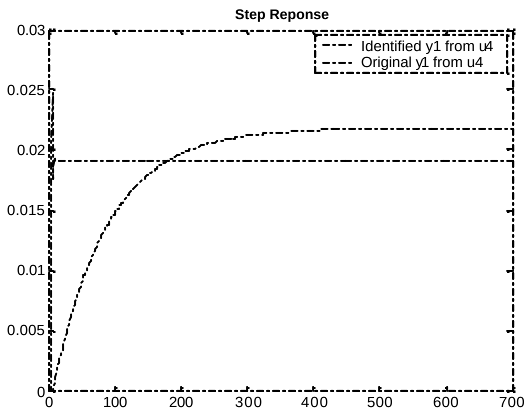

Figure 3.164 : Case 3- Step response of Output-1 from Input-4 (OE)

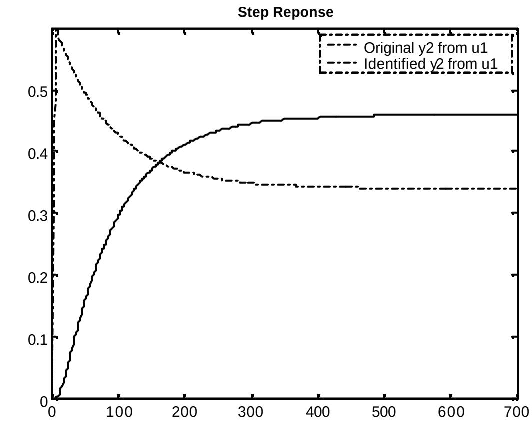

Figure 3.165 : Case 3- Step response of Output-2 from Input-1 (OE)

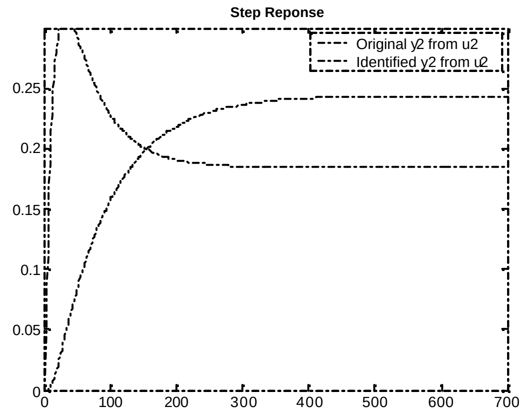

Figure 3.166 : Case 3- Step response of Output-2 from Input-2 (OE)

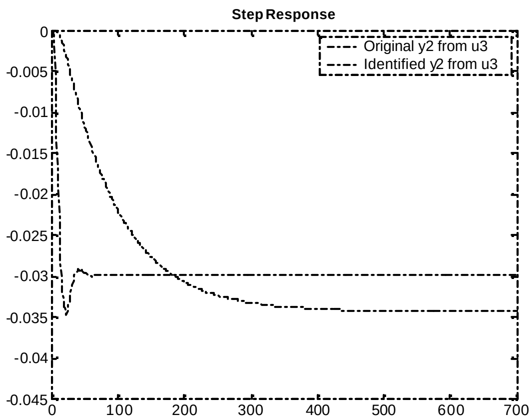

Figure 3.167 : Case 3- Step response of Output-2 from Input-3 (OE)

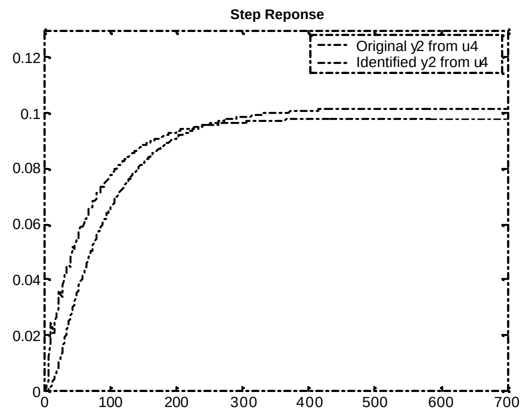

Figure 3.168 : Case 3- Step response of Output-2 from Input-4 (OE)

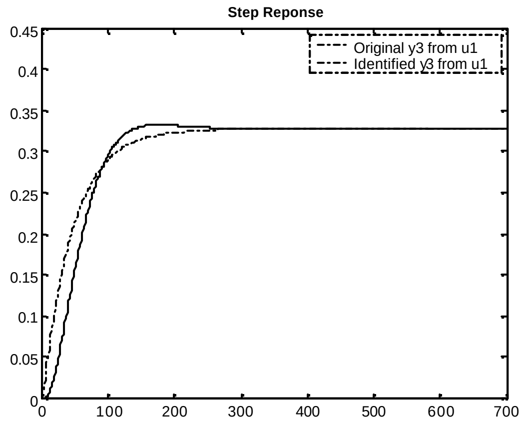

Figure 3.169 : Case 3- Step response of Output-3 from Input-1 (OE)

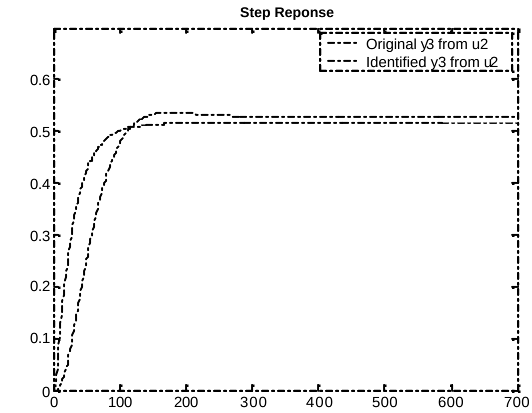

Figure 3.170 : Case 3- Step response of Output-3 from Input-2 (OE)

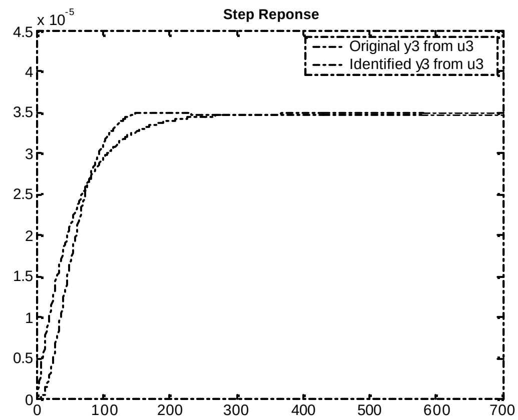

Figure 3.171 : Case 3- Step response of Output-3 from Input-3 (OE)

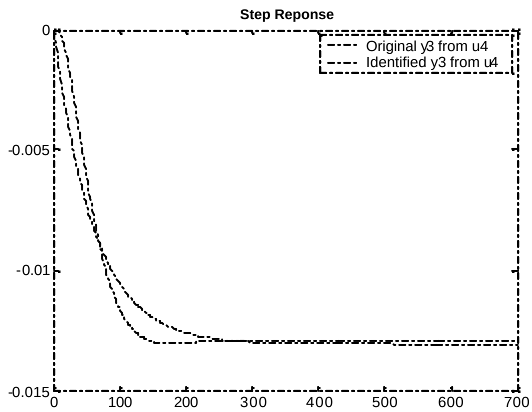

Figure 3.172 : Case 3- Step response of Output-3 from Input-4 (OE)

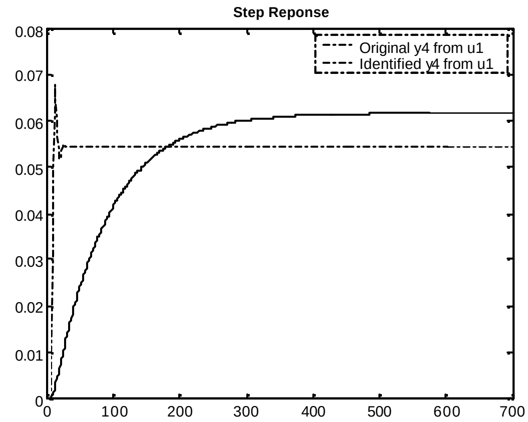

Figure 3.173 : Case 3- Step response of Output-4 from Input-1 (OE)

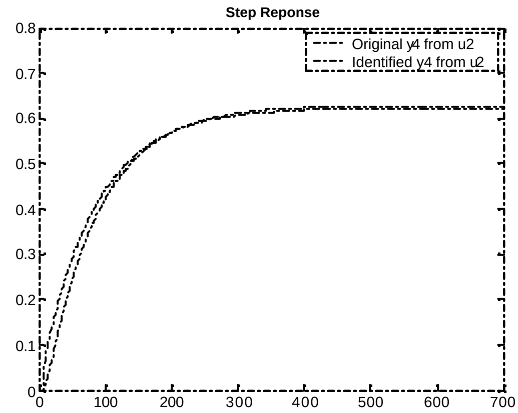

Figure 3.174 : Case 3- Step response of Output-4 from Input-2 (OE)

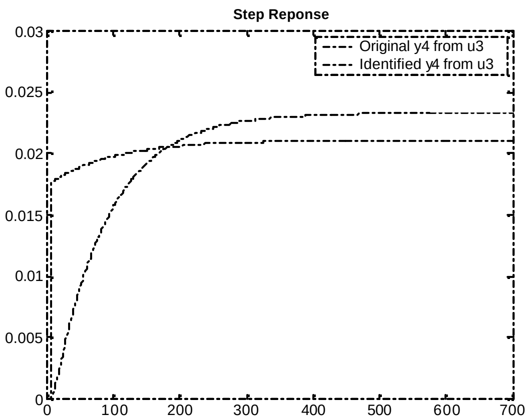

Figure 3.175 : Case 3- Step response of Output-4 from Input-3 (OE)

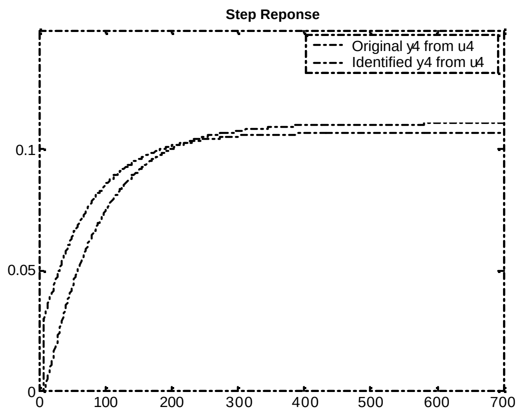

Figure 3.176 : Case 3- Step response of Output-4 from Input-4 (OE)

3.3.3.4 Performance of state space model

In the two previous cases state space modeling scheme failed to give accurate process models from given closed loop data. Its performance is now evaluated on this last case of plant model mismatch.

As before, sub space method is used to estimate the unknown parameters of the state space model. The order for the four process outputs are selected as 4, 3, 2 and 3 respectively. In Figs. 3.177 – 3.180, the step responses of the actual open loop model (solid) and the identified state space model (solid) for process output 1 are shown. The results show large bias and mismatch between the two models for all inputs. The steady state gain sign is also different for responses from inputs 1 and 4. Figs. 3.181 – 3.184 give the step response for the process output 2. The responses from inputs 2 and 4 are to an extent accurate with non-trivia mismatch. However, the responses from inputs 1 and 3 are inaccurate and exhibit large mismatch with the actual open loop model. The step responses for process output 3 are shown in Figs. 3.185 – 3.188. The responses from inputs 3 and 4 have different steady state gain signs as compared to the open loop model Clearly the model is inaccurate. The results for process output 4 are shown in Figs. 3.189 – 3.1192. As before, the state space model is unable to identify the open loop process dynamics correctly. The effect of plant-model mismatch is apparent on the performance of this scheme. Large inaccuracies and mismatch is observed between the identified state space model and the actual process model. For this case, the identified model is given in Eqs. 3.57 – 3.60.

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.99484 & -0.058446 & -0.016992 & 0.019203\\ 0.029961 & 0.82221 & -0.28224 & 0.070212\\ -0.021792 & 0.26304 & 0.70461 & 0.44358\\ 0.068491 & 0.20685 & -0.3377 & 0.50862\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\\ x_4(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.0068468 & -0.016217 & -0.0012721 & 0.017534\\ -0.025401 & -0.063956 & -0.030924 & 0.10758\\ 0.14505 & 0.0064645 & -0.020195 & 0.20405\\ -0.32106 & 0.059505 & -0.059226 & -0.16307\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.032776\\ -0.02424\\ 0.04994\\ -0.27512\end{bmatrix}e(k)$$

$$y_1(k)=\begin{bmatrix}2.3533 & 0.95809 & 0.87484 & -0.5067\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix} \tag{3.57}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.7934 & -0.24311 & 0.072664\\ -0.10044 & 0.83434 & 0.13325\\ 0.030407 & 0.078887 & 0.85794\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.16425 & 0.022425 & 0.0096869 & -0.010847\\ 0.091142 & -0.0014267 & 0.018959 & -0.015316\\ -0.034221 & 0.0078045 & -0.020525 & 0.010703\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.61478\\ -0.37563\\ -0.11904\end{bmatrix}e(k)$$

$$y_2(k)=\begin{bmatrix}1.0562 & -0.061084 & 0.0012186\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\\ x_4(k)\end{bmatrix} \tag{3.58}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.97014 & -0.021987\\ 0.0092153 & 0.9516\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.0039154 & 0.0068864 & -0.0026755 & 0.00058674\\ -0.0088743 & -0.015021 & -0.0051664 & 0.0012725\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.53323\\ -0.91381\end{bmatrix}e(k)$$

$$y_3(k)=\begin{bmatrix}1.4164 & -0.0075039\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\end{bmatrix} \tag{3.59}$$

$$\boldsymbol{x}(k+1)=\begin{bmatrix}0.96156 & -0.22378 & -0.10445\\ 0.094267 & -0.0092212 & 0.53024\\ -0.064251 & -0.34871 & 0.43769\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix}$$

$$+\begin{bmatrix}0.052303 & 0.00082182 & 0.029782 & 0.037528\\ 0.028252 & 0.10616 & 0.048177 & -0.097151\\ 0.23146 & -0.057977 & 0.080691 & 0.16074\end{bmatrix}\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\\ u_4(k)\end{bmatrix}+\begin{bmatrix}0.16907\\ -0.53126\\ -0.084537\end{bmatrix}e(k)$$

$$y_4(k)=\begin{bmatrix}1.7022 & 0.011502 & -0.10965\end{bmatrix}\begin{bmatrix}x_1(k)\\ x_2(k)\\ x_3(k)\end{bmatrix} \qquad (3.60)$$

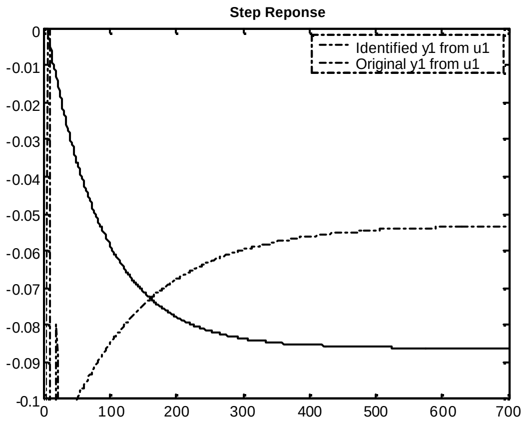

Figure 3.177 : Case 3- Step response of Output-1 from Input-1 (State space)

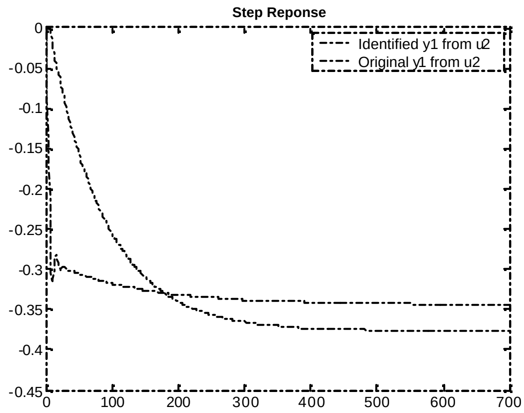

Figure 3.178 : Case 3- Step response of Output-1 from Input-2 (State space)

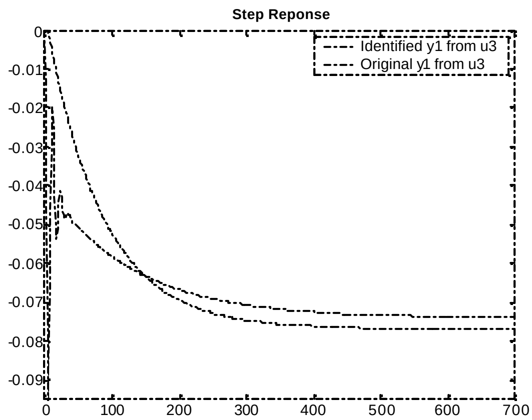

Figure 3.179 : Case 3- Step response of Output-1 from Input-3 (State space)

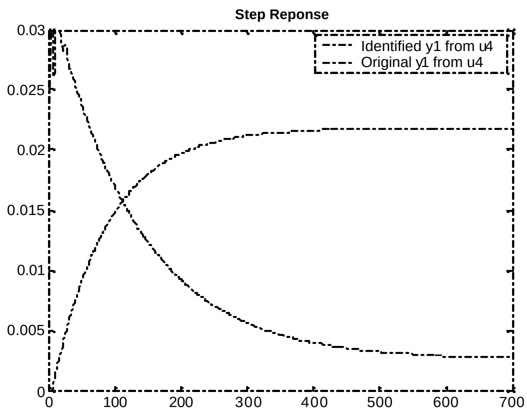

Figure 3.180 : Case 3- Step response of Output-1 from Input-4 (State space)

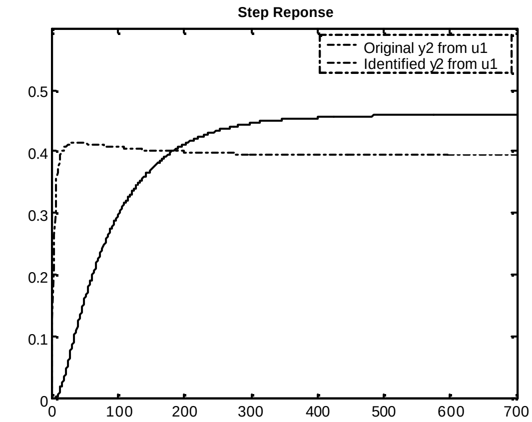

Figure 3.181 : Case 3- Step response of Output-2 from Input-1 (State space)

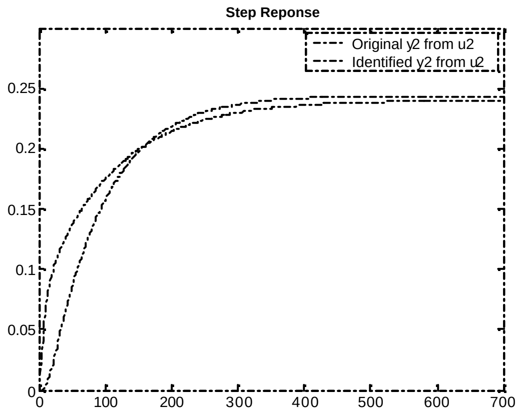

Figure 3.182 : Case 3- Step response of Output-2 from Input-2 (State space)

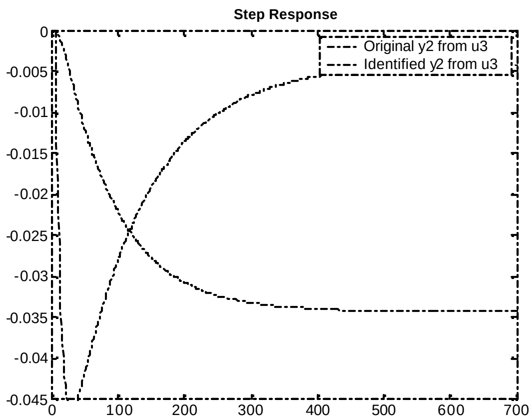

Figure 3.183 : Case 3- Step response of Output-2 from Input-3 (State space)

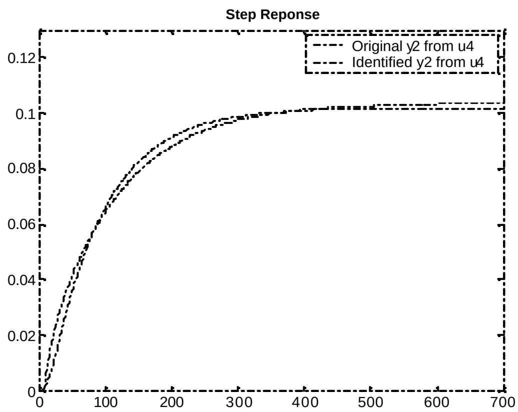

Figure 3.184 : Case 3- Step response of Output-2 from Input-4 (State space)

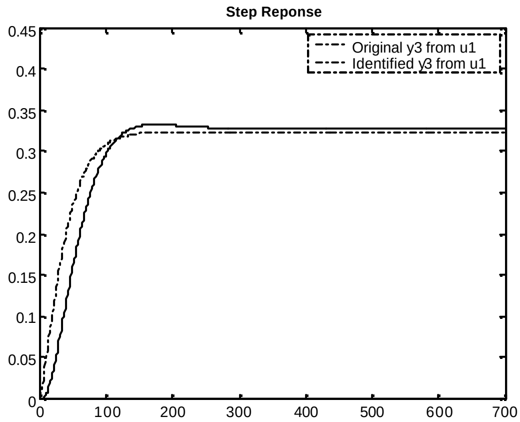

Figure 3.185 : Case 3- Step response of Output-3 from Input-1 (State space)

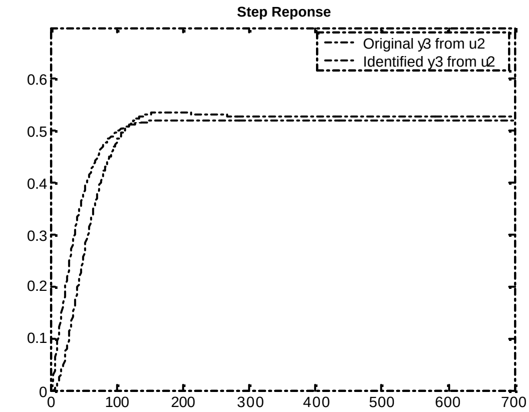

Figure 3.186 : Case 3- Step response of Output-3 from Input-2 (State space)

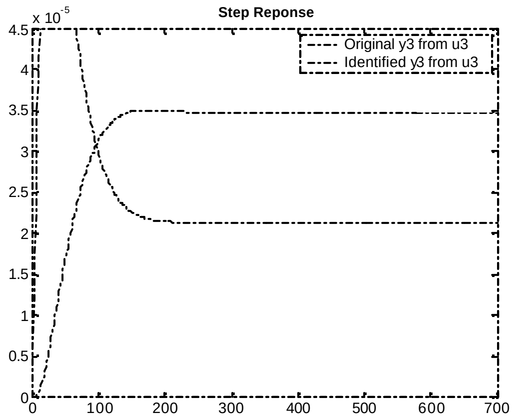

Figure 3.187 : Case 3- Step response of Output-3 from Input-3 (State space)

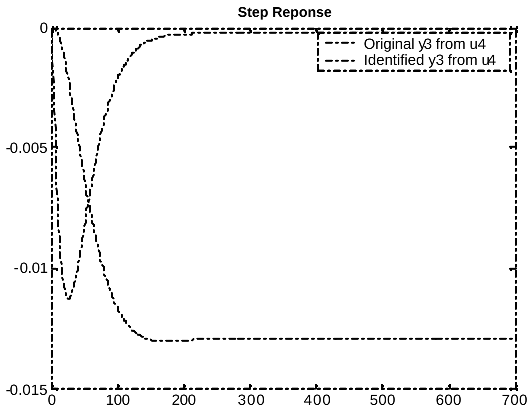

Figure 3.188 : Case 3- Step response of Output-3 from Input-4 (State space)

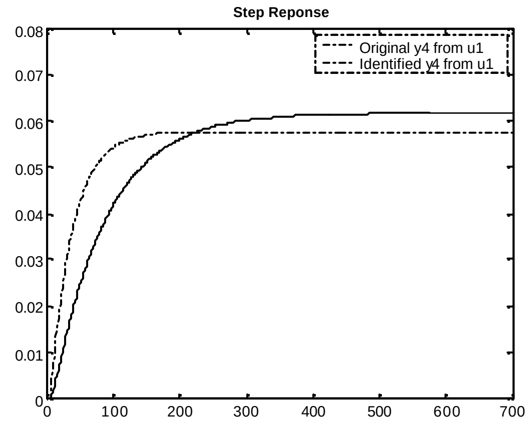

Figure 3.189 : Case 3- Step response of Output-4 from Input-1 (State space)

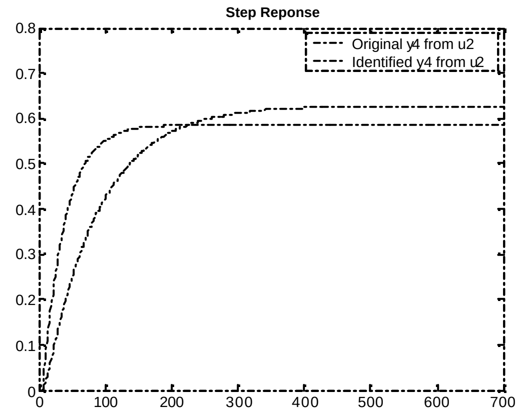

Figure 3.190 : Case 3- Step response of Output-4 from Input-2 (State space)

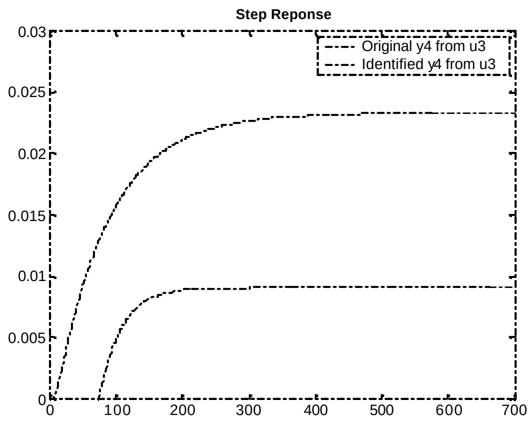

Figure 3.191 : Case 3- Step response of Output-4 from Input-3 (State space)

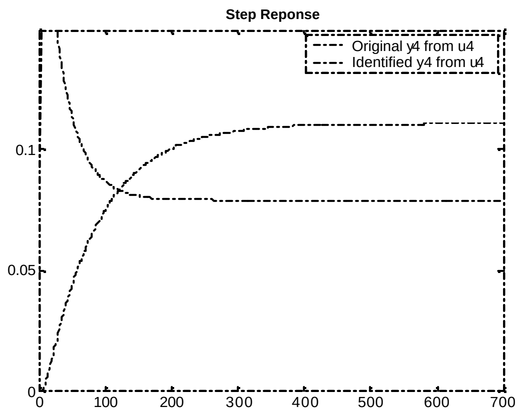

Figure 3.192 : Case 3- Step response of Output-4 from Input-4 (State space)

3.4 Chapter Summary

In this chapter some important results related to closed loop identification with MPC have been presented. ARX, ARMAX, OE and state space modeling schemes have been applied to three possible cases of MPC process. In the first case closed loop data is collected when the MPC process is running under high disturbances, noise and amplitude constraints. Models are identified in closed loop using this data. From the results it is found that ARX and ARMAX models are the only ones that gave good estimation from the closed loop data. In the second case the disturbances are reduced and rate constraints are added. Again ARX and ARMAX models give accurate estimation from the closed loop data. Although state space modeling scheme did show some improvement but overall the results demonstrated a high mismatch between the actual open loop MPC model and the identified state space closed loop model. In the third case, plant-model mismatch is taken into account. Again identification schemes based on ARX and ARMAX models manage to give good results. From these results, following important observations can be stated in the case of closed loop identification for MPC:

- Direct identification method works regardless of the complexity of MPC.
- Consistency and accuracy is achieved if the model structure contains the true system (including noise properties) as in the ARX and ARMAX modeling schemes.
- OE modeling scheme with a fixed noise model yield biased and inaccurate parameter estimates irrespective of the levels of disturbance and noise.

- State space models fail to give accurate representation of the process. This can be attributed to the fact that the input is correlated with high disturbance. This is in possible contradiction with the subspace method where it is assumed that the input is uncorrelated with the process noise and disturbances.
- In ARX and AMRAX modeling schemes the poles of the process and disturbance/noise model are the same which is a precondition for obtaining a stable model [36]. This is not the case for OE modeling scheme.
- The compactness of ARMAX modeling scheme is highest because the disturbance is explicitly modeled. This means that the order of the system is generally smaller as compared to ARX. Same holds for OE and State space modeling schemes. Further discussion on this subject is given in [34].
- The numerical complexity is highest for ARMAX scheme because prediction error method involves complex optimization routines which results in a large amount of numerical computations. Same is true for OE modeling scheme.
- State space models are much simpler to implement as the Kalman filter states are obtained directly from input output data using linear algebra tools, after which the identification problem reduces to least squares problem.

From these results a benchmark of all these models can now be made in tabulated form. Table 3-1 categorizes these parametric models in terms of their compactness (less parameters to describe the process dynamics) and numeric complexity (optimization routines). In Table 3-2 the results of the simulations are presented which summarizes the findings of this work.

Table 3-1 Comparison of various model structures

Model Structure	**Numerical difficulty**	**Compactness**
ARX	Low	Medium
ARMAX	High	Highest
OE	High	High
State space	Low	High

Table 3-2 Comparison of performance of various model structures

Model Structure	**Amplitude constraints, high disturbances, noise**	**Amplitude & Rate constraints, medium disturbances, noise**	**Amplitude & constraints, nominal disturbances, noise, mismatch**
ARX	Best	Best	Best
ARMAX	Best	Best	Best
OE	Poor	Poor	Poor
State space	Poor	Poor	Poor

Chapter 4

Closed Loop Identification – Process Data

4.1 Introduction

In Chapter 3, it has been shown that ARX and ARMAX modeling schemes work fine and give consistent estimates of the open loop system with closed-loop data. In this Chapter, the results of model estimation from closed loop process data collected during normal process operation are presented. The process is again the Demethanizer column controlled using MPC from a gas plant in Saudi Arabia. The data is collected over a three month time period with a sampling time of 1 minute. It is essential to point out here that closed loop data samples collected have to be large enough so as to exhibit the process dynamics correctly. Data for a month or less may not be enough for closed loop identification purposes as the relevant plant behavior to specified set points may not be represented in it.

Before proceeding with identification from this data it is necessary to do some pre-processing of the closed loop data. This is explained in the next section.

4.2 Pre-Processing of the Closed Loop Data

4.2.1 Outliers and Bad Data

Real closed loop data collected from the process under operation is subject to possible missing data samples mostly due to malfunctions in the sensors or communication links. Moreover, certain measured values can be in obvious error due to measurement failures. Such bad values are often called outliers, and have a substantial effect on the model estimation. To deal with outliers and bad data, there are a few possibilities. One is to cut out segments of the data sequence so that portions with bad data are avoided. In this case it is natural to select segments of the original data set which are considered to contain relevant information about the dynamics of the system. There is no hard and fast rule for this procedure and it is basically subjective to intuition and process insights.

4.2.2 Drifts and Trends

Low-frequency disturbances, offsets, trends, drift and periodic variations are not uncommon in closed loop data. They typically stem from external sources that may not be relevant to modeling. Ljung [36] has suggested a basic approach to dealing with such a problem. It involves removing these trends by explicit pretreatment of the data. This involves removing trends and offsets by removing the mean values from the signal. This procedure simply implies that the mean values of both output *(y)* and input *(u)* data are computed as follows

$$y^*(t) = \frac{1}{N}\sum_{t=1}^{N} y(t) \qquad (4.1)$$

$$u^*(t) = \frac{1}{N}\sum_{t=1}^{N} u(t) \tag{4.2}$$

and then compute the ‘detrended’ data by

$$\bar{y}(t) = y(t) - y^*(t) \tag{4.3}$$

$$\bar{u}(t) = u(t) - u^*(t) \tag{4.4}$$

4.3 Closed Loop Process Data

As mentioned in section 4.1, closed loop field data is made available from the Demethanizer column process running with MPC. It is collected for three months period at a sampling time of 1 minute using step testing. After some pre-processing and detrending, the data is made ready for the purpose of modeling and estimation. ARX and AMRAX modeling schemes have shown to be effective in estimating a reliable model from closed loop data collected from processes running with MPC. These schemes are now tested and verified on this ‘real’ closed loop data. The following subsections give further details about the results of these simulations.

4.3.1 Performance of ARX Models

ARX models are now used to estimate relationships between the outputs and inputs of the given closed loop field data. Recall that the ARX models are of the form as given below.

$$A(q^{-1})\,y(t) = q^{-d}\,B(q^{-1})\,u(t) + e(t) \tag{4.5}$$

The order n_a of the polynomial A is selected as 3, 2, 5 and 5 respectively for the four process outputs. Least squares method is used to estimate the unknown parameters of the

ARX model. The estimated model is given in Eqs. 4.6 - 4.9. The response of the identified ARX closed loop model is plotted with the actual closed loop data for each output in Figs. 4.1 – 4.4. The results demonstrate excellent performance of the identified ARX model in reproducing the actual data. The error is minimal between the two responses. In order to analyze the performance of the identified ARX model in recovering open loop process model from the closed loop field data, the step responses for the actual open loop process model (solid line) and the closed loop identified ARX model (dashed line) are plotted in Figs. 4.5 - 4.20. These results reaffirm the excellent capability of the ARX modeling scheme. There is minimal or no mismatch between the responses of the two models. For all process outputs the steady state part is captured accurately. No mismatch, whatsoever is observed in the step responses. This has verified that closed loop identification for processes running with MPC can be performed successfully by using ARX modeling scheme, which can estimate the open loop process dynamics in closed loop. This has also confirmed the conclusion made in chapter 3, where ARX modeling scheme is shown to have given good representation of the open loop process model from simulated data. Thus, if suitable amount of data samples are collected from processes running with MPC, open loop process model can be identified using this scheme without MPC controller shutdown.

$$
\begin{aligned}
y1(t) = {} & \frac{-0.0001963}{1-1.781q^{-1}+0.7917q^{-2}-0.008375q^{-3}}\, q^{-1}\, u_1(t) \\
& + \frac{-0.0008560}{1-1.781q^{-1}+0.7917q^{-2}-0.008375q^{-3}}\, q^{-1}\, u_2(t) \\
& + \frac{-0.0001748}{1-1.781q^{-1}+0.7917q^{-2}-0.008375q^{-3}}\, q^{-1}\, u_3(t) \\
& + \frac{0.00004958}{1-1.781q^{-1}+0.7917q^{-2}-0.008375q^{-3}}\, q^{-1}\, u_4(t)
\end{aligned}
\tag{4.6}
$$

$$y2(t) = \frac{0.0017}{1-1.643q^{-1} + 0.6466q^{-2}} q^{-2}\, u_1(t)$$

$$+ \frac{0.0008880}{1-1.643q^{-1} + 0.6466q^{-2}} q^{-2}\, u_2(t)$$

$$+ \frac{-0.0001249}{1-1.643q^{-1} + 0.6466q^{-2}} q^{-2}\, u_3(t)$$

$$+ \frac{0.0003705}{1-1.643q^{-1} + 0.6466q^{-2}} q^{-2}\, u_4(t) \tag{4.7}$$

$$y3(t) = \frac{0.0033}{1-1.487q^{-1}+0.7437q^{-2}-0.6307q^{-3}+0.5986q^{-4}-0.2146q^{-5}} q^{-1}\, u_1(t)$$

$$+ \frac{0.0053}{1-1.487q^{-1}+0.7437q^{-2}-0.6307q^{-3}+0.5986q^{-4}-0.2146q^{-5}} q^{-1}\, u_2(t)$$

$$+ \frac{0.0000003445}{1-1.487q^{-1}+0.7437q^{-2}-0.6307q^{-3}+0.5986q^{-4}-0.2146q^{-5}} q^{-1}\, u_3(t)$$

$$+ \frac{-0.0001285}{1-1.487q^{-1}+0.7437q^{-2}-0.6307q^{-3}+0.5986q^{-4}-0.2146q^{-5}} q^{-1}\, u_4(t) \tag{4.8}$$

$$y4(t) = \frac{0.0001778}{1-1.496q^{-1}+0.3345q^{-2}+0.1237q^{-3}+0.01932q^{-4}+0.021q^{-5}} q^{-2}\, u_1(t)$$

$$+ \frac{0.0018}{1-1.496q^{-1}+0.3345q^{-2}+0.1237q^{-3}+0.01932q^{-4}+0.021q^{-5}} q^{-2}\, u_2(t)$$

$$+ \frac{0.00006689}{1-1.496q^{-1}+0.3345q^{-2}+0.1237q^{-3}+0.01932q^{-4}+0.021q^{-5}} q^{-2}\, u_3(t)$$

$$+ \frac{0.0003189}{1-1.496q^{-1}+0.3345q^{-2}+0.1237q^{-3}+0.01932q^{-4}+0.021q^{-5}} q^{-2}\, u_4(t) \tag{4.9}$$

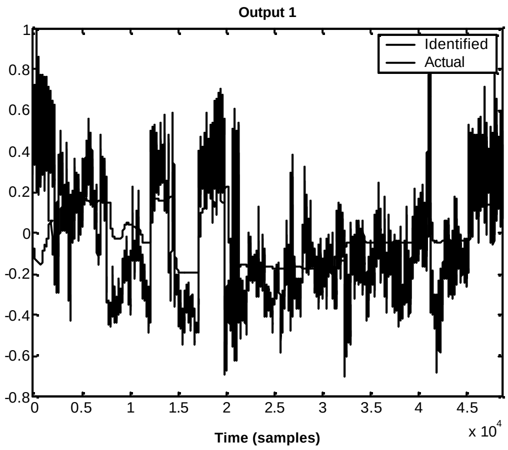

Figure 4.1 : Responses of Output 1 Actual and Identified ARX Model

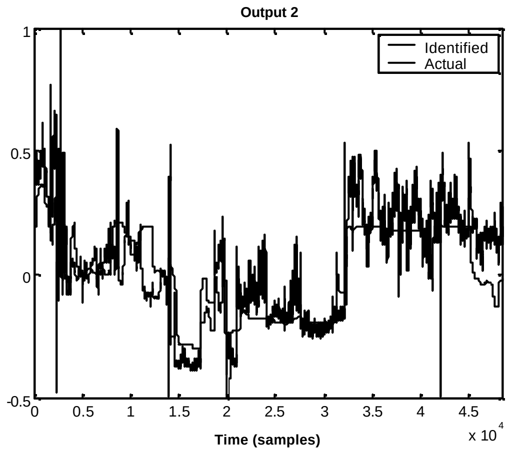

Figure 4.2 : Responses of Output 2 Actual and Identified ARX Model

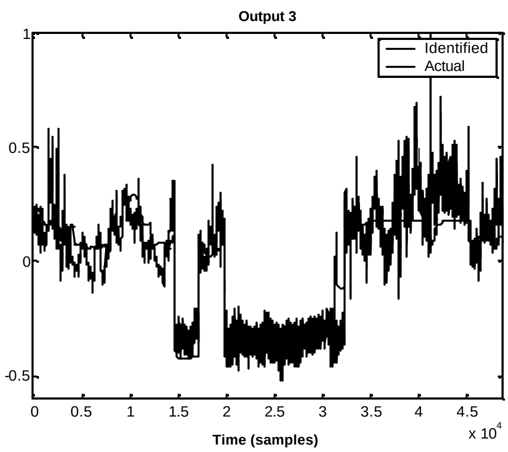

Figure 4.3: Responses of Output 3 Actual and Identified ARX Model

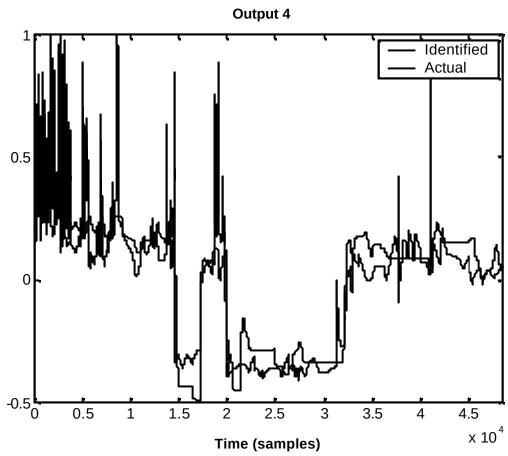

Figure 4.4: Responses of Output 4 Actual and Identified ARX Model

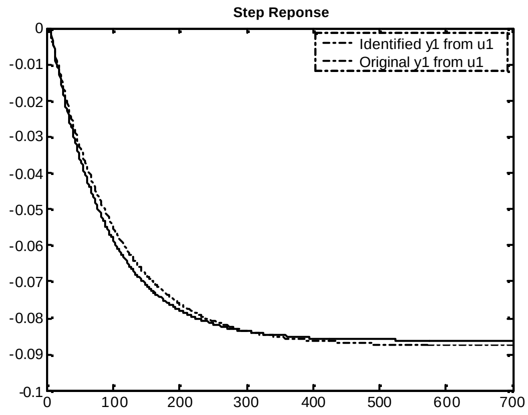

Figure 4.5 : Step Responses of the Actual Open Loop and Identified ARX models (Output-1 from Input-1)

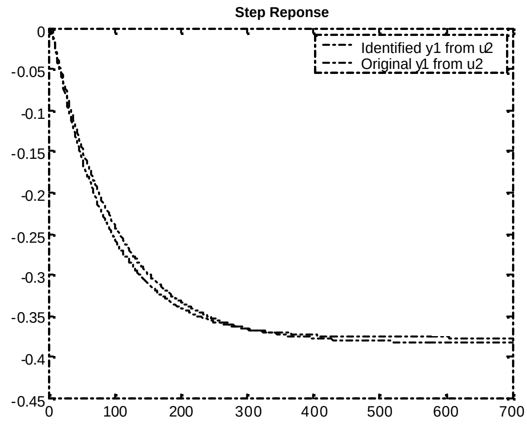

Figure 4.6 : Step Responses of the Actual Open Loop and Identified ARX models (Output-1 from Input-2)

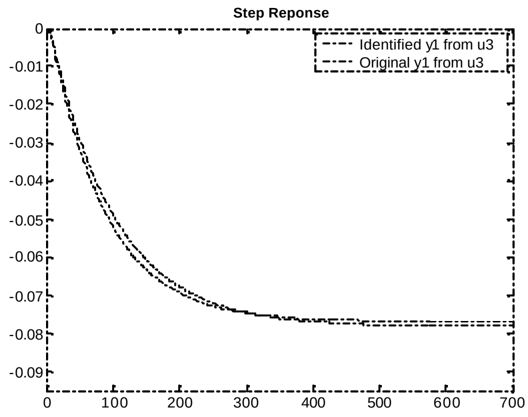

Figure 4.7 : Step Responses of the Actual Open Loop and Identified ARX models (Output-1 from Input-3)

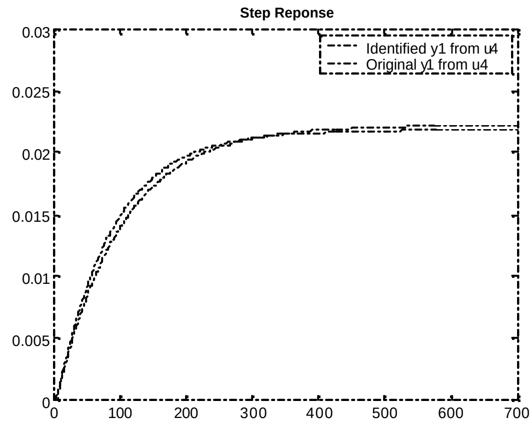

Figure 4.8 : Step Responses of the Actual Open Loop and Identified ARX models (Output-1 from Input-4)

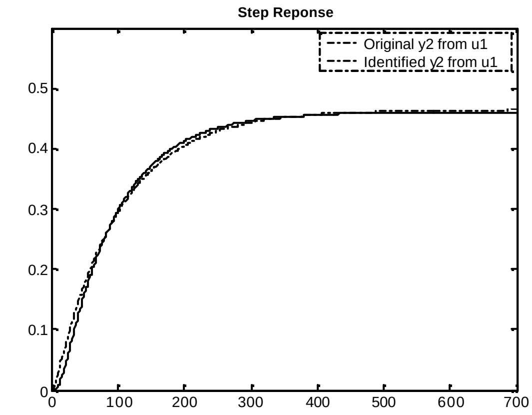

Figure 4.9 : Step Responses of the Actual Open Loop and Identified ARX models (Output-2 from Input-1)

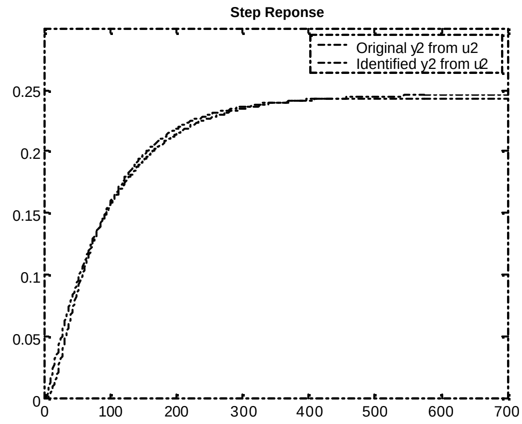

Figure 4.10 : Step Responses of the Actual Open Loop and Identified ARX models (Output-2 from Input-2)

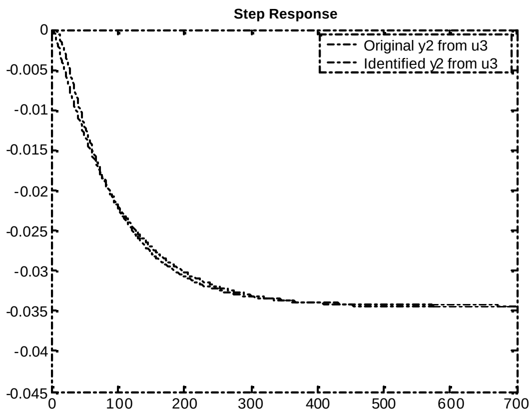

Figure 4.11 : Step Responses of the Actual Open Loop and Identified ARX models (Output-2 from Input-3)

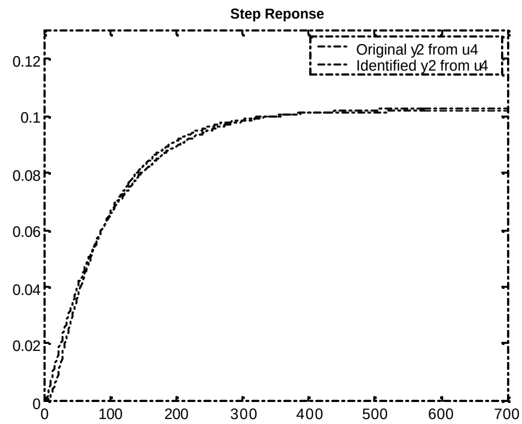

Figure 4.12 : Step Responses of the Actual Open Loop and Identified ARX models (Output-2 from Input-4)

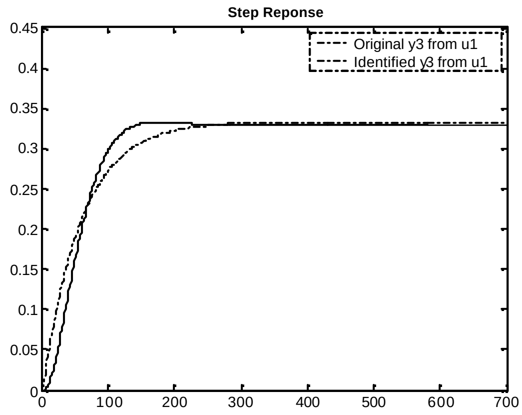

Figure 4.13 : Step Responses of the Actual Open Loop and Identified ARX models (Output-3 from Input-1)

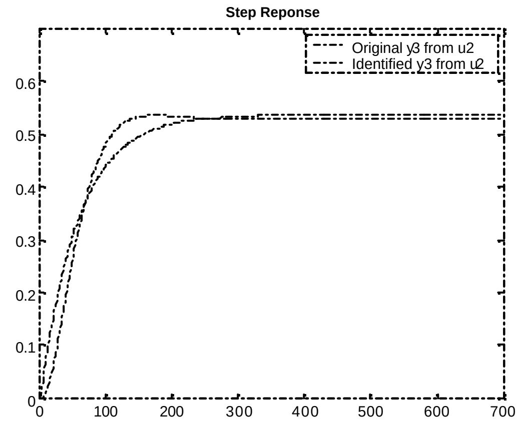

Figure 4.14 : Step Responses of the Actual Open Loop and Identified ARX models (Output-3 from Input-2)

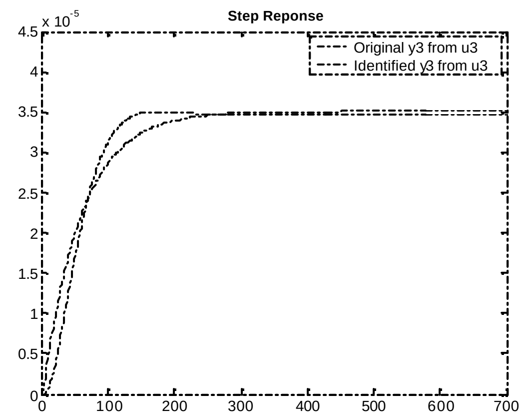

Figure 4.15 : Step Responses of the Actual Open Loop and Identified ARX models (Output-3 from Input-3)

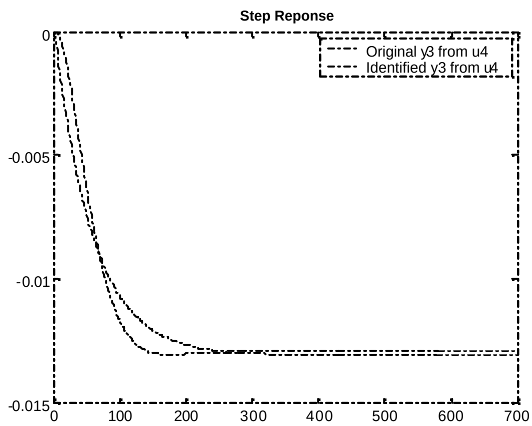

Figure 4.16 : Step Responses of the Actual Open Loop and Identified ARX models (Output-3 from Input-4)

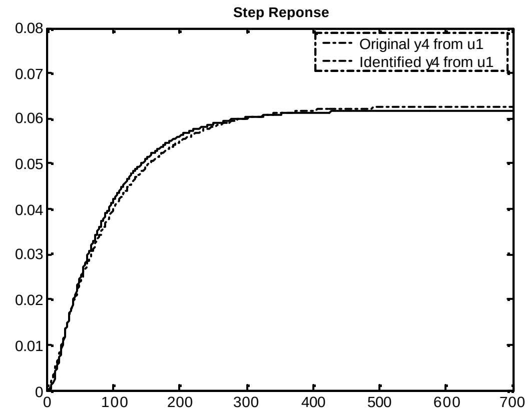

Figure 4.17 : Step Responses of the Actual Open Loop and Identified ARX models (Outp ut-4 from Input-1)

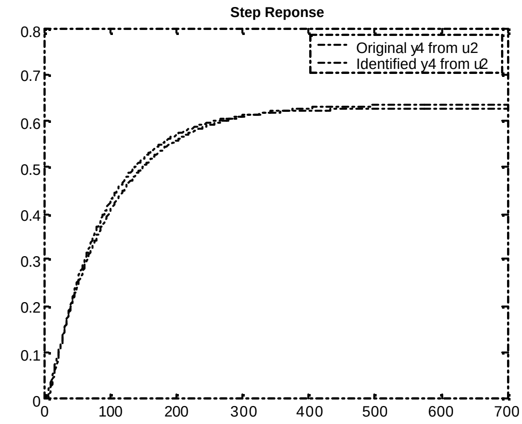

Figure 4.18 : Step Responses of the Actual Open Loop and Identified ARX models (Output-4 from Input-2)

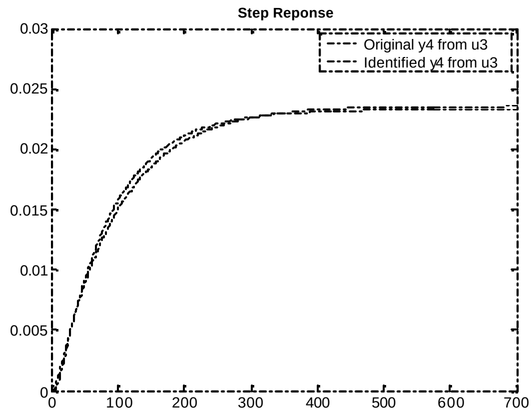

Figure 4.19 : Step Responses of the Actual Open Loop and Identified ARX models (Output-4 from Input-3)

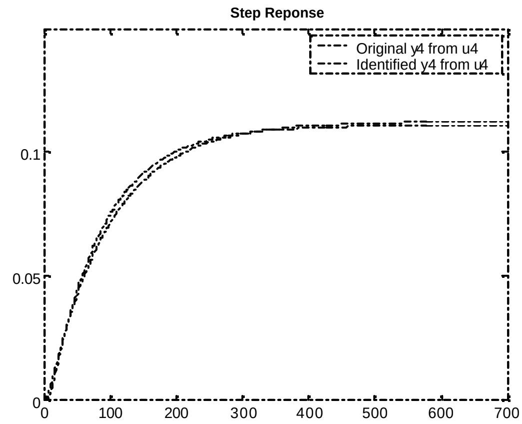

Figure 4.20 : Step Responses of the Actual Open Loop and Identified ARX models (Output-4 from Input-4)

4.3.2 Performance of ARMAX Models

In addition to ARX scheme, ARMAX modeling scheme have also shown equivalent performance if not better, when used for identification from closed loop data obtained from a process running with MPC. In this subsection, ARMAX modeling scheme is tested for its performance on real closed loop data. Recall that an ARMAX model is of the type

$$A(q^{-1})\,y(t) = B(q^{-1})\,u(t) + C(q^{-1})e(t) \tag{4.10}$$

The order n_a of the polynomial A is selected as 2, 2, 3 and 3 respectively. Responses of the closed loop ARMAX model and the actual closed loop data are shown in Figs. 4.21 - 4.24. As in the case of ARX modeling scheme the response of the identified ARMAX model is identical to the actual closed loop field data. Not taking into consideration the oscillations in the actual data due to disturbance and noise, it can be stated that the response of the model is a perfect match. This identified ARMAX model is now compared with the actual open loop process model. The step responses of the actual open loop model (solid) and the closed loop ARMAX model (dashed) are shown in Figs. 4.25 - 4.40. It is noted that for process output 1 there is no mismatch between the responses. For process output 2 there is however a trivial mismatch at the steady state, but the error is very small. Reponses for process outputs 3 and 4 show perfect fit at the steady state. This also confirms the conclusion made in Chapter 3. ARMAX modeling scheme manages to successfully recover the open loop dynamics of the process from closed loop data. The estimated model is given in the transfer function format in Eqs. 4.11 - 4.15, where unlike ARX scheme the noise has also been estimated explicitly.

$$y1(t) = \frac{-0.0001873}{1-1.824q^{-1}+0.826q^{-2}}q^{-12}u_1(t)$$

$$+\frac{-0.0008169}{1-1.824q^{-1}+0.826q^{-2}}q^{-12}u_2(t)$$

$$+\frac{-0.0001668}{1-1.824q^{-1}+0.826q^{-2}}q^{-12}u_3(t)$$

$$+\frac{-0.00004731}{1-1.824q^{-1}+0.826q^{-2}}q^{-12}u_4(t)$$

$$+\frac{1+0.2774q^{-1}}{1-1.824q^{-1}+0.826q^{-2}}e(t) \tag{4.11}$$

$$y2(t) = \frac{0.0014}{1-1.67q^{-1}+0.6733q^{-2}}q^{-16}u_1(t)$$

$$+\frac{0.0007467}{1-1.67q^{-1}+0.6733q^{-2}}q^{-16}u_2(t)$$

$$+\frac{-0.0001050}{1-1.67q^{-1}+0.6733q^{-2}}q^{-16}u_3(t)$$

$$+\frac{0.0003115}{1-1.67q^{-1}+0.6733q^{-2}}q^{-16}u_4(t)$$

$$+\frac{1+0.1391q^{-1}}{1-1.67q^{-1}+0.6733q^{-2}}e(t) \tag{4.12}$$

$$y2(t) = \frac{0.0029}{1-1.381q^{-1}+0.2386q^{-2}+0.1518q^{-3}}q^{-1}u_1(t)$$

$$+\frac{0.0047}{1-1.381q^{-1}+0.2386q^{-2}+0.1518q^{-3}}q^{-1}u_2(t)$$

$$+\frac{0.0000003080}{1-1.381q^{-1}+0.2386q^{-2}+0.1518q^{-3}}q^{-1}u_3(t)$$

$$+\frac{-0.0001149}{1-1.381q^{-1}+0.2386q^{-2}+0.1518q^{-3}}q^{-1}u_4(t)$$

$$+\frac{1+0.3375q^{-1}-0.1972q^{-2}}{1-1.381q^{-1}+0.2386q^{-2}+0.1518q^{-3}}e(t) \tag{4.13}$$

$$
\begin{aligned}
y4(t) = {} & \frac{0.00003642}{1-2.415q^{-1}+1.884q^{-2}-0.4679q^{-3}} q^{-6}\, u_1(t) \\
& + \frac{0.0003701}{1-2.415q^{-1}+1.884q^{-2}-0.4679q^{-3}} q^{-6}\, u_2(t) \\
& + \frac{0.00001370}{1-2.415q^{-1}+1.884q^{-2}-0.4679q^{-3}} q^{-6}\, u_3(t) \\
& + \frac{0.00006533}{1-2.415q^{-1}+1.884q^{-2}-0.4679q^{-3}} q^{-6}\, u_4(t) \\
& + \frac{1-0.4116q^{-1}}{1-2.415q^{-1}+1.884q^{-2}-0.4679q^{-3}}\, e(t)
\end{aligned}
\tag{4.14}
$$

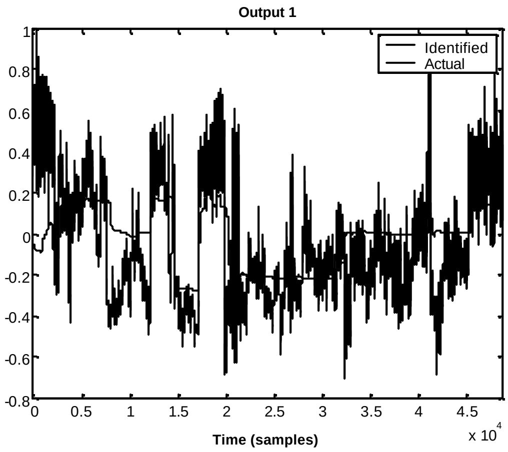

Figure 4.21: Responses of Output 1 Actual and Identified ARMAX Model

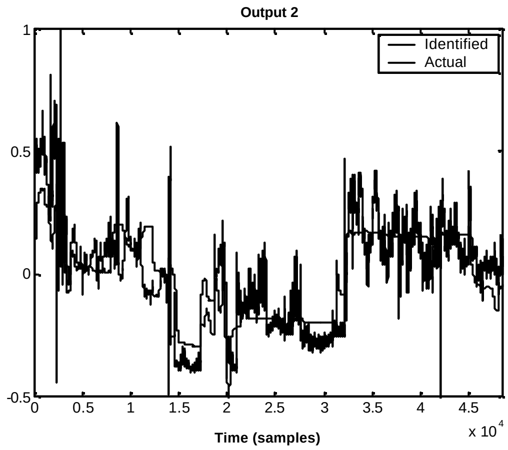

Figure 4.22: Responses of Output 2 Actual and Identified ARMAX Model

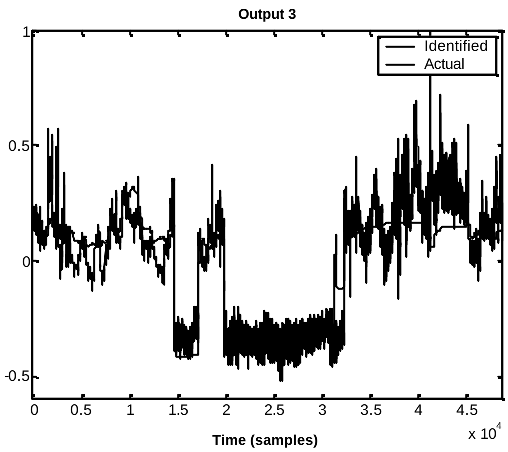

Figure 4.23: Responses of Output 3 Actual and Identified ARMAX Model

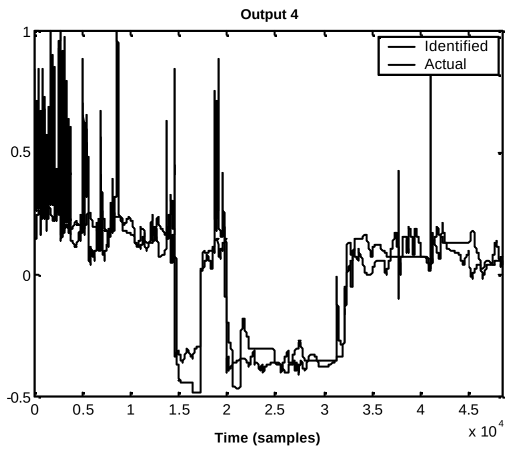

Figure 4.24: Responses of Output 4 Actual and Identified ARMAX Model

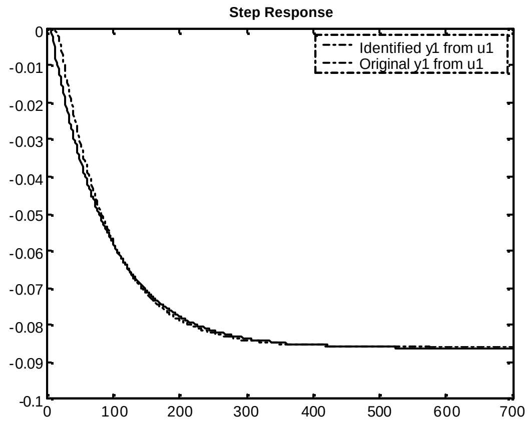

Figure 4.25: Step Responses of the Actual Open Loop and Identified ARMAX models (Output-1 from Input-1)

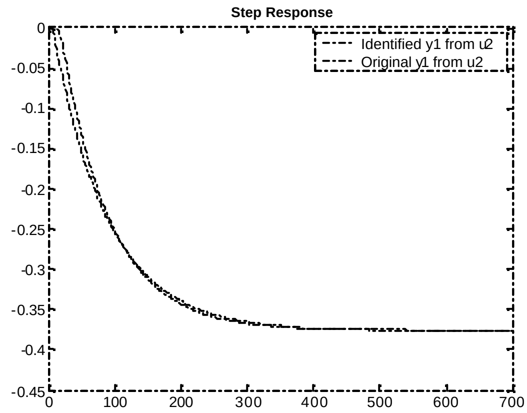

Figure 4.26: Step Responses of the Actual Open Loop and Identified ARMAX models (Output-1 from Input-2)

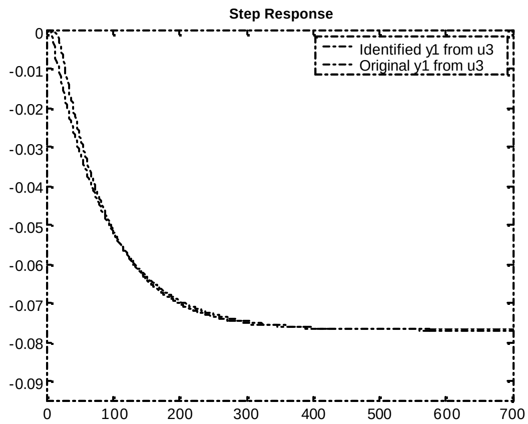

Figure 4.27 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-1 from Input-3)

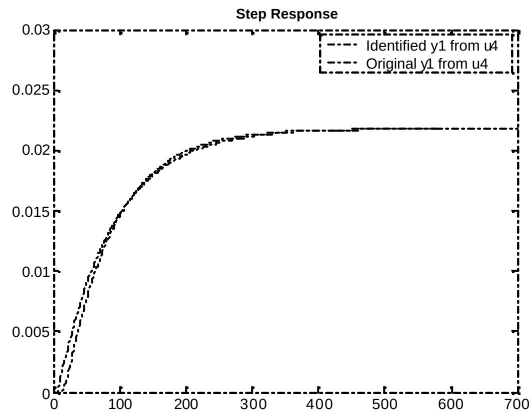

Figure 4.28 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-1 from Input-4)

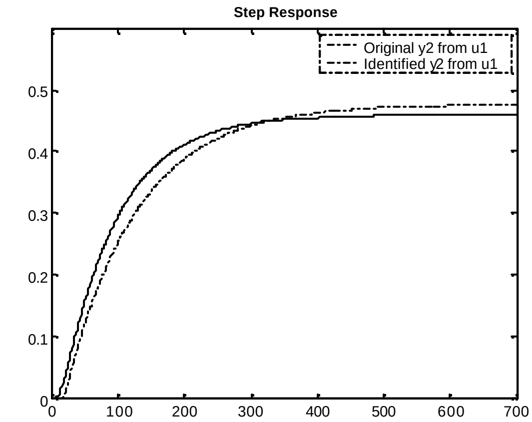

Figure 4.29 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-2 from Input-1)

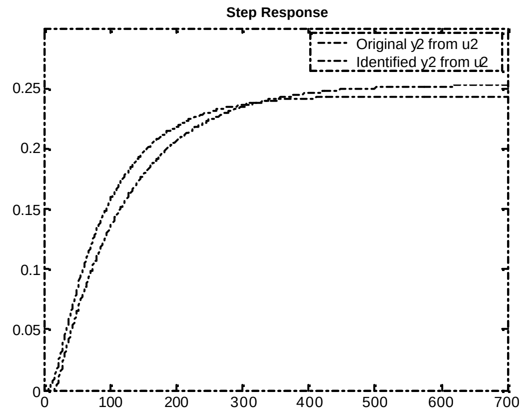

Figure 4.30 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-2 from Input-2)

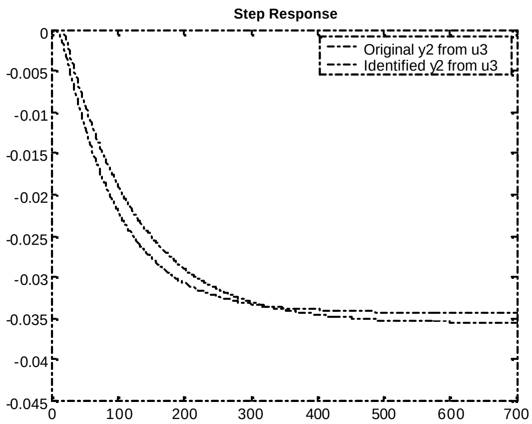

Figure 4.31 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-2 from Input-3)

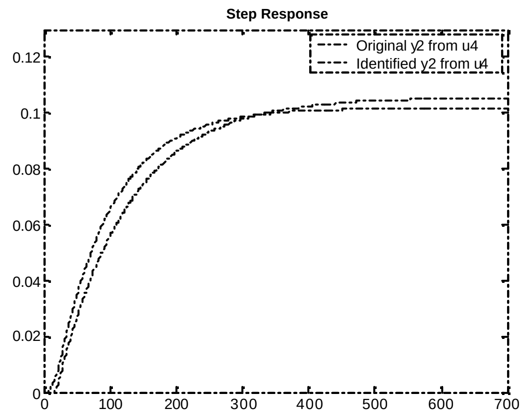

Figure 4.32 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-2 from Input-4)

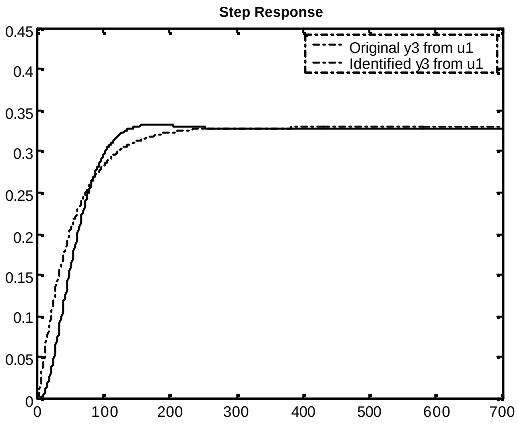

Figure 4.33 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-3 from Input-1)

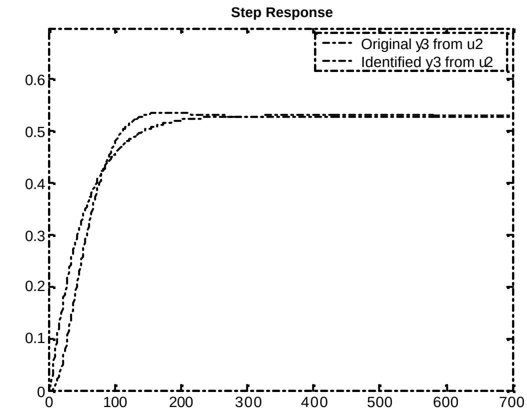

Figure 4.34 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-3 from Input-2)

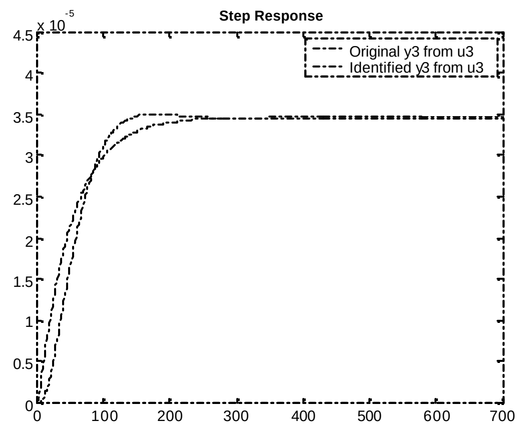

Figure 4.35 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-3 from Input-3)

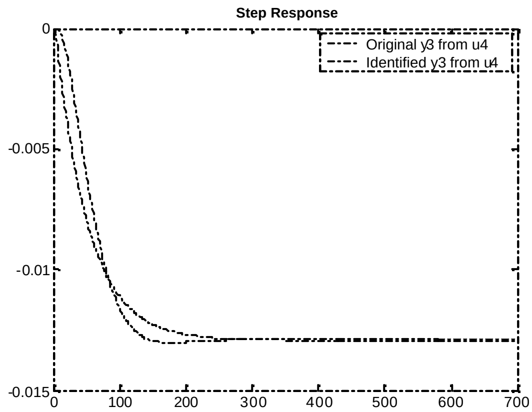

Figure 4.36 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-3 from Input-4)

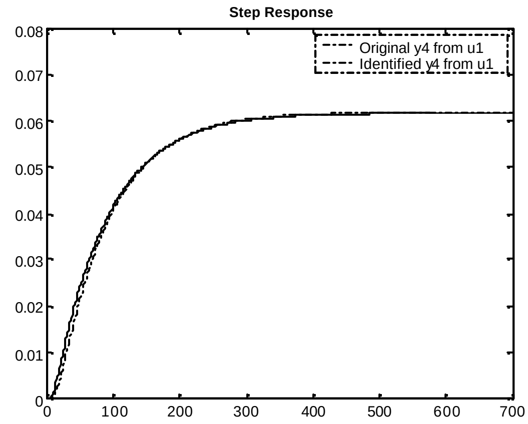

Figure 4.37 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-4 from Input-1)

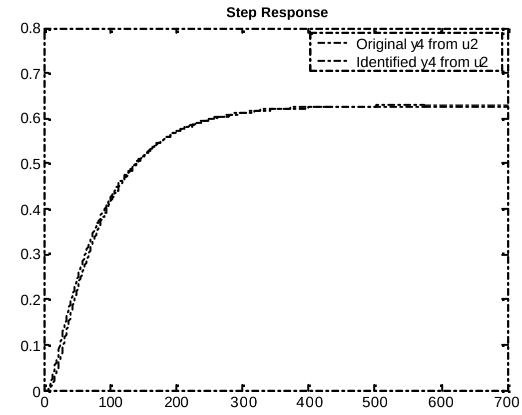

Figure 4.38 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-4 from Input-2)

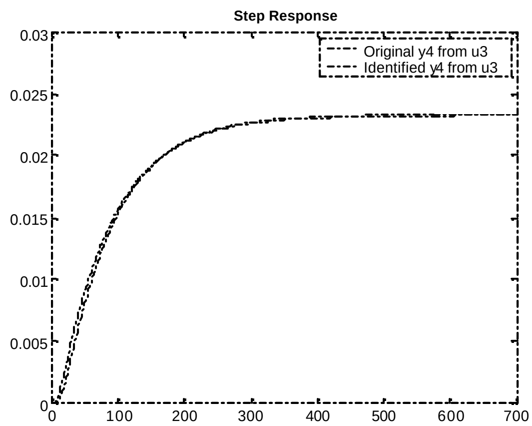

Figure 4.39 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-4 from Input-3)

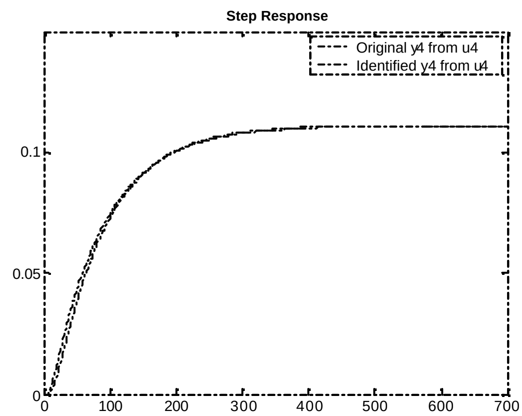

Figure 4.40 : Step Responses of the Actual Open Loop and Identified ARMAX models (Output-4 from Input-4)

4.4 Chapter Summary

In this chapter, identification based on ARX and ARMAX modeling schemes have been successfully applied to actual process closed loop data obtained from a Demethanizer column running with MPC. As discussed in Section 3.4, the compactness of ARMAX model as compared to ARX model is higher. The order of ARMAX is 2, 2, 3 and 3 whereas the order of ARX is 3, 2, 5 and 5, respectively for the four outputs.

Sometimes it may be required to have a closed loop model while a process running with MPC is in operation. In this case, the model has to be based on observations up to the current time. For this purpose recursive identification methods are used. These methods are sometimes also referred to as on-line or real time identification methods. In this regard recursive identification schemes based on ARX and ARMAX modeling schemes can also be used. These recursive variants will give similar results to batch ARX and ARMAX schemes, if not better.

Chapter 5

Identification - Neural networks

5.1 Introduction

In conventional linear identification schemes whether it be least squares or prediction error methods, the focus has been on estimation of the true plant from closed loop data using linear models for the purpose of MPC design. However, a new direction has emerged in the past few years in which nonlinear models are being used in the design of MPC. This emerging field is termed as NonLinear Model Predictive Control (NLMPC). NLMPC is presently viewed as one of the most promising areas in automatic control. This is partly due to the increasing industrial need for advanced control techniques that address explicitly the process nonlinearity and operation constraints and the ever-demanding control performance requirement.

However, despite the wide publicity and the intensive research efforts, it is still being perceived as an academic concept rather than a practical control strategy. One reason for this disparity is the inability to construct a nonlinear model on a reliable and consistent basis. An important factor that has been emphasized throughout this thesis is that the identified models are to be used in a closed loop environment. Because systems' input output pattern can change dramatically after closing a loop, it is entirely possible that a model that provides good performance in open loop may lose that capability once the predictive controller is designed and loop is closed. While it may be possible to establish patterns of disturbances and system noises before a closed loop implementation, either from prior knowledge or from available open loop plant data, it is generally very difficult to do the same for manipulated inputs whose patterns will depend on, among many things, the controller. Therefore, in NLMPC, a model should be able to handle accurately the effects of both known and unknown changes on the system (output) behavior in a closed loop setting. For this purpose, a Multilayer FeedForward Neural Network (MFNN) model can be used.

5.2 Neural Networks for System Identification

Neural networks, in general, are not new to the field of identification. Since 1990 many papers have not only demonstrated promising results in applying the approaches of neuro-identification, but also have begun to address fundamental issues such as system approximation and identification, controllability, observability and stability theory. Although major results in approximation and identification of systems using neural networks are available, only a small group of people are actually familiar with them. Perhaps the most popular structure has been the static Multi-layer FeedForward Neural Network (MFNN) trained via the back propagation-learning algorithm. In this structure, the neurons are generally grouped into layers. Input signals propagate through the network in a forward direction, layer by layer, through to the output layer. On the other hand, with the incorporation of feedback connections and delay elements within the neurons, static neural networks are made recurrent by construction. Recurrent Neural Networks (RNN) are characterized by their internal memory and thus are very suitable for imitating the behavior of dynamic systems. This type of networks allow connections between any pair of neurons but keep the concept of input and output neurons inherent to MFNN. Such networks were first proposed by Hopfield in 1982 and have recently been rediscovered as dynamic neural networks (DNN) in the context of identification and control of dynamic system.

5.3 Major Works on MFNN for System Identification

Much of the early research on neural networks for system identification date back to beginning 1990s when Narendra and Parthasarathy [67] demonstrated that MFNN structure could be effectively used for identification and control. The same year Bhat et al. [68] used neural networks for modeling nonlinear chemical process systems such as steady-state reactor and a dynamic pH continuously stirred tank reactor. They used the back-propagation algorithm for interpreting biosensor data by utilizing MFNN modeling scheme. In 1991, Tai et al. [69] presented a survey report on the algorithms and techniques of neural networks including MFNN implemented in the areas of identification and control. Around the same time, the lack of generic and efficient methodology for nonlinear system identification with unknown system architecture prompted Qin et al. [70] to re-derive pattern learning and batch learning rules for both MFNN and RNN respectively. This was one of the pioneering works in black-box modeling vis-à-vis neural networks. Chen and Mars [71] discussed the feasibility of using MFNN for system identification. They scrutinized the work of Narendra et. al. [67] and provided some solutions to the constraints pointed out in that work. In 1993, Yamada and Yabuta [72] proposed practical design methods for the identification of both the direct and inverse transfer functions of a nonlinear dynamic system through the use of neural networks. Sjoberg [73] in 1994 utilized MFNN based NNARX modeling techniques to simulate nonlinear systems having different kinds of non-linearities.

In 1995, Songwu and Tamer [74] presented system identification schemes in a neural network framework, using FFNN. Both MFNN and Radial Basis Function Neural Networks (RBFNN) were used to identify nonlinear systems in the presence of unknown driving noise. Judistky et al. [75] surveyed and discussed different techniques including MFNN for this purpose. As a companion paper to this Sjoberg et al. [76] compared by simulation, the performance of MFNN based NNARX model to other nonlinear identification techniques. Abdallah et al. [77] in their technical report addressed issues of capabilities versus actual performance of MFNN for both discrete time and continuous time cases.

In 1996, Mhaskar [78] examined the complexity of MFNN required to approximate an unknown system to a degree of accuracy for a worst-case scenario. Moody [79] presented a new 'dependence identification' algorithm for developing a new form of MFNN for system identification. This proposed algorithm transformed the training problem into a set of quadratic optimization problems that were solved by a number of linear equations. Suykens and Bersini [80] studied nonlinear system identification using MFNN with respect to model based control. Mauro [81] in his PhD thesis applied MFNN for model updating in closed loop. Duwaish et al. [82] showed the use of MFNN for learning nonlinear relationships from plant input output data.

In 1998, Sorensen [83] used NNARMAX modeling technique to identify and predict a nonlinear system. Songwu and Tamer [84] used MFNN and RBFNN to identify nonlinear systems in the presence of driving noise. They used H^{∞} and genetic based identification algorithms for network parameters update. In 2000, Bendtsen and Sorensen [85] used MFNN networks for the identification of a nonlinear injection valve for a super-heater attemporator at a power plant. In 2001, Miima et al. [86] utilized MFNN for modeling input-output behavior of points on a deforming bridge. Recently Norgaard et al. [87] have developed two toolsets for system identification and control with neural networks. These are NNSYID and NNCTRL for use in Matlab engineering software.

5.4 MFNN

In feedforward family of neural networks, MFNN structure is the most widely used. It is a network structure composed of several ordered layers of neurons connected in sequence without lateral inhibition. The first layer is accessible from outside through its input and outputs can be observed from the last layer. The information flows only in one direction.

In MFNN the model structure selection is basically dependent on two issues: Selecting the inputs to the network and selecting internal network architecture. An often used approach is to reuse the inputs from the linear models while letting the internal architecture be multilayer feedforward network. This approach has several attractive advantages.

- It is a natural extension of the well known linear model structures
- The internal architecture can be expanded gradually as a higher flexibility is needed to model more complex nonlinear relationships.

- The structural decisions required by the user are reduced to a level that is reasonable to handle.
- Suitable for design of control systems.

Nonlinear counterparts to the linear model structures (A1.1) are obtained by using the following predictor form

$$\hat{y}(t \mid q) = g\left[j\,(t), q\right] + e(t) \tag{5.1}$$

where *f (t)* is the regression vector while *?* is the vector containing the adjustable parameters in the neural network known as the weights. *g* is the function realized by the neural network and it is assumed to have a feedforward structure. Depending on the choice of regression different nonlinear model structures can be selected. The most common is the NNARX which is the acronym for Neural Network ARX. Figure 5.1 shows such a general structure of MFNN. The figure shows three layers but more layers are a direct generalization. The input layer has $n_i = n_y M + n_u N$ neurons, where M is the number of outputs, N is the number of inputs and n_y and n_u are the maximum lags at the input and output vector sequences respectively. The input to neural networks is then defined by Eq. 5.2.

$$\begin{aligned} \boldsymbol{X}(t) &= [u_1(t), u_2(t), \ldots, u_{n_i}(t)]^T \\ &= [\boldsymbol{Y}^T(t-1), \ldots, \boldsymbol{Y}^T(t-n_y), \boldsymbol{U}^T(t-1), \ldots, \boldsymbol{U}^T(t-n_u)]^T \end{aligned} \tag{5.2}$$

The input vector of the network consists of the past values of the network and output vector of the system. The input layer simply feeds the vector *X(t)* to the hidden layer without any modification. The hidden layer has user-defined n_h neurons with nonlinear transfer functions (such as sigmoid function). The output layer has M neurons, which

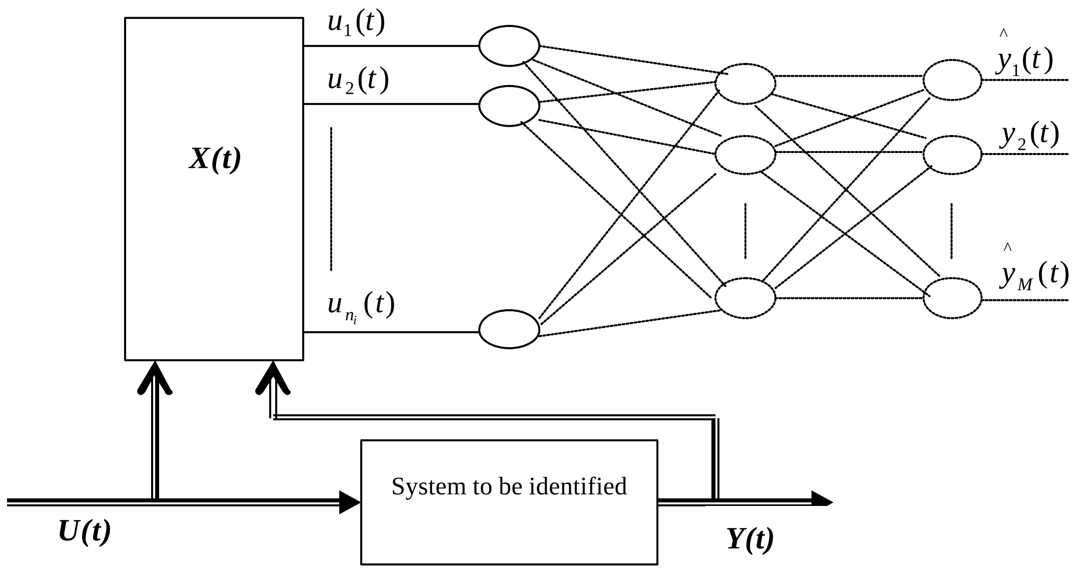

Figure 5.1 The MFNN for Nonlinear Identification

correspond to the M outputs of the system. The output of the network is represented as

$$\hat{y}_j^{(l)}(t) = g_k\left(\sum_{j=1}^{n_h} w_{ji}^{(l)} q_i^{(l-1)}(t) + b_j^{(l)}\right) \tag{5.3}$$

where W_{ij}^{l} is the synaptic weight of the neuron j in layer l that is fed from neuron i in layer $l-1$, $q_i(t)$ is the output signal of neuron i in the previous layer $l-1$, $b_j^{(l)}$ is the biases function of neuron j in layer l and $g_k(\bullet)$ is the activation function. The output vector provided by the network is defined in Eq. 5.4 and the error is defined as in Eq. 5.5.

$$\hat{\boldsymbol{Y}}(t) = [\hat{y}_1(t), \hat{y}_2(t), \ldots\ldots, \hat{y}_M(t)] \tag{5.4}$$

$$\boldsymbol{E}(t) = \boldsymbol{Y}(t) - \hat{\boldsymbol{Y}}(t) \tag{5.5}$$

The weights are updated by using backpropagation algorithm. It is expressed as in Eq. 5.6.

$$w_{ji}^{(l)}(t+1) = w_{ji}^{(l)}(t) + h d_j^{(l)}(t)\, q_i^{l-1}(t) \tag{5.6}$$

where h is the learning coefficient. The local gradients ds for the neuron j in output layer L and in hidden layer l are defined by Eq. 5.7 and 5.8 respectively.

$$d_j^{(l)}(t) = -2e_j^{L}(t)\left[g_j\left(\sum_{j=1}^{n_h} w_{ji}^{(L)}(t)\, q_j^{L-1}(t) + b_j^{(L)}(t)\right)\right]' \tag{5.7}$$

$$d_j^{(l)}(t) = \left[f_j\left(\sum_{i=1}^{n_i} W_{ji}^{l}(t) q_j^{l-1}(t) + b_i^{(1)}(t)\right)\right]' \sum_{k=1}^{M} d_k^{(l+1)}(t) W_{kj}^{(l+1)}(t) \tag{5.8}$$

The biases can be updated by using the following expression of Eq. 5.9.

$$b_j^{(l)}(t+1) = b_j^{(l)}(t) + d_j^{(l)} \qquad for\ h = 1,2 \tag{5.9}$$

5.5 Open Loop Identification

MFNN is now used to model the Demethanizer column process. One month data is collected at sampling time of 1 minute. For the purpose of training the network, 70% of the data samples are used. The rest of the 30% data samples are kept for testing and validating the identified MFNN model. The maximum lags for the output and input vector sequences in Eq. 5.2 are selected as 2 and 1 respectively. One hidden layer with 10 neurons is used. Tangent sigmoid nonlinearity is selected as the activation function for these neurons. The results for training are presented in Figs. 5.2 - 5.6. The results are shown for 300 data samples for better display. The results demonstrate the accuracy by which the MFNN is trained. There is practically no mismatch between the two models. However, to check or validate the performance of the trained MFNN model, test data which has not been used previously in the training, is utilized. The results of the validation are shown in Figs. 5.6 - 5.9, where the actual data is shown by a solid line and the MFNN response is shown in dashed line. For all process outputs, no mismatch is seen and the trained MFNN model manages to reproduce the test data excellently.

Now that MFNN model is trained and validated on open loop process data from the Demathanizer column, the next logical step is to use it for MPC design. However, as mentioned earlier, a model that provides excellent performance in open loop may not be able to maintain the same when the loop is closed (feedback). For this reason, the next section deals with the validation of the trained MFNN model from closed loop field data such as to ascertain if this model is reliable for close loop operation.

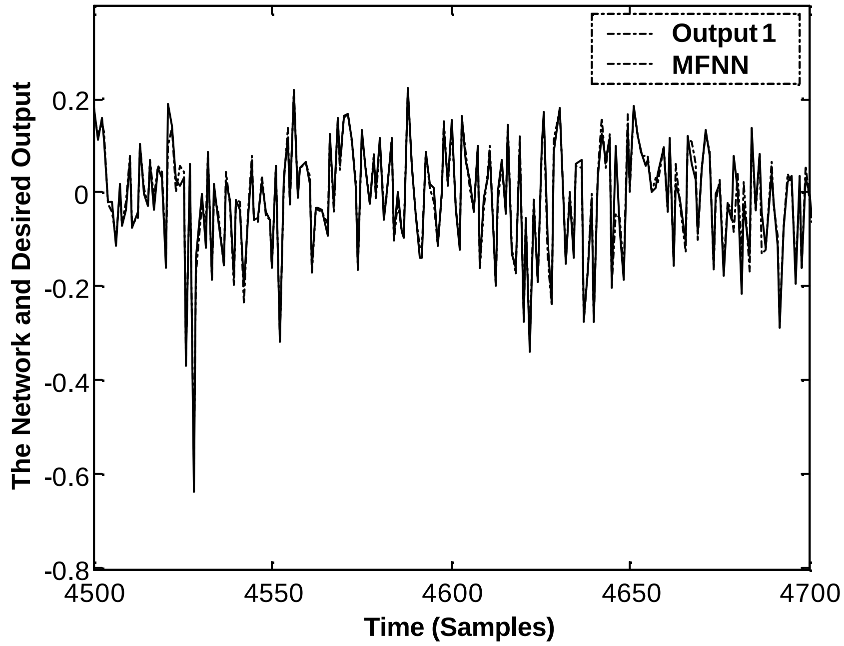

Figure 5.2 : Training MFNN – Output 1

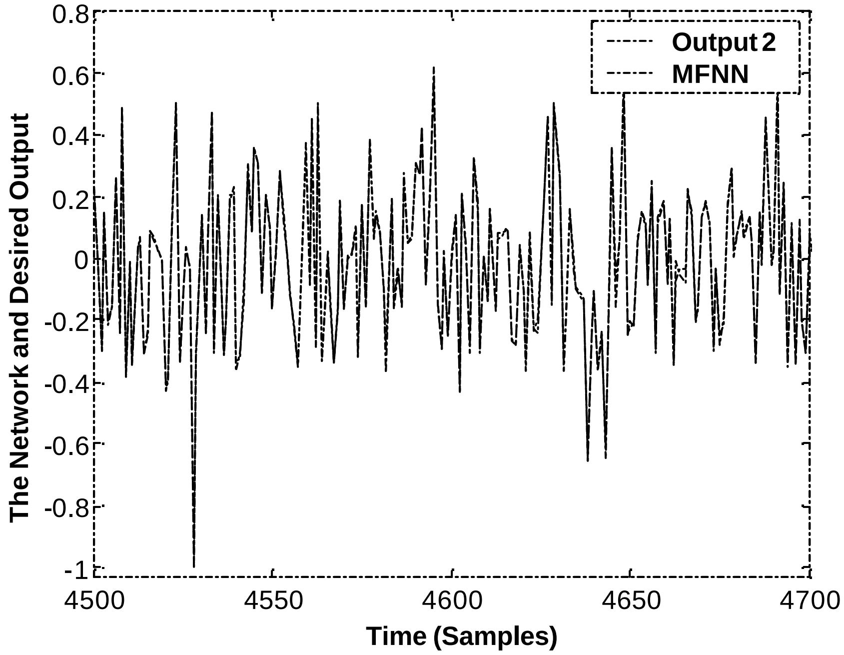

Figure 5.3 : Training MFNN – Output 2

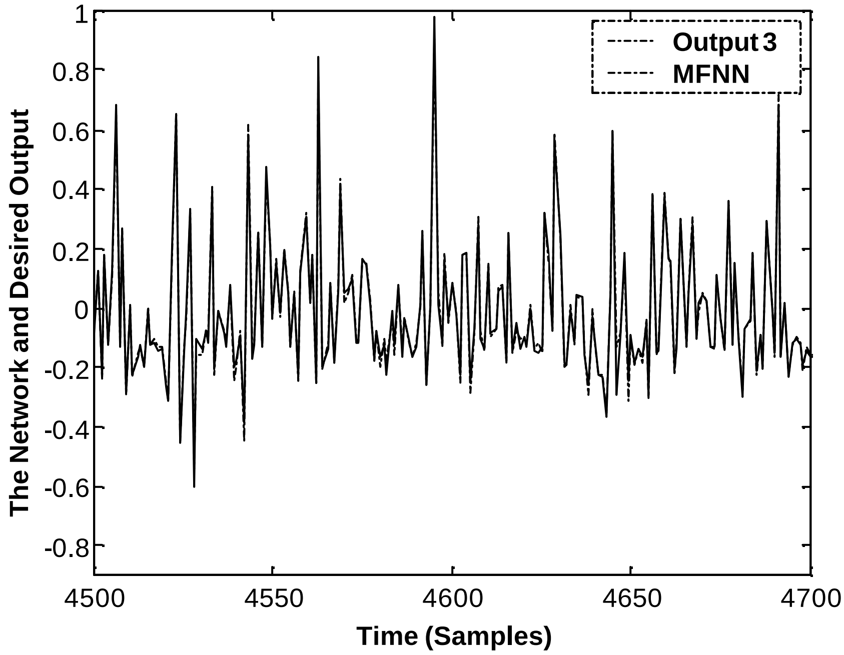

Figure 5.4 : Training MFNN - Output 3

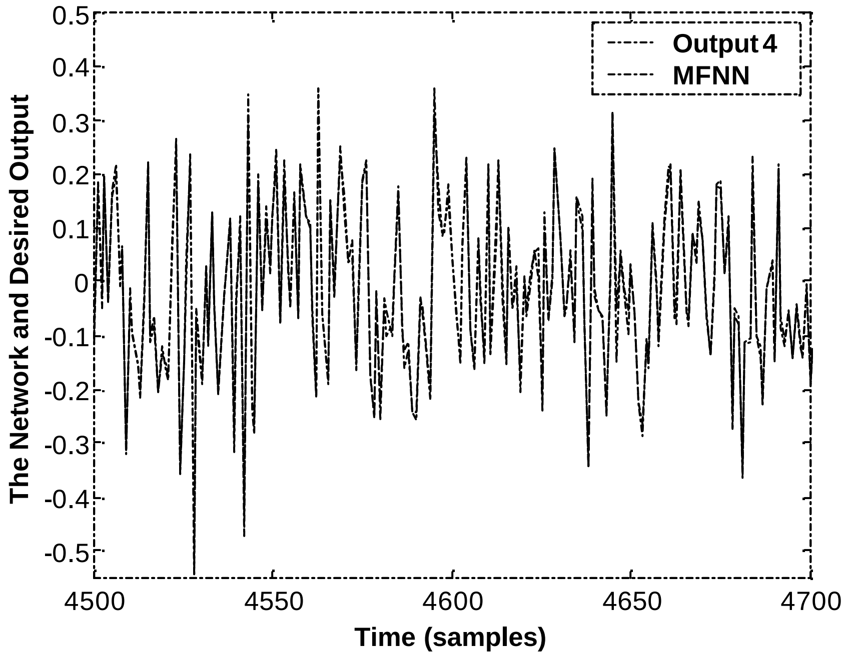

Figure 5.5 : Training MFNN – Output 4

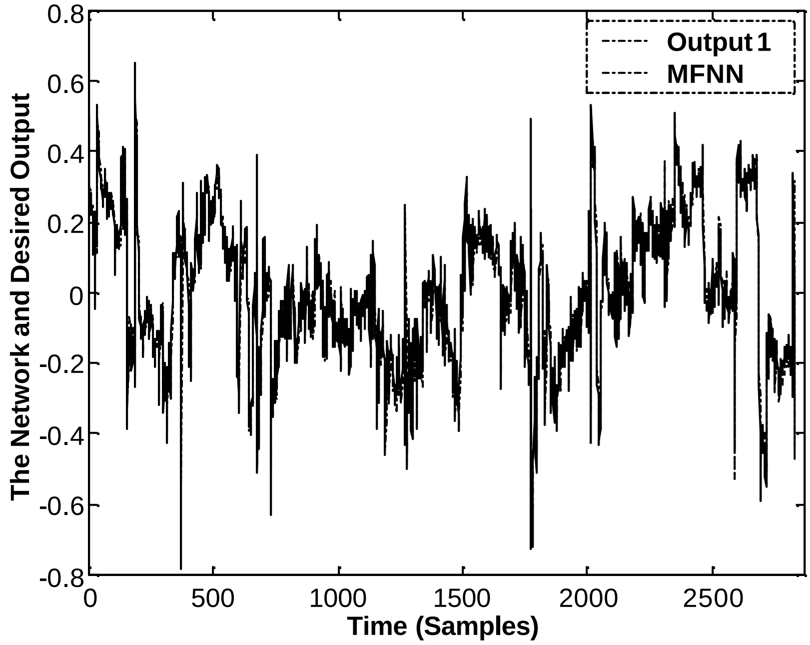

Figure 5.6 : MFNN and Desired Output 1 –Validation

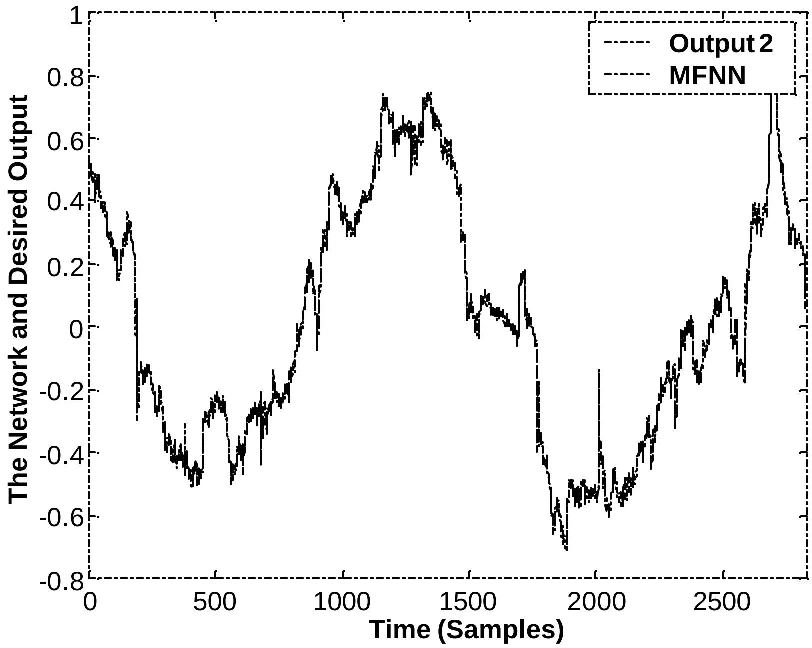

Figure 5.7 : MFNN and Desired Output 2 -Validation

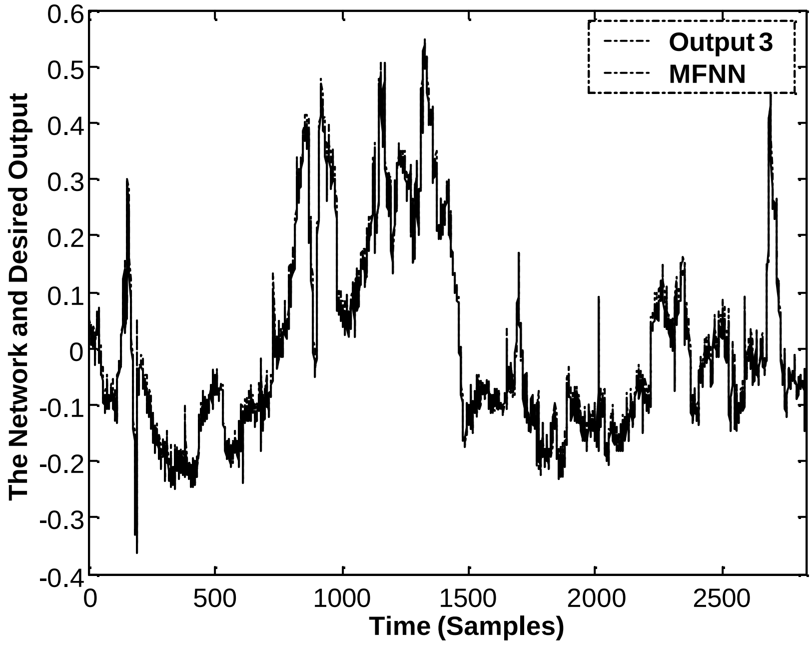

Figure 5.8 : MFNN and Desired Output 3 –Validation

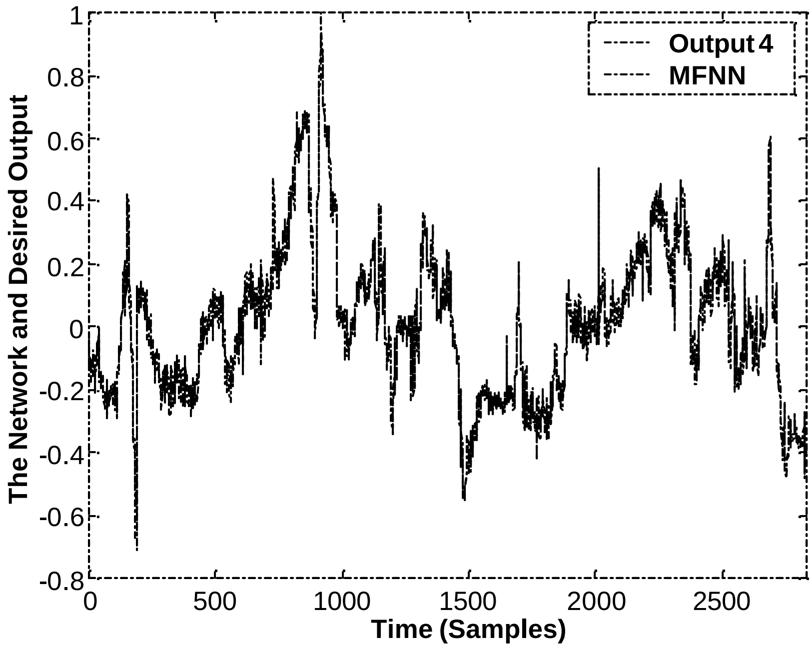

Figure 5.9 : MFNN and Desired Output 4 -Validation

5.6 Performance of MFNN on closed Loop Data

The trained MFNN model in now tested for its consistency and accuracy on closed loop field data collected from MPC process. This is done to ascertain the performance of this model in closed loop environment especially when MPC is running. According to Norgaard et al. [90], validation of a neural network model is highly dependent on its intended use. In this case the MFNN model is required to predict the future behavior of the plant in an MPC environment. This means that the MFNN model has to be consistent and its performance should not deteriorate with closed loop operation. Recall that the system input and outputs patterns can change dramatically after closing the loop. Keeping this in mind, the closed loop data collected from Demethanizer column process running with MPC (see Sec. 4.1), which is unfamiliar (totally fresh) to the trained model is now used to validate the performance of this model. The entire data is used for this purpose.

Figures 5.10 - 5.13 illustrate the excellent performance of the MFNN model. The response of MFNN model is shown in dashed line and the actual field data is shown by a solid line for all process outputs. The model trained with open loop data exhibits its versatility on closed loop data and manages to predict the correct behavior of the process. This is an essential requirement, as the model is going to be used in MPC scheme where it is required to predict the future behavior of the process in order to determine the next control action. In Figs. 5.18 - 5.21 the error distributions associated with the performance of the neural network model are shown. The errors are very small and remain mostly between the ranges of ± 0.2.

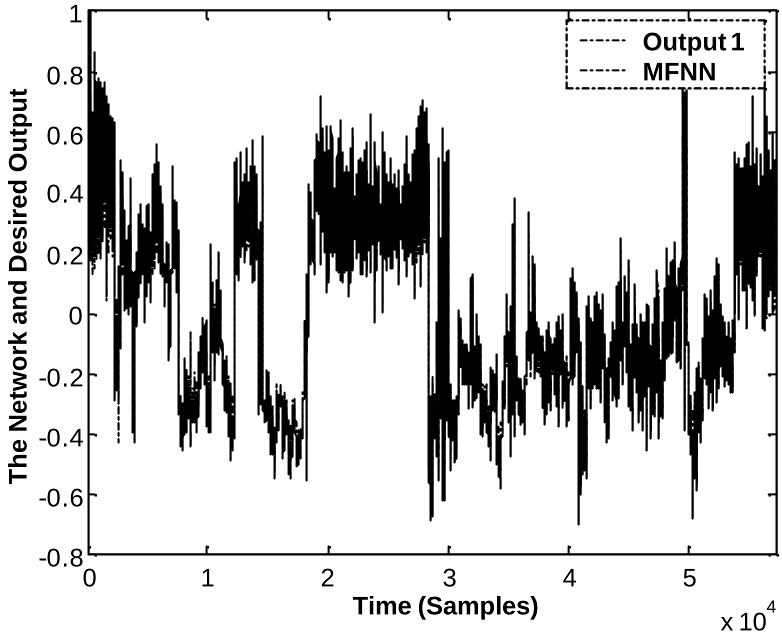

Figure 5.10 : MFNN and Desired Output 1 –Closed Loop Data

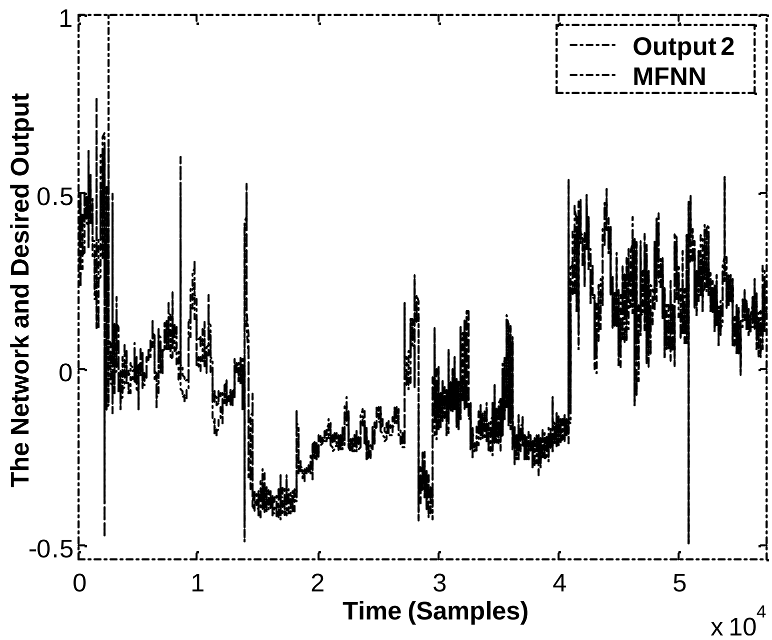

Figure 5.11 : MFNN and Desired Output 2 – Closed Loop Data

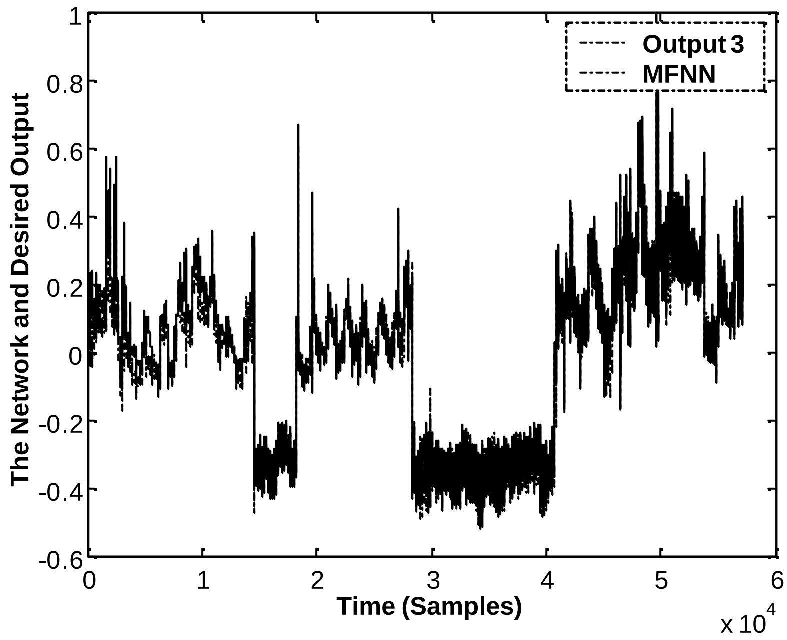

Figure 5.12 : MFNN and Desired Output 3 – Closed Loop Data

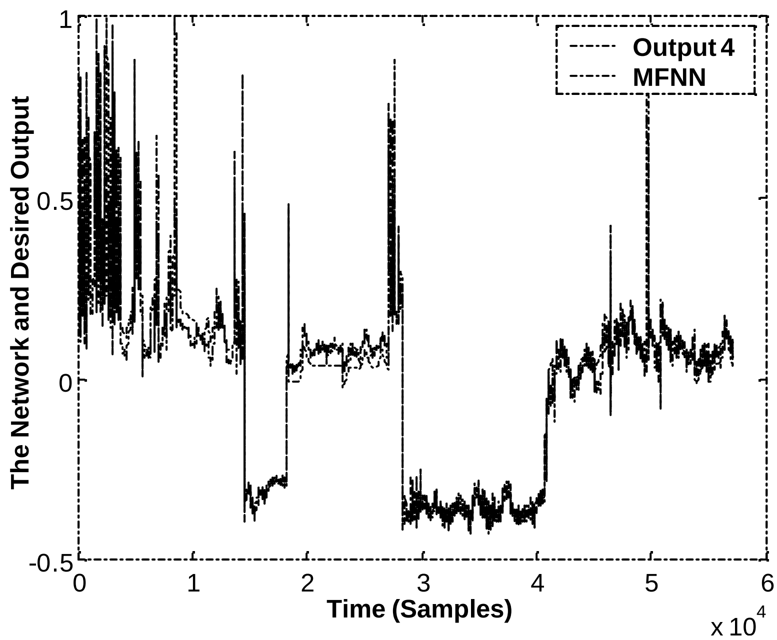

Figure 5.13 : MFNN and Desired Output 4 - Closed Loop Data

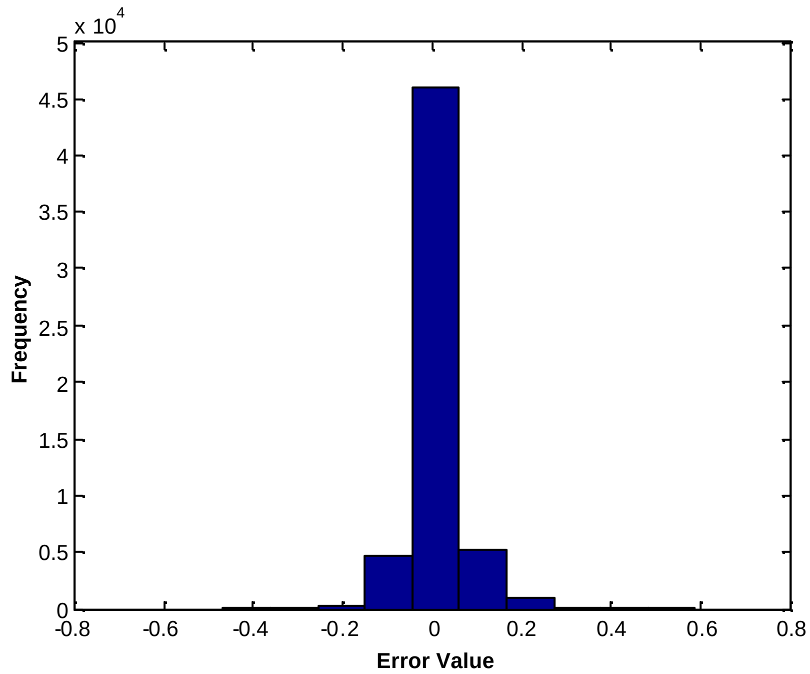

Figure 5.14 : Error Distribution (Output 1)

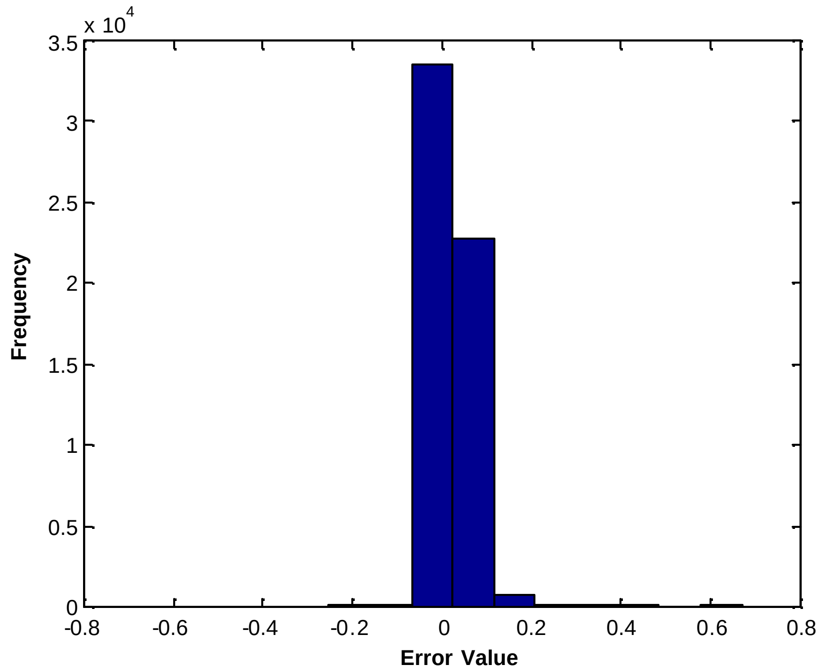

Figure 5.15 : Error Distribution (Output 2)

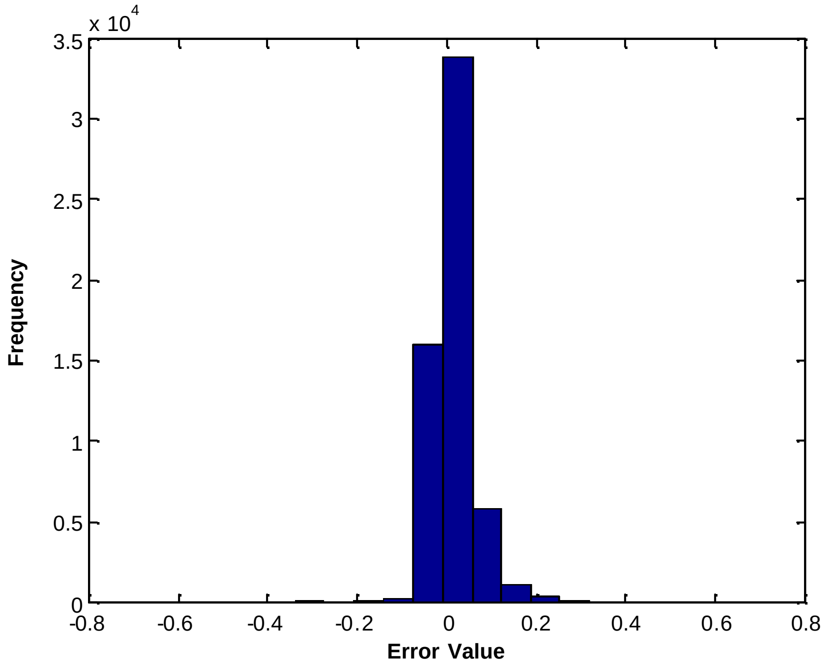

Figure 5.16 : Error Distribution (Output 3)

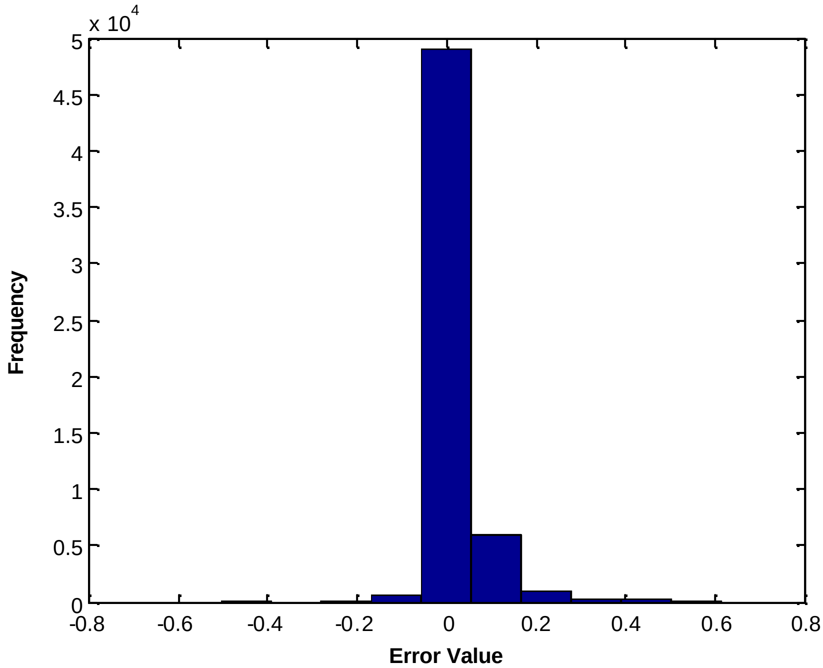

Figure 5.17 : Error Distribution (Output 4)

5.7 Chapter Summary

The idea of modeling a particular behavior using neural networks is attractive for several reasons. Neural networks are computing systems characterized by the ability to learn from examples rather than having to be programmed in a conventional sense. Their use enables the behavior of complex systems to be modeled and predicted through training without *a priori* information about the systems structures or parameters.

This chapter has dealt with this issue thoroughly and has shown that a trained (identified) MFNN model is a suitable candidate for MPC scheme. The model is identified with open loop data and its massive prediction capability and richness has been tested on closed loop data. This model certainly represents a good choice for use in the design of MPC.

Chapter 6

Conclusions and Future Work

In this chapter, some concluding remarks on this work are presented and the main contributions of this thesis are highlighted.

6.1 Concluding Remarks

The importance of plant model identification in closed loop operation has enhanced in recent years. For the purpose of model based controller design, closed loop identification offers a number of advantages such as better models, controller maintenance and validation.

MPC applications in industry involve dozens of inputs and outputs. To determine such a multivariable model from a given data puts an unprecedented demand on model identification and estimation techniques. To deal with this predicament, direct closed loop identification method is implemented for MPC applications in this thesis. Different identification techniques based on ARMA, state space and neural network modeling schemes are analyzed in this regard. Their performance is examined on both simulated and field data. An industrial application is used for this purpose. It is shown through simulations that identification schemes involving linear ARX and ARMAX models give consistency and optimal accuracy, on both simulated and field data. The advantages of using these closed loop identification and modeling schemes amongst other things, encompass performance improvement, reduced identification costs, controller diagnosis, online retuning of models, performance monitoring and avoiding unnecessary plant shut down for maintenance.

It is also shown that a MFNN model with its massive parallelism and learning capabilities can offer a new promising direction towards MPC design. In this regard, simulation results have shown that a neural network model trained with open loop data can perform extremely well and retain its prediction capability even when the loop is closed.

6.2 Future Work

During the course of this thesis, it was found that future research can be directed towards the following areas:

- The state space framework provides a powerful tool for designing and analyzing MPC. Improvement of subspace (N4SID) algorithm for MPC closed loop relevant identification and model estimation would prove useful.

- Further work is required to design a robustly performing predictive controller based on these identified models.

- RNN based models have not been considered in this thesis. They can also be investigated for MPC relevant identification.

Appendix A

Least Squares and Prediction Error Methods

The family of methods that minimize the error between the predicted and the observed values of the output are called prediction error methods. Consider the general model structure of the from

$$y(t) = G(q,q)\,u(t) + H(q,q)\,e(t) \qquad \text{(A1.1)}$$

where G is the dynamics model and H the noise model. $u(t)$ and $y(t)$ are measured inputs and outputs respectively and $e(t)$ is an uncorrelated random sequence. The parameter vector *?* is confined to a subset of $\Re^d$, called $\boldsymbol{D}_d$ where d is the dimension of *?*

$$q \in \boldsymbol{D}_d \subset \Re^d \qquad \text{(A1.2)}$$

The set of models in which the estimation procedure will search for the best model is determined by Eqs. A1.1 and A1.2. The one step ahead predictor for the model structure in Eq. A 1.1 is

$$\hat{y}(t\,|\,q) = H^{-1}(q,q)\,G(q,q)\,u(t) + (I - H^{-1}(q,q))\,y(t) \qquad \text{(A1.3)}$$

$\hat{y}(t\,|\,q)$ denotes a prediction of $y(t)$ given the data up to and including time $t-1$ and based on the model parameter vector ?. The prediction errors are

$$e(t,q) = y(t) - \hat{y}(t,q) = H^{-1}(q,q)\left(y(t) - G(q,q)\,u(t)\right) \qquad \text{(A1.4)}$$

Given the model of Eq. A1.3 and measured data Z^N, the prediction error estimate is determined through

$$\hat{q}_N = \arg\min_{q \in D_d} V_N(q, Z^N) \qquad \text{(A1.5)}$$

by minimizing the following criterion function

$$V_N(q, Z^N) = \frac{1}{N}\sum_{t=1}^{N} e^2(t,q) \qquad \text{(A1.6)}$$

For predictors that are linear in the data a closed form solution (least squares method) can be found. In other cases nonlinear search algorithms (prediction error method) are required to find a solution.

A.1 Least Squares Estimate

If the predictor is a linear function of the unknown parameters then the model in Eq. 1.1 can be expressed as:

$$y(t,q) = j^T(t)q + e(t) \qquad \text{(A1.7)}$$

Here f is an n-vector of regressors, the regression vector and q is an n-vector of unknown parameters. The prediction error becomes

$$e(t,q) = y(t) - j^{T}(t)q \qquad \text{(A1.8)}$$

and the criterion function resulting from Eq. A1.6 is

$$V_N(q, Z^N) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - j^{T}(t)q)^2 \qquad \text{(A1.9)}$$

This is the least squares criterion for the linear regression of Eq. A1.7. The name 'equation error method' also appears in the literature. The unique feature of this criterion, developed from the linear parameterization and the quadratic criterion is that it is a quadratic function in ?. Therefore, it can be minimized analytically, which gives, provided the inverse exists, the least squares estimate (LSE).

$$q_N^{LS} = \arg\min\ V_N(q, Z^N) = \left[\frac{1}{N}\sum_{t=1}^{N} j^{T}(t)\, j(t)\right]^{-1} \frac{1}{N}\sum_{t=1}^{N} j(t)\, y(t) \qquad \text{(A1.10)}$$

The Eq. A1.10 can also be expressed as

$$\hat{q} = (F^T F)^{-1}\, F\, Y \qquad \text{(A1.11)}$$

where

$$F = \begin{pmatrix} j^{T}(1) \\ j^{T}(2) \\ \vdots \\ j^{T}(N) \end{pmatrix}, \text{ an } (N \mid n) \text{ matrix} \qquad \text{(A1.12)}$$

$$Y = \begin{pmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{pmatrix}, \text{ an } (N \mid 1) \text{ matrix} \qquad \text{(A1.13)}$$

Least squares estimate method is also known as a special case of the prediction error (identification) method. This method holds for FIR and ARX models only.

In the case of ARMAX and OE models, a numerical search algorithm is required to find the parameter estimates $\hat{q}$ that minimizes V_N *(?)*. Guass-Newton algorithm is commonly used for this purpose. This algorithm is briefly discussed in the following sub-section.

A.2 Gauss-Newton Algorithm (Prediction Error Method)

In general the criterion function of Eq. A1.9 cannot be minimized by analytical methods. Least squares estimate method does not give consistent estimates if the noise is not a sequence of independent and identically distributed random variables (white) in Eq A1.1. In this case, methods for numerical minimization of the function *V(?)*, update the estimate of the minimizing point iteratively. This is usually done according to

$$\hat{q}^{(i+1)} = \hat{q}^{(i)} + a \ f^{(i)} \tag{A1.12}$$

where $f^{(i)}$ is a search direction based on information about *V(?)* acquired at previous iterations, and a is a positive constant such that an appropriate decrease in the value of *V(?)* is obtained. Normally the correction in Eq. A1.12 is chosen in the *Newton direction* [36]:

$$f^{(i)} = -\left[V_N''(\hat{q}^{(i)})\right]^{-1} V_N'(\hat{q}^{(i)}) \tag{A1.13}$$

Here $\hat{q}^{(i)}$ denotes the ith iteration point in the search The criterion of Eq. A1.6 has the gradient (Soderstrom et. al. [7])

$$V_N'(q, Z^N) = -\frac{1}{N}\sum_{t=1}^{N} y(t,q)\, e(t,q) \tag{A1.14}$$

where the notation

$$y(t,q) = -\left(\frac{\partial e(t,q)}{\partial q}\right)^T \tag{A1.15}$$

and

$$V_N''(q, Z^N) = \frac{1}{N}\sum_{t=1}^{N} y(t,q)\, y^T(t,q) - \frac{1}{N}\sum_{t=1}^{N} y'(t,q)\, e(t,q) \tag{A1.16}$$

At the global minimum point $e(t,q$ becomes asymptotically (as N $\rightarrow \infty$) white noise $(e(t,q_0) = e_0(t))$ which is independent of $y(t,q)$. Then

$$V_N''(q, Z^N) \approx \frac{1}{N}\sum_{t=1}^{N} y(t,q)\, y^T(t,q) \tag{A1.17}$$

It is appealing to neglect the second term in Eq. A1.16 for two reasons. First is that by construction $V_N''(q)$ is guaranteed to be positive definite. Therefore the loss function will decrease in every iteration if a is chosen appropriately. Second, the computations are simpler. The algorithm obtained in this manner is written as

$$\hat{q}^{(i+1)} = \hat{q}^{(i)} + a\left[\sum_{t=1}^{N} y(t,\hat{q}^{(i)})\, y^T(t,\hat{q}^{(i)})\right]^{-1}\left[\sum_{t=1}^{N} y(t,\hat{q}^{(i)})\, e(t,\hat{q}^{(i)})\right] \tag{A1.18}$$

This is called Gauss-Newton algorithm and is generally referred to as the prediction error method.

Appendix B

Sub Space Identification Method

Subspace identification aims at constructing state space models from input-output data. In this method first the (Kalman filter) states are estimated directly (either implicitly or explicitly) from input-output data, then the system matrices are obtained. In the model in Eqs. B1.1 and B1.2, $u_k \in \Re^m$ is the input, $x_k \in \Re^n$ is the state and $y_k \in \Re^l$ is the output. w_k and v_k are zero mean, white noise sequences.

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{B1.1}$$

$$y_k = Cx_k + Du_k + v_k \tag{B1.2}$$

The main steps in N4SID algorithm are the following [65]:

- Determine the model order n and a state sequence estimates $\hat{x}_i \quad \hat{x}_{i+1} \quad \cdots \quad \hat{x}_{i+j}$. They are found by first projecting row spaces of data block Hankel matrices, and then applying a singular value decomposition.
- Solve a least squares problem to obtain the state space matrices A, B, C, and D.

B.1 Notation

In this section, some notations are introduced. In Section 1.1, the notations for the data block Hankel matrices and in Section 1.2 for the system related matrices are presented.

B.1.1 Block Hankel Matrices and State Sequences

Block Hankel matrices with output and/or input data play an important role in subspace identification algorithms. These matrices can be easily constructed from the given input-output data. Input block Hankel matrices are defined as

$$U_{0|i-1} = \begin{pmatrix} u_0 & u_1 & u_2 & & u_{j-1} \\ u_1 & u_2 & u_3 & \cdots & u_j \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ u_{i-1} & u_i & u_{i+1} & \cdots & u_{i+j-2} \end{pmatrix} \in \Re^{mi \times j} \tag{B1.3}$$

The number of block rows (i) is selected as larger than the maximum order i.e. $i > n$. The number of columns (j) is typically equal to s-$2i$+1, which implies that all s available data samples are used. In any case, j should be larger than $2i$-1.

From here on the following input matrices notations are used

$$U_p = U_{0|i-1} \quad , \quad U_f = U_{i|2i-1} \tag{B1.4}$$

Here, the subscript 'p' refers to 'past', 'f' refers to future. The matrices U_p^+ and U_f^- on the other hand are defined by shifting the border between past and future one block row down. They are defined as $U_p^+ = U_{0|i}$ and $U_f^- = U_{i+1|2i-1}$. Similar definitions hold for the block Hankel matrices with the output vectors, which will denoted by Y_p and Y_f.

For prediction purposes a combination of input and output are used as regressors and defined as

$$W_p = W_{0|i-1} = \begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} \in \Re^{(m+l)\times j} \tag{B1.5}$$

State sequences play an important role in the derivation and interpretation of subspace identification algorithms. The state sequence X_i is defined as:

$$X_i \overset{def}{=} \begin{pmatrix} x_i & x_{i+1} & \cdots & x_{i+j-2} & x_{i+j-1} \end{pmatrix} \in \Re^{n\times j} \tag{B1.6}$$

where the subscript *i* denotes the subscript of the first element of the state sequence.

B.1.2 Model Matrices

Subspace identification algorithm makes extensive use of the extended observability matrix Γ_i which is defined as:

$$\Gamma_i = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{i-1} \end{pmatrix} \in \Re^{l\,i\times n} \tag{B1.7}$$

It is assumed that $\{A,C\}$ are observable, which implies that the rank of Γ_i is equal to n.

B.1.3 Geometric Tools

In section 2.1 through 2.2 the main geometric tools used to reveal some system characteristics are introduced. They are described from a linear algebra point of view, independent of the subspace identification framework which will be discussed in the next sections.

In the following sections it is assumed that the matrices $A \in \Re^{p \times j}$, $B \in \Re^{q \times j}$ and $C \in \Re^{p \times j}$ are given (they are dummy matrices in this section). It is also assumed that $j \geq \max(p, q, r)$, which will always be the case in the identification algorithm.

B.1.3.1 Orthogonal Projections

The orthogonal projection of the row space of A into the row space of B is denoted by *A/B* and its matrix representation is

$$A/B = A.\prod_B = AB^T\ (BB^T)^{\dagger}\ B \tag{B1.8}$$

where $\bullet^{\dagger}$ denotes the Moor-Penrose pseudo-inverse of the matrix and $\prod_B$ denotes the operator that projects the row space of a matrix onto the row space of the matrix B. Similarly $A/B^{\perp}$ is short hand for the projection of the row space of A onto $B^{\perp}$, the orthogonal complement of the row space of B:

$$A/B^{\perp} = A.\prod_B = A - A/B = A(I_j - \prod_{\boldsymbol{B}}) \tag{B1.9}$$

The combination of the projections $\prod_B$ and $\prod_{B^{\perp}}$ decomposes a matrix A into two matrices, the row spaces of which are orthogonal:

$$A = A\prod_B + A\prod_{B^{\perp}} \tag{B1.10}$$

The matrix representation of these projections can be easily computed using RQ decomposition of $\begin{pmatrix} B \\ A \end{pmatrix}$, which is the numerical matrix version of the Gram-Schmidt orthogonalization procedure. Let A and B be matrices of full rank and let RQ decomposition of $\begin{pmatrix} B \\ A \end{pmatrix}$ be denoted by

$$\begin{pmatrix} B \\ A \end{pmatrix} = RQ^T = \begin{pmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} \tag{B1.11}$$

where $R \in \Re^{(p+q)\times(p+d)}$ is lower triangular, with $R_{11} \in \Re^{q\times q}$, $R_{21} \in \Re^{p\times q}$, $R_{22} \in \Re^{p\times p}$ and $Q \in \Re^{j\times(p+q)}$ are orthogonal i.e. $Q^T Q = \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} = \begin{pmatrix} I_q & 0 \\ 0 & I_p \end{pmatrix}$. Then, the matrix representations of the orthogonal projections can be written as

$$A/B = R_{21}\, Q_1^T \tag{B1.12}$$

$$A/B^{\perp} = R_{22}\, Q_2^T \tag{B1.13}$$

B.1.3.2 Oblique Projections

A matrix A can also be decomposed as a linear combination of the rows of two non-orthogonal matrices B and C and of the orthogonal complement of B and C. This can be written as:

$$A = R_B B + R_C C + R_{B^{\perp}, C^{\perp}} \begin{pmatrix} B \\ C \end{pmatrix}^{\perp} \tag{B1.14}$$

The matrix $R_C C$ is defined as the oblique projection of row space of A along the row space of B into the row space of C:

$$A/_B C = R_C C \tag{B1.15}$$

The oblique projection can also be interpreted through the following recipe: project the row space of A orthogonally into the joint row space of B and C and decompose the result along the row space of B and C.

If the RQ decomposition of $\begin{pmatrix} B \\ C \\ A \end{pmatrix}$ is given by Eq. B1.17,

$$\begin{pmatrix} B \\ C \\ A \end{pmatrix} = \begin{pmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \\ R_{31} & R_{32} & R_{33} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \end{pmatrix} \tag{B1.16}$$

then the matrix representation of the orthogonal projection of the row space of A onto the joint row space of B and C is equal to (see previous section):

$$A / \begin{pmatrix} B \\ C \end{pmatrix} = \begin{pmatrix} R_{31} & R_{31} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} \tag{B1.17}$$

This can also be written as linear combination of the rows of B and C:

$$A / \begin{pmatrix} B \\ C \end{pmatrix} = \begin{pmatrix} R_B B & R_C C \end{pmatrix} = \begin{pmatrix} R_B & R_C \end{pmatrix} \begin{pmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} \tag{B1.18}$$

The oblique projection of the row space of A along the row space of B onto the row space of C can thus be computed as

$$A /_B C = R_C C = R_{32} R_{22}^{-1} \begin{pmatrix} R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} \tag{B1.19}$$

B.2 Subspace Identification Algorithm (N4SID)

In this section, N4SID algorithm for the identification of A, B, C, D, Q, R and S matrices is presented. The algorithm works in two main steps. First the row space of Kalman filter state sequence is obtained directly from input output data, without any knowledge of the system matrices. This is explained in Section 3.1. In the second step, which is given in

Section 3.2, the system matrices are extracted from the state sequence via a least squares problem.

B.2.1 Calculation of a State Sequence

The state sequence of a combined deterministic–stochastic model can again be obtained from input output data in two steps. First, the future output row space is projected along the future input row space into the joint row space of past input and past output. Singular value decomposition is carried out to obtain the model order, the observability matrix and a state sequence, which has a very precise and specific interpretation.

B.2.1.1 Oblique projection

RQ decomposition is used to compute the oblique projection $Y_f /_{U_f} \begin{pmatrix} U_p \\ Y_p \end{pmatrix}$. Let $U_{0|2i-1}$ be the $2mi$ x j and $Y_{0|2i-1}$ the $2li$ x j block Hankel matrices of the input and output observations. Then the RQ decomposition of $\begin{pmatrix} U \\ Y \end{pmatrix}$ is partitioned as follows:

$$\begin{pmatrix} U_{0|i-1} \\ U_{i|i} \\ U_{i+1|2i-1} \\ Y_{0|i-1} \\ Y_{i|i} \\ Y_{i+1|2i-1} \end{pmatrix} = \begin{pmatrix} R_{11} & 0 & 0 & 0 & 0 & 0 \\ R_{21} & R_{22} & 0 & 0 & 0 & 0 \\ R_{31} & R_{32} & R_{33} & 0 & 0 & 0 \\ R_{41} & R_{42} & R_{43} & R_{44} & 0 & 0 \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} & 0 \\ R_{61} & R_{62} & R_{63} & R_{64} & R_{65} & R_{66} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \\ Q_4^T \\ Q_5^T \\ Q_6^T \end{pmatrix} \tag{B1.20}$$

The matrix representation of the oblique projection $Y_f /_{U_f} \begin{pmatrix} U_p \\ Y_p \end{pmatrix}$ of the future output row space along the future input row space onto the joint space of past input and past output, is denoted by o_i and is obtained as follows (see section B1.3.2):

$$o_i = Y_f \Big/_{U_f} \begin{pmatrix} U_p \\ Y_p \end{pmatrix} = R_{U_p} R_{11} Q_1^T + R_{Y_p} (R_{41}\ R_{42}\ R_{43}\ R_{44}) \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \\ Q_4^T \end{pmatrix} \tag{B1.21}$$

where,

$$\begin{pmatrix} R_{U_p} & R_{U_f} & R_{Y_p} \end{pmatrix} \left(\begin{array}{c|cc|c} R_{11} & 0 & 0 & 0 \\ \hline R_{21} & R_{22} & 0 & 0 \\ R_{31} & R_{32} & R_{33} & 0 \\ \hline R_{41} & R_{42} & R_{43} & R_{44} \end{array}\right) = \left(\begin{array}{c|cc|c} R_{51} & R_{52} & R_{53} & R_{54} \\ R_{61} & R_{62} & R_{63} & R_{64} \end{array}\right) \tag{B1.22}$$

from which R_{U_p}, R_{U_f}, and R_{Y_p} can be calculated. The oblique projection $Y_f^- \Big/_{U_f^-} \begin{pmatrix} U_p^+ \\ Y_p^+ \end{pmatrix}$, denoted by o_{i-1}, on the other hand, is equal to

$$o_{i-1} = R_{U_p^+} \begin{pmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix} + R_{Y_p^+} \begin{pmatrix} R_{41} & R_{42} & R_{43} & R_{44} & 0 \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \\ Q_4^T \\ Q_5^T \end{pmatrix} \tag{B1.23}$$

where,

$$\begin{pmatrix} R_{U_p^+} & R_{U_p^+} & R_{Y_p^+} \end{pmatrix} \left(\begin{array}{cc|c|cc} R_{11} & 0 & 0 & 0 & 0 \\ R_{21} & R_{22} & 0 & 0 & 0 \\ \hline R_{31} & R_{32} & R_{33} & 0 & 0 \\ \hline R_{41} & R_{42} & R_{43} & R_{44} & 0 \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} \end{array}\right) = \left(\begin{array}{cc|c|cc} R_{61} & R_{62} & R_{63} & R_{64} & R_{65} \end{array}\right) \tag{B1.24}$$

Under the assumption that:

- The process noise w_k and measurement noise v_k are uncorrelated with input u_k
- The input u_k is persistently exciting of order $2i$

- The number of available data is large, so that $j \to \infty$

It can be shown [65] that the oblique projection o_i is equal to the product of the extended observability matrix Γ_i and a sequence of Kalman filter state.

$$o_i = \Gamma_i \, \tilde{X}_i \tag{B1.25}$$

Similarly the oblique projection o_{i-1} is equal to

$$o_{i-1} = \Gamma_{i-1} \, \tilde{X}_{i+!} \tag{B1.26}$$

B.2.1.2 Singular value decomposition

The singular value decomposition of $R_{U_p} \, (R_{11} \quad 0 \quad 0 \quad 0) + R_{Y_p} \, (R_{41} \quad R_{42} \quad R_{43} \quad R_{44})$ is equal to:

$$\begin{aligned} R_{U_p} \, (R_{11} \quad 0 \quad 0 \quad 0) + R_{Y_p} \, (R_{41} \quad R_{42} \quad R_{43} \quad R_{44}) &= \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \\ &= U_1 \, S_1 \, V_1^T \end{aligned} \tag{B1.27}$$

where $U_1 \in \Re^{li \times n}$, $S_1 \in \Re^{n \times n}$ and $V_1 \in \Re^{li \times n}$. Then the order of the system of Eq. B1.1 is equal to the number of singular values in Eq. B1.27. The extended observability matrix Γ_i is chosen as:

$$\Gamma_i = U_1 \, S_1^{1/2} \tag{B1.28}$$

and the state sequence $\tilde{X}_i$ is equal to:

$$\tilde{X}_i = (\Gamma_i)^{\dagger} \boldsymbol{o}_i = S_1^{1/2} V_1^T \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \\ Q_4^T \end{pmatrix} \tag{B1.29}$$

The ‘shifted’ state sequence $\tilde{X}_{i+1}$, on the hand can be obtained as

$$\tilde{X}_{i+1} = (\underline{\Gamma_i})^{\dagger} \boldsymbol{o}_{i-1} \tag{B1.30}$$

where $\underline{\Gamma_i} = \Gamma_{i-1}$ denotes the matrix $\underline{\Gamma_i}$ without the last l rows.

B.2.2 Computing the System Matrices

From previous section, the following information has been found.

- The order of the system from inspection of the singular values of Eq. B1.27
- The extended observability matrix Γ_i from Eq. B1.28 and the matrix Γ_{i-1} as $\underline{\Gamma_i}$ which denotes the matrix Γ_i without the last l rows.
- The state sequences $\tilde{X}_i$ and $\tilde{X}_{i+1}$.

The state space matrices *A*, *B*, *C* and *D* can now be found by solving a set of over-determined equations in a least squares sense:

$$\begin{pmatrix} \tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} \hat{A} & \hat{B} \\ \hat{C} & \hat{D} \end{pmatrix} \begin{pmatrix} \tilde{X} \\ U_{i|i} \end{pmatrix} \tag{B1.31}$$

This is the N4SID algorithm commonly used in state space modeling schemes.

Nomenclature

Abbreviations

ARMA	AutoRegressive Moving Average
ARX	AutoRegressive with eXternal input
ARMAX	AutoRegressive Moving Average with eXternal input
CV	Controlled Variable
DMC	Dynamic Matrix Control
MV	Manipulated Variable
MFNN	Multilayer Feedforward Neural Network
MIMO	Multi-Input Multi-Output
MPC	Model Predicative Control
N4SID	Numerical algorithm for Subspace State Space System Identification
NNARX	Neural Network AutoRegressive with eXternal input
OE	Output Error
PEM	Prediction error method
RBFNN	Radial Basis Function Neural Network
RNN	Recurrent Neural Network
SISO	Single-Input Single-Output

Notations

$A(q^{-1})$	The polynomial for the poles of the system
d_1	Measure disturbance 1 (Ambient Temperature)
d_2	Measure disturbance 2 (Feed Compressor Discharge Pressure)

$\boldsymbol{D}_d$	The set of values over which q ranges
$e(t)$	White noise sequence with zero mean
$E(t)$	The error vector for MIMO system
E	Mathematical expectation
$f^{(i)}$	Search direction at iteration i
$g_k(\bullet)$	Activation function
$G(q^{-1},q)$	Transfer function in a model set, corresponding to parameter value q
H_p	Prediction horizon
H_c	Control horizon
N	Available data samples
o_{i-1}	Oblique projection of the row space of $Y_{i\|2i-1}$ along the row space of $U_{i\|2i-1}$ on the row space of $W_{0\|i-1}$
q^{-1}	Backward shift operator $q^{-1}x(t) = x(t-1)$
$q_i(t)$	Output signal of neuron i
Q	Tracking error weighting matrix
$r(t)$	Set point variable at time t
R	Control move penalty weight matrix
$u(t)$	Input variable at time t
$u(k)$	Controller output and process input at sample time k
u_1	Input variable 1 (LP Residue Gas Pressure Set point)
u_2	Input variable 2 (Jump Over Valve Opening)
u_3	Input variable 3 (Trim Re-boiler Valve Opening)
u_4	Input variable 4 (Demethanizer Tray 6 Temperature)
$U_{0\|i-1}$	Input block Hankel matrix. The subscript indicates the indices of the first column of the matrix
U_p^+	Past inputs $U_{0\|i}$
v_k	Measurement noise
$V_N(q, Z^N)$	Criterion to be minimized
w_k	Process noise
w_{ji}	The weight of a connection between neurons i and j

$w(k)$	Reference trajectory at sample time k
$x(t)$	State vector at time t
X_i	State sequence.
$y(t)$	Output variable at time t
y_1	Output variable 1 denotes Bottom C1 over C2
y_2	Output variable 2 denotes LP Residue Gas Valve Opening
y_3	Output variable 3 denotes Demathanizer Pressure Differential
y_4	Output variable 4 denotes Tray 6 Bypass Valve Opening
$y(k)$	Process output at sample time k
$\hat{y}(t \mid q)$	Predicted output at time t, based on data samples Z^{t-1}

Greek Symbols

a	Learning rate
$b_j^{(l)}$	Basis function of neuron j in layer l
Γ_i	Extended observability matrix
$d_j^{(l)}$	Gradient of neuron j in layer l
$e(t,q)$	Prediction error
$j(t)$	Regression vector at time t
$y(t,q)$	Gradient of $e(t,q)$ with respect to q
h	Learning coefficient
q	Vector used to parameterize models
$\prod_B$	Operator projecting the row space of a matrix onto the row space of B

Bibliography

[1] Yucai Zhu. Multivariable Process Identification for MPC: The Asymptotic Method and its Applications. *Tai-Ji Control,* The Netherlands, 2001.

[2] Wei Xing Zheng. A Modified Method for Closed Loop Identification of Transfer Function Models. *IEEE,* 2002.

[3] Urban Forsell and Lenart Ljung. Closed-loop Identification Revisited. *IFAC Automatica,* Vol. 35 No. 7, pp 1215-1241, 1999.

[4] Gustavson, L. Ljung and T. Soderstrom. Identification of processes in closed loop – Identifiability and Accuracy Aspects. *Automatica,* 22:59-75, 1977.

[5] N. K. Sinha and B. Kuszta. Modeling and Identification of Dynamic Systems. *Van Nostrand Reinhold Company,* 1983.

[6] Lennart Ljung and T. Soderstrom. Theory and Practice of Recursive Identification. *MIT Press Series in Signal Processing, Optimization and control,* 1983.

[7] T. Soderstrom and P. Stoica. System Identification. *Prentice Hall International*, UK, 1989.

[8] J. W. Polderman and C. Praagman, The Closed Loop Identification Problem in Indirect Adaptive Control. *IEEE 28th Control Conference on Decision and Control,* Florida, December 1989.

[9] Ketao Liu and Robert Skelton. Closed-Loop Identificastion and Iterative Controller Design. *Proceeding of the 29th Conference on Decision and Control IEEE,* 1990.

[10] A. C. van der Klauw and M. Verhaegen. State Sapce Identification of Closed Loop Systems. *Proceedings of 30th Conference on Decision and Control IEEE,* 1991.

[11] Ruud J. P. Schrama. Accurate Identification for Control: The necessity of an Iterative Scheme. *IEEE Transactions on Automatic Control,* Vol. 37, 1992.

[12] Joseph C. Musto and L Ken Lauderbaugh. A Heuristic Algorithm for On Line System

Identification. *IEEE International Symposium on Intelligent Control*, 1991.

[13] A. Voda and I. D. Landau. Real-Time Evaluation of an Iterative Scheme for Closed Loop Identification and Control Design. *Proceedings of the American Control Conference Baltimore, IEEE,* 1994.

[14] A.C. van der Klauw, G.E. van Ingen, A. van Rhijn, S. Olivier, P.P.J. van den Bosch and R. A. de Callafon. Closed-loop Identification of a Distillation Column. *IEEE,* 1994.

[15] Paul M. J. Van Den Hof and Ruud J. P. Scharma. Identification and Control – Closed-loop Issues. *Automatica,* Vol. 31, No. 12 pp. 1751-1770, 1995.

[16] Ryszard Gessing and Marian B. Lachita. Set Point and Identifiability in the Closed Loop with a Minimum Variance Controller. *Proceedings of International Conference on Adaptive Control and Signal Processing.* 1995.

[17] Hakan Hjalmarsson, Michel Geverns and Franky De. Bruyne. For Model-based Control Design, Closed-loop Identification Gives better Performance. *Automatica,* Vol. 32, No.12 pp. 1659-1673, 1996.

[18] I.D. Landau and A. Karimi. Recursive Algorithms for Identification in Closed Loop-A Unified Approach and Evaluation. *Proceeding of the 35th Conference on Decision and control, Japan, IEEE,* 1996.

[19] Michel Geverns, Lennart Ljung and Paul M. J. Van den Hof. Asymptotic Variance Expressions for Closed –loop Identification and their Relevance in Identification for Control. *Progress report on research activities in the Mechanical Engineering Systems and Control groups, Delft University Press,* Vol. 9, 1996.

[20] Paul M. J. Van den Hof and Raymond A. de Callafon. Multivariable closed-loop identification: from indirect identification to dual-Youla parameterization. *IEEE Conference on Decision and Control,* 1996.

[21] Mario Milanese, Michele Taragna and Paul M. J. Van den Hof. Closed-loop Identification of Uncertainty Models for Robust Control Design: A Set Membership Approach. *Proceeding of the 36th Conference on Decision and control, USA, IEEE,* 1997.

[22] Lennart Ljung and Urban Forsell. Variance Results for Closed-loop Identification Methods. *Proceeding of the 36th Conference on Decision and control, USA, IEEE*, 1997.

[23] Lianming Sun, Wenjiang Liu and Akiro Sano. Over-sampling Approach to Closed-loop Identification. *Proceeding of the 36th Conference on Decision and control, USA, IEEE,* 1997.

[24] Natasha Linard, Brian D. O Anderson and Franky De Bruyne. Closed Loop

Identification Of Nonlinear Systems. *Proceeding of the 36th Conference on Decision and control, USA, IEEE*, 1997.

[25] Akihiko Yoneya, Yoshihiro Hashimoto, Yoshitaka Togari and Colin Mcgreavy. Iterative Closed Loop Identification using Linear Functional Method. *SICE*, 1997.

[26] Biao Huang and Shrish L. Shah. Closed-loop Identification: A Two Step Approach. *Journal of Process Control*, 7, 6, pp. 425-438, 1997.

[27] Michel Geverns, Brian D. O. Anderson and Benot Codorns. Issues in Modeling for Control. *Proceedings of the American Control Conference,* USA, 1998.

[28] Paul Van den Hof. Closed-Loop Issues in System Identification. *Annual Reviews in Control, Elsevier Science Ltd.* Vol. 22, pp 173-186, 1998.

[29] Franky De Bruyne, Brian D. O. Anderson, Michael Geverns and Natasha Linard. On Closed-Loop Identification with a Tailor-Made Parameterization. *Proceedings of the American Control Conference,* Philadelphia, 1998.

[30] G. G. Kulikov, R. D. Akhmetsafin, A. N. Fayzullin and T. V. Breikin. A Modeified Recurrent Least Squares Algoirhtm for On-Line Closed-Loop Identification for Process with Time Delay. *UKACC International Conference on Control*, IEE, 1998.

[31] Lianming Sun, Wenjiang Liu and Akiro Sano. Direct Closed-loop Identification of Unstable System by Output Inter-sampling Scheme. *Proceeding of the American Control Conference, USA, AACC*, 1999.

[32] C. T. Chou and M. Verhaegen. An Indirect Approach to Closed-loop Identification of Wiener Models. *Proceeding of the American Control Conference,* USA, 1999.

[33] Alex J. Pasadyn, S. Joe Qin and S. Valle-Cervantes. Closed-loop and Open-loop Identification of an Industrial Wastewater Reactor. *Proceeding of the American Control Conference,* USA, 1999.

[34] Urban Forssell. Closed-loop Identification Methods, Theory and Applications. *Thesis-Department of Electrical Engineering, Linkoping University,* Sweden, 1999.

[35] Lianming Sun, Wenjiang Liu and Akiro Sano. Direct Closed-loop Identification of Magnetic Suspension System. *Proceeding of the International Conference on Control, USA, IEEE*, 1999.

[36] Lennart Ljung. System Identification-Theory for the User. *Prentice Hall, USA,* 1999.

[37] I. D. Landau, B. D. O. Anderson and F. De Bruyne, Closed-loop Output Error Identification Algorithms for Nonlinear Plants. *IEEE Conference on Decision and Control,* Phoenix, Arizona, 1999.

[38] Peter Gaspar, Zoltan Szabo and Jozsef Bokor. Closed-Loop Identification Using Generalized Orthonormal Basis Functions. *Proceedings of the 38th IEEE Conference on Decision and Control,* Arizona, 1999.

[39] G. Jin, M. K. Sain and B. F. Spencer. Closed-loop Identification and Control Re-design: An Experimaental Structural Control Example. *8th ASCE Special Conference on Probabilistic Mechanics and Structural Reliability*, 2000.

[40] Wei Xing Zheng. Least-Squares Identification of Dynamic Systems in Closed Loop. *Proceedings of the 39th IEEE Conference on Design and Control, Australia,* 2000.

[41] Lianming Sun, Hiromitsu Ohmari and Akiro Sano. Direct Closed-Loop Identification Apporach to Unstable Plant. *Proceedings of the 39th IEEE Conference on Design and Control*, Australia, 2000.

[42] Urban Forsell and Lenart Ljung. A Projection Method for Closed-Loop Identification. *IEEE Transactions on Automatic Control,* Vol. 45, No. 11, 2000.

[43] Alexander Schwarm, S. Alper Eker and Michael Nikolaou. Model Predictive Control and Identification: A New Approach to Closed-Loop Identification and Adaptive Control. *Spectron (www.prediktor.no/SpektronInternal)*, 2000.

[44] Paul Van den Hof, Raymond de Callafon and Edwin van Donkelaar. CLOSID – A Closed-Loop System Identification Toolbox for Matlab. *Proceeding of the 12th IFAC Sympoium on system Identification,* SYSID, 2000.

[45] Su Ki Ooi and Erik Weyer. Closed Loop Identification of an Irrigation Channel. *Proceedings of the 40th IEEE Conference on Design and Control,* Australia, 2001.

[46] Wei Xing Zheng. An Efficient Method for Direct Closed-Loop Identification. *Proceedings of the 40th IEEE Conference on Decision and Control,* USA, 2001.

[47] Wei Xing Zheng, Parmetric Identification Of Linear Systems Operating Under Feedback Control. *IEEE Transactions on Circuits and Systems,* Vol. 48, No.4, April 2001.

[48] E. de Klerk and I. K. Craig. An Assignment to Teach Closed-Loop System Identification. *15th Triennial World Congress, IFAC,* 2002.

[49] L. Keviczky and Cs. Banyasz. Closed-Loop Identification of the K-B and d-Y Parameters. *IEEE Instrumentation and Measurement Technology Conference, Hungary,* 2001.

[50] L. Y. Wang and G. Yin. Time Complexity of Closed-Loop Identification. *Proceedings of the 40th IEEE Conference on Decision and Control*, Florida, 2001.

[51] S. Alper Eker and Michael Nikolaou. Simultaneous Model Predicitve Control and Identification: Closed Loop Properties. *Submitted to Automatica,* 2001.

[52] M. Leskens, L.B.M. Van Kessel and P.M.J. Van den Hof. MIMO Closed-loop Identification of an MSW Incinerator. *Control Engineering Practice,*Vol. 10, pp. 315-326, 2002.

[53] Tohru Katayama, Hidetoshi Kawauchi and Giorgio Picci. A Subspace Identification of Closed Loop System by Orthogonal Decomposition. *Proceedings of the 40th IEEE Conference on Decision and Control,* Florida, 2001.

[54] S. M. Veres. Synergy of Predictive Control and Identification. *IEE Two-Day Workshop on Techniques and Application,* 4/1- 4/3, 1999.

[55] J. Richalet, A. Rault, J. L. Testud and J. Papan. Model Predictive Heuristic Controller Applications to Industrial Processes. *Automatica* Vol. 14, pp. 413-428, 1978.

[56] C. R. Cutler and B. L. Ramaker. Dynamic Matrix Control – A Computer Control Algorithm. *Proceedings of Joint American Control Conference,* San Francisco, USA, 1980.

[57] R. Rouhani and R. K. Mehra. Model Algorithmic Control. *Automatica*, Vol. 18, pp. 401-414, 1982.

[58] N. R. Sripada and D. G. Fisher. Multivariable Optimal Constrained Control Algorithm: Part 1. Formulations and Application. *Proceedings of International Conference on Modeling and Control*, Hangzhou, China, 1985.

[59] D. W. Clarke, C. Mohtadi and P. S. Tuffs. Generalized Predictive Control – part 1 and 2. *Automatica*, Vol. 23, pp. 137-160, 1987.

[60] Manfred Morari. Model Predictive Control: Multivariable Control Technique of Choice in the 1990's. *Advances in Model Based Predictive Control,* Oxford Science Publications, 1994.

[61] Ronald Soeterboek. Predictive Control – A Unified Approcah. *Prentice Hall International Series in Systems and Control Engineering*, 1992.

[62] Manfred Morari and N. Lawrence Ricker. Model Predictive Control Toolbox User's Guide. *The Mathworks Partner Series*, 1995.

[63] Manfred Morari and Jay H. Lee. Model Predictive Control: Past, Present and Future. *Joint sixth International Symposium on Process Systems Engineering and thirtieth Eurpoean Symposium on Computer Aided Process Engineering, PSE-ESCAPE,* May 25-29, Norway, 1997.

[64] S. Joe Qin and Thomas A. Badgwell. An Overview of Industrial Model Predictive Control Technology. *Fifth International Conference on Chemical Process Control, AIChE and CACHE*, pp. 232-256, 1997.

[65] Peter Van Overschee and Bart De Moor. N4SID: Subspace Algorithms for the Identification of Combined Deterministic Stochastic Systems. *Automatica, Special Issue on Statistical Signal Processing and Control, Vol. 30, pp. 75-93,* 1994.

[66] C. E. Gracia and M. Morari. Internal Model Control 1 – A Unifying Review and Some Review Results. *Ind. Eng. Chem. Process Des. And Dev*, Vol. 21, pp. 308-332, 1982.

[67] K. S. Narendra and Kannan Parthasarathy. Identification and Control of Dynamic Systems Using Neural Networks. *IEEE Transaction on Neural Networks,* Vol. 1, 1990.

[68] Naveen V. Bhat, Peter A. Minderman, Thomas McAvoy and Nam Sung Wang. Modeling Chemical Process via Neural Networks. *IEEE,* 1980.

[69] Philip Tai, Helen A. Ryaciotaki-Boussalis and David Hollaway. Neural Network Implementation to Control Systems: A Survey of Algorithms and Techniques. *IEEE,* 1991.

[70] Si-Zhao Qin, Hong Te Su and Thomas J. McAvoy. Comparison of Four Neural Net Learning Methods for Dynamic System Identification. *IEEE,* 1992.

[71] J.R. Chen and P. Mars. The feasibility of Using MLP Neural Networks for System Identification. *School of Engineering and Applied Science, University of Durham,* UK, 1992.

[72] Takayuki Yamada and Tetsuro Yabuta. Dynamic System Identification using Neural Networks. *IEEE*, 1993.

[73] J. Sjoberg. Some Examples of Identification with Neural Network. *Lith-isy-r-1650, Department of Electrical Engineering, Linkoping University,* Sweden, 1994.

[74] Songwu Lu and Tamer Basar. Robust Non linear System Identification using Neural Network models. *IEEE,* 1995.

[75] Judistky, H. Hjalmarsson, A. Benveniste, B. Deylon, L. Ljung, J. Sjoberg and Q. Zhang. Nonlinear Black-Box Models in System Identification: Mathematical Foundations. *Automatica,* Vol. 31, 1995.

[76] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Daylon, Pierre-Yves Glorennec, H. Hjalmarsson and A. Jdistky. Nonlinear Black-Box Modeling in System Identification: a Unified Overview. *Automatica,* Vol. 31, 1995.

[77] C. Abdallah, G. L. Heilman, D. Hush and M. Georgiopoulos. An overview of Neural Networks Results for Systems and Control. *Department of Electrical and Computer*

Engineering, University of Central Florida, USA, 1995.

[78] H.N. Mhaskar. System Identification Using Neural Networks. *IEEE*, 1996.

[79] John O. Moody. The Dependence Identification Neural Network Construction Algorithm. *IEEE Transactions on Neural Networks*, 1996.

[80] J. A. K. Suykens, H. Bersini. Neural Control Theory: an Overview. *Journal A*, Vol. 37, No. 3, pp. 4-10, 1996.

[81] Mauro J. Atalla. Model Updating Using neural Networks. *PhD. Dissertation, Virginia Polytechnic Institute and State University*, USA, 1996.

[82] H.N. Al-Duwaish, A. O. Al Khowaiter and H.S. Al-Hajji. Inferential Prediction of Distillation Properties Using Neural Networks. *Saudi Aramco Journal of Technology*, 1997.

[83] Ole Sorensen. System Identification, Prediction, Simulation and Control with Neural Networks. *Inst. of Electronic Systems, Dept. of Control Engineering*, 1998.

[84] Songwu Lu and Tamer Basar. Robust Non linear System Identification using Neural Network models. *IEEE*, 1998.

[85] Jan Dimon Bendtsen and Ole Sorensen. Simulation, State Estimation and Control of Nonlinear Superheater Attemporator using Neural Networks. *American Control Conference*, Chicago, USA, 2000.

[86] J. Miima, W. Neimeier and B. Kraus. A Neural Network Approach to Modeling Geodetic Deformations. *First International Symposium on Robotic Statistics and Fuzzy Techniques in Geodesy and GIS*, Switzerland, 2001.

[87] Magnus Norgaard, Ole Ravn and Niels Kjolstad Poulsen. NNSYSID and NNCTRL Tools for System Identification and Control with Neural Networks. *Computing and Control Engineering Journal*, 2001.

Vitae

- Shiraz Amjad

- Born in Lahore, Pakistan on September 15, 1976

- Received Bachelor of Science (B.S.) degree in Electrical and Electronics Engineering from Eastern Mediterranean University, Turkish Republic of North Cyprus, Turkey in 1999.

- Joined King Fahd University of Petroleum and Minerals in September 2001.

- Publications:

[1] Shiraz Amjad and H. N. Al-Duwaish, 'Nonlinear System Identification using Neural Networks', *First GCC Industrial Electrical and Electronic Conference, Manama, Bahrain*, 2003.

[2] Shiraz Amjad and H. N. Al-Duwaish, 'Closed Loop Identification with Model Predictive Control: A Case Study', *Society of Instrument and Control Engineers, Fukui, Japan,* 2003.

[3] Hussain Al-Duwaish, Taha Othman and Shiraz Amjad, 'Neural Networks Identification of a Gas Plant NGL', *International Association of Science and Technology for Development (IASTED) annual conference on Intelligent Systems and Control, Salzburg, Austria,* 2003.

[4] Shiraz Amjad and Hussain N. Al-Duwaish, 'Model Predictive Control of Shell Benchmark Process', *10th IEEE International Conference on Electronics, Circuits and Systems, Sharjah, UAE*, 2003.

- *Email*: shiraz@kfupm.edu.sa